2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

VICAN: Very Efficient Calibration Algorithm for Large Camera Networks

Gabriel Moreira[1,2] Manuel Marques[2] João Paulo Costeira[2] and Alexander Hauptmann[1]

Fig. 1. Our method is motivated by applications in smart-retail that require fast pose estimation of large camera networks. The image in the center shows a 358m^2 store on which 342 cameras were mounted according to the locations and orientations depicted on the right. From over 100000 images of a moving object, as the three examples on the left illustrate, our algorithm computes accurate pose estimates in a short timespan.

***Abstract*— The precise estimation of camera poses within large camera networks is a foundational problem in computer vision and robotics, with broad applications spanning autonomous navigation, surveillance, and augmented reality. In this paper, we introduce a novel methodology that extends state-of-the-art Pose Graph Optimization (PGO) techniques. Departing from the conventional PGO paradigm, which primarily relies on camera-camera edges, our approach centers on the introduction of a dynamic element - any rigid object free to move in the scene - whose pose can be reliably inferred from a single image. Specifically, we consider the bipartite graph encompassing cameras, object poses evolving dynamically, and camera-object relative transformations at each time step. This shift not only offers a solution to the challenges encountered in directly estimating relative poses between cameras, particularly in adverse environments, but also leverages the inclusion of numerous object poses to ameliorate and integrate errors, resulting in accurate camera pose estimates. Though our framework retains compatibility with traditional PGO solvers, its efficacy benefits from a custom-tailored optimization scheme. To this end, we introduce an iterative primal-dual algorithm, capable of handling large graphs. Empirical benchmarks, conducted on a new dataset of simulated indoor environments, substantiate the efficacy and efficiency of our approach.**

I. INTRODUCTION

In this paper, we address the problem of localizing a network of calibrated cameras distributed in 3D space. The importance of this task lies in its role as a prerequisite: accurately determining the pose of each camera within a shared reference frame is paramount for enabling spatial awareness, in order to facilitate applications ranging from augmented reality and tracking to action recognition. Without precise camera pose estimates, the capabilities of camera networks are significantly limited.

Modern approaches to solve the camera network localization problem fall in one of two groups. Structure-from-Motion (SfM) systems, such as COLMAP [1], simultaneously reconstruct the 3D scene and solve for the camera poses. In large scenes however, this can be prohibitive from a computational standpoint. On the other hand, Pose Graph Optimization (PGO) approaches [2], [3], [4], [5], [6], [5], also referred to as motion synchronization [7], [8] or averaging, [9], [10] and consensus algorithms [11], [12], provide an expedited two-step way of solving the problem. Initially, each image is probed for keypoints, which are then matched with those extracted from other images. For each image pair with enough point correspondences, the relative direction and rotation between the respective cameras is estimated via the epipolar constraint. Subsequently, a non-convex optimization problem is solved in order to localize the cameras in a common reference frame. Its optimum is the maximum likelihood point estimate (MLE) of the set of camera poses.

In recent years, the MLE formulation of motion synchronization relying on the bi-dual of the original non-convex problem, a Semidefinite Program (SDP), has seen wide adoption in the literature pertaining to Visual Simultaneous Localization in Mapping (SLAM) [3], [8] and rotation synchronization [13], [14], [15], [10]. Under the moderate noise levels commonly found in most applications, this SDP relaxation is tight, yielding the optimum of the PGO problem. Fast and non-iterative spectral approximations for the camera rotations have also been proposed [7], [16], yielding estimates surprisingly close to the optimum [17].

[1]Language Technologies Institute, Carnegie Mellon University, School of Computer Science, USA {gmoreira, alex}@cs.cmu.edu.
[2]Institute for Systems and Robotics, Instituto Superior Técnico, Lisboa, Portugal {manuel, jpc}@isr.tecnico.ulisboa.pt.

Despite these notable achievements, for the same camera network topology, the synchronization step is only as good as the pairwise transformations [17], [8]. In low-light, low-texture, or occluded regions of the environment, the acquisition of precise point correspondences becomes challenging. This subsequently hinders the accurate estimation of relative camera poses, ultimately restricting the efficacy of PGO. These difficulties are further exacerbated in poorly connected, or 'bottleneck', regions of the graph, which obstruct the redistribution of the pairwise errors [18].

As a means to circumvent the aforementioned issues, we introduce a novel methodology. Alongside the static camera network, we incorporate a known dynamic rigid object, within the field-of-view of a subset of the cameras, such that its pose can be reliably inferred from a single image. Our focal point is the bipartite pose graph comprising the ensemble of static camera poses, the object poses evolving over time, and the set of relative transformations binding them together. The indirect estimation of camera poses through the interplay of camera-object and pairwise object transformations allows us to mitigate some of the adverse aspects of camera pose estimation, by integrating a large number of camera-object transformations.

While our novel problem formulation initially appears amenable to conventional motion synchronization solvers such as g2o [2], GTSAM [5], SE-Sync [8] and MAKS [16], which exhibit a node-agnostic nature regarding graph elements, it is important to note that these methods were originally tailored for graphs composed exclusively of camera nodes. Within the PGO literature, which predominantly targets SLAM applications, datasets frequently encompass tens of thousands of pairwise transformations. In our framework, the introduction of moving objects adds a multitude of new nodes and edges with each time step. Consequently, existing solvers prove inadequate in handling these augmented graphs. To this end, we propose a fast iterative primal-dual algorithm to solve for the cameras and object rotations.

In summary, we make the following contributions

- We propose a new MLE formulation of the camera network localization problem via a bipartite camera-object pose graph.
- We introduce a new iterative method capable of handling a large number of object nodes from image streams. When compared to ground-truth poses, our method achieves average rotation and translation errors of 0.04deg and 3cm in a 358m^2 shop covered by 342 cameras (Fig. 1), and of 0.07deg and 0.7cm in a 72m^2 room covered by 25 cameras (Fig. 3).

The remainder of the paper[1] is organized as follows. We introduce our formulation of the problem and the iterative scheme to solve it in Section II. In Section III we present a new image and 3D dataset for camera network pose estimation and benchmarks of our method therein. Concluding remarks are drawn in Section IV.

[1]Code, dataset and extended version of the paper with full derivations are available in `https://github.com/gabmoreira/vican`.

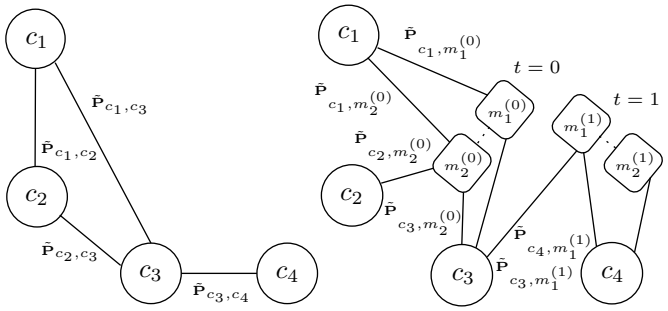

Fig. 2. Standard PGO (left) vs our augmentation with object nodes (right). Pairwise relative transformations are shown as $\tilde{\mathbf{P}}_{\cdot,\cdot}$. The i-th camera node is c_i, and $m_i^{(t)}$ is the i-th object node at time t.

II. PROPOSED METHOD

Consider a static network of calibrated cameras, indexed by $c \in \mathcal{C}$, with $C = |\mathcal{C}|$. The pose of the c-th camera w.r.t. an arbitrary but fixed reference frame is given by $\mathbf{P}_c \in \mathrm{SE}(3) \subset \mathbb{R}^{4\times4}$, or equivalently $(\mathbf{R}_c, \mathbf{t}_c) \in \mathrm{E}(3)$. We then consider a rigid object that is free to move in $\mathbb{R}^3$. We decompose this object into M nodes which may correspond *e.g.*, to M fiducial markers, indexed via $m \in \mathcal{M}$. We indicate the pose of the m-th object node at time t, for $t \in \mathcal{T}$, as $\mathbf{P}_{m^{(t)}}$. Our goal is to estimate $\{c \in \mathcal{C} : \mathbf{P}_c\}$ using pairwise camera-object relative transformations $\tilde{\mathbf{P}}_{c,m^{(t)}}$, obtained *e.g.*, via Perspective-n-Point (PnP). We assume the following convention for the composition of transformations $\mathbf{P}_{ij} = \mathbf{P}_i\mathbf{P}_j^{-1}$.

Denote by $G = (\mathcal{V}, \mathcal{E})$, the bipartite graph with $\mathcal{V} = \mathcal{C} \cup (\mathcal{T} \times \mathcal{M})$ *i.e.*, each camera and each object node at each time instant are associated with a vertex. The latter component of the graph is assumed to be much larger than the former. The edge $(c, m^{(t)}) \in \mathcal{E}$ corresponds to an image captured by camera c at time t of the m-th object node. The relative pose measurement of this node w.r.t. the camera is denoted as $(\tilde{\mathbf{t}}_{c,m^{(t)}}, \tilde{\mathbf{R}}_{c,m^{(t)}})$. A comparison between traditional PGO and our formulation is showcased in Fig. 2.

A. Maximum likelihood estimation

Under the assumption of isotropic Gaussian noise with precision $\tau_{c,m^{(t)}}$ in the translation variables and isotropic Langevin noise [18] with concentration $k_{c,m^{(t)}}$ in the rotations, the negative log-likelihood function (NLL) [3] reads as

$$-\log f = \sum_{c,m^{(t)}\in\mathcal{E}} \frac{\tau_{c,m^{(t)}}}{2} \|\tilde{\mathbf{t}}_{c,m^{(t)}} - \mathbf{t}_c + \mathbf{R}_c\mathbf{R}_{m^{(t)}}^\top \mathbf{t}_{m^{(t)}}\|_2^2 - k_{c,m^{(t)}} \mathrm{Tr}\left(\tilde{\mathbf{R}}_{c,m^{(t)}}\mathbf{R}_{m^{(t)}}\mathbf{R}_c^\top\right). \quad (1)$$

Instead of optimizing jointly the poses of the C cameras and the $M \times T$ object nodes, we rely on the rigidity of the object and further assume that we have access to the pairwise relative transformation between any two object nodes at the same time instant, $m_i^{(t)}$ and $m_j^{(t)}$, denoted $\bar{\mathbf{P}}_{m_i^{(t)},m_j^{(t)}}$. We will henceforth drop the time superscript and write

$\bar{\mathbf{P}}_{m_i,m_j} = (\bar{\mathbf{R}}_{m_i,m_j}, \bar{\mathbf{t}}_{m_i,m_j})$. Without loss of generality, we have thus,

$$\forall t \in \mathcal{T}\ \forall m \in \mathcal{M}\ \begin{cases} \mathbf{R}_{m^{(t)}} = \bar{\mathbf{R}}_{m,m_1}\mathbf{R}_{m_1^{(t)}} \\ \mathbf{t}_{m^{(t)}} = \bar{\mathbf{R}}_{m,m_1}\mathbf{t}_{m_1^{(t)}} + \bar{\mathbf{t}}_{m,m_1} \end{cases} \tag{2}$$

i.e., the pose of each marker at time t is given relatively to the pose of the object in that instant, $\mathbf{P}_{m_1^{(t)}}$. Thus, we only optimize over $\{t \in \mathcal{T} : \mathbf{P}_{m_1^{(t)}}\}$. Introducing (2) in the first term of the NLL (1) yields

$$\sum_t \sum_{c,m^{(t)}\in\mathcal{E}} \frac{\tau_{c,m^{(t)}}}{2} \Big\| \tilde{\mathbf{t}}_{c,m^{(t)}} + \mathbf{R}_c\mathbf{R}_{m^{(t)}}^\top \bar{\mathbf{t}}_{m,m_1} - \left(\mathbf{t}_c - \mathbf{R}_c\mathbf{R}_{m^{(t)}}^\top \bar{\mathbf{R}}_{m,m_1}\mathbf{t}_{m_1^{(t)}}\right) \Big\|_2^2. \tag{3}$$

The second term becomes

$$\sum_{t\in\mathcal{T}c\in\mathcal{C}} -\mathrm{Tr}\left(\sum_{m^{(t)}\in N(c)} \left(k_{c,m^{(t)}} \tilde{\mathbf{R}}_{c,m^{(t)}} \bar{\mathbf{R}}_{m,m_1}\right) \mathbf{R}_{m_1^{(t)}} \mathbf{R}_c^\top \right), \tag{4}$$

where $N(c)$ denotes the neighborhood of the c-th camera, *i.e.*, the set of markers that are visible from it. In summary, we simplify the graph by merging all the objects nodes at each time instant t, into a single object node at time t, corresponding to the marker m_1.

We will now turn our attention to the minimization of the non-convex NLL with terms (3) and (4). To this end we will adopt the strategy used in [6], [16], wherein the problem is decoupled in: non-convex rotations synchronization by minimizing (4) and a least-squares problem that yields the translations by minimizing (3). The main difficulty of this method lies in optimally and efficiently solving the former.

B. Synchronization of rotations

Define the block-matrix $\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}} \in \mathbb{R}^{3C\times 3T}$, indexed via c and t, as

$$[\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}]_{c,t} := \sum_{m:\ m^{(t)}\in N(c)} k_{c,m^{(t)}} \tilde{\mathbf{R}}_{c,m^{(t)}} \bar{\mathbf{R}}_{m,m_1}. \tag{5}$$

We assume $k_{c,m^{(t)}} = 0$ if $(c, m^{(t)}) \notin \mathcal{E}$. The block-entry c,t of $\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}$ is a weighted sum of rotation measurements from camera c to marker m_1 at time t. Define the pairwise block matrix $\tilde{\mathbf{R}} \in \mathbb{R}^{3(C+T)\times 3(C+T)}$ as

$$\tilde{\mathbf{R}} := \begin{bmatrix} \mathbf{0} & \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}} \\ \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top & \mathbf{0} \end{bmatrix} \tag{6}$$

and the block-vector $\mathbf{R} \in \mathrm{SO}(3)^{3(C+T)} \subset \mathbb{R}^{3(C+T)\times 3}$ as

$$\mathbf{R} := \begin{bmatrix} \mathbf{R}_{\mathcal{C}}^\top & \mathbf{R}_{\mathcal{T}}^\top \end{bmatrix}^\top = \begin{bmatrix} \mathbf{R}_{c_1}^\top & \dots & \mathbf{R}_{cC}^\top & \mathbf{R}_{m_1^{(1)}}^\top & \dots & \mathbf{R}_{m_1^{(T)}}^\top \end{bmatrix}^\top. \tag{7}$$

The set of rotations that minimize (4) is the solution of the rotation synchronization problem

$$\min_{\mathbf{R}\in\mathrm{SO}(3)^{(C+T)}} -\mathrm{Tr}\left(\tilde{\mathbf{R}}\mathbf{R}\mathbf{R}^\top\right). \tag{8}$$

The adjacency matrix associated with this problem, denoted $\mathbf{A} \in \mathbb{R}^{(C+T)\times(C+T)}$, can be written in the same fashion

$$\mathbf{A} := \begin{bmatrix} \mathbf{0} & \mathbf{A}_{\mathcal{C}\mathcal{T}} \\ \mathbf{A}_{\mathcal{C}\mathcal{T}}^\top & \mathbf{0} \end{bmatrix}, \tag{9}$$

with entry c,t given by

$$[\mathbf{A}_{\mathcal{C}\mathcal{T}}]_{c,t} := \sum_{m:\ m^{(t)}\in N(c)} k_{c,m^{(t)}}. \tag{10}$$

The KKT and optimality conditions of Problem (8), which can be found in [13], are as follows. Denote by $\mathbf{\Lambda} \in \mathbb{R}^{3(C+T)\times 3(C+T)}$, the symmetric 3×3 block-diagonal dual variable. The first-order KKT condition reads as $\mathbf{\Lambda}\mathbf{R} = \tilde{\mathbf{R}}\mathbf{R}$. If a primal-dual pair $(\mathbf{R}^*, \mathbf{\Lambda}^*)$ verifies the KKT, then $\mathbf{\Lambda}^* - \tilde{\mathbf{R}} \succeq 0$ is a sufficient optimality condition.

Theorem 1: Assume strong duality holds and denote by $(\mathbf{\Lambda}^*, \mathbf{R}^*)$ a primal-dual optimal pair of Problem (8), where the dual variable $\mathbf{\Lambda}$ is decomposed as

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{\Lambda}_{\mathcal{C}} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{\mathcal{T}} \end{bmatrix}, \tag{11}$$

with $\mathbf{\Lambda}_{\mathcal{C}} \in \mathbb{R}^{3C\times 3C}$ and $\mathbf{\Lambda}_{\mathcal{T}} \in \mathbb{R}^{3T\times 3T}$. The optimal camera poses $\mathbf{R}_{\mathcal{C}}^*$ are a solution of

$$\min_{\mathbf{R}_{\mathcal{C}}\in\mathrm{SO}(3)^C} -\mathrm{Tr}\left(\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{*^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\right)\mathbf{R}_{\mathcal{C}}\mathbf{R}_{\mathcal{C}}^\top\right) \tag{12}$$

and the optimal object poses $\mathbf{R}_{\mathcal{T}}^*$ a solution of

$$\min_{\mathbf{R}_{\mathcal{T}}\in\mathrm{SO}(3)^T} -\mathrm{Tr}\left(\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\mathbf{\Lambda}_{\mathcal{C}}^{*^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\right)\mathbf{R}_{\mathcal{T}}\mathbf{R}_{\mathcal{T}}^\top\right). \tag{13}$$

This results leads us then to the following block-coordinate descent updates, which produce feasible primal-dual pairs $(\mathbf{R}^{(k)}, \mathbf{\Lambda}^{(k)})$ *i.e.*, at the k-th iteration $\mathbf{R}^{(k)} \in \mathrm{SO}(3)^{(C+T)}$ and $\mathbf{\Lambda}^{(k)} - \tilde{\mathbf{R}} \succeq 0$, that maximize the dual function *i.e.*, $-\mathrm{Tr}(\mathbf{\Lambda}^{(k)}) \leq -\mathrm{Tr}(\mathbf{\Lambda}^{(k+1)})$,

$$\mathbf{R}_{\mathcal{C}}^{(k)} = \operatorname*{argmin}_{\mathbf{R}_{\mathcal{C}}} -\mathrm{Tr}\left(\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\right)\mathbf{R}_{\mathcal{C}}\mathbf{R}_{\mathcal{C}}^\top\right) \tag{14}$$

$$\mathbf{\Lambda}_{\mathcal{C}}^{(k)} = \mathrm{blkdiag}\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\mathbf{R}_{\mathcal{C}}^{(k)}\mathbf{R}_{\mathcal{C}}^{(k)^\top}\right) \tag{15}$$

$$\mathbf{R}_{\mathcal{T}}^{(k)} = \operatorname*{argmin}_{\mathbf{R}_{\mathcal{T}}} -\mathrm{Tr}\left(\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\mathbf{\Lambda}_{\mathcal{C}}^{(k)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\right)\mathbf{R}_{\mathcal{T}}\mathbf{R}_{\mathcal{T}}^\top\right) \tag{16}$$

$$\mathbf{\Lambda}_{\mathcal{T}}^{(k)} = \mathrm{blkdiag}\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top\mathbf{\Lambda}_{\mathcal{C}}^{(k)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{R}_{\mathcal{T}}^{(k)}\mathbf{R}_{\mathcal{T}}^{(k)^\top}\right) \tag{17}$$

We assume that strong duality holds. Then, the update of $\mathbf{\Lambda}_{\mathcal{C}}^{(k)}$ in (15) can be written as

$$\mathbf{\Lambda}_{\mathcal{C}}^{(k)} = \operatorname*{argmax}_{\mathbf{\Lambda}:\ \mathbf{\Lambda}_{\mathcal{C}} - \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^\top \succeq 0} -\mathrm{Tr}(\mathbf{\Lambda}_{\mathcal{C}}). \tag{18}$$

From Schur's complement, the positive semidefinite constraint is equivalent to $\mathbf{\Lambda} - \tilde{\mathbf{R}} \succeq 0$ and $\mathbf{\Lambda}_{\mathcal{T}} = \mathbf{\Lambda}_{\mathcal{T}}^{(k-1)}$. We thus rewrite the update as

$$\mathbf{\Lambda}_{\mathcal{C}}^{(k)} = \operatorname*{argmax}_{\mathbf{\Lambda}_{\mathcal{C}}:\ \mathbf{\Lambda} - \tilde{\mathbf{R}} \succeq 0,\ \mathbf{\Lambda}_{\mathcal{T}} = \mathbf{\Lambda}_{\mathcal{T}}^{(k-1)}} -\mathrm{Tr}(\mathbf{\Lambda}). \tag{19}$$

Same reasoning applies for update (17), which yields

$$\mathbf{\Lambda}_{\mathcal{T}}^{(k)} = \operatorname*{argmax}_{\mathbf{\Lambda}_{\mathcal{T}}:\ \mathbf{\Lambda} - \tilde{\mathbf{R}} \succeq 0,\ \mathbf{\Lambda}_{\mathcal{C}} = \mathbf{\Lambda}_{\mathcal{C}}^{(k)}} -\mathrm{Tr}(\mathbf{\Lambda}). \tag{20}$$

Therefore, the sequence $\{-\mathrm{Tr}(\mathbf{\Lambda}^{(k)})\}_{k\geq 1}$ is dual feasible and non-decreasing, as desired. The caveat with this approach is that, assuming $T >> C$, (16) is more expensive to solve than (14), and thus the gain is minimal.

We address this by replacing the minimization in (16) by a step of the Frank-Wolfe method, also known as the generalized power method (GPM) [19], which has a closed-form solution. We have then

$$\mathbf{R}_{\mathcal{C}}^{(k)} = \underset{\mathbf{R}_{\mathcal{C}}}{\mathrm{argmin}} -\mathrm{Tr}\left(\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\right)\mathbf{R}_{\mathcal{C}}\mathbf{R}_{\mathcal{C}}^{\top}\right) \quad (21)$$

$$\mathbf{\Lambda}_{\mathcal{C}}^{(k)} = \mathrm{blkdiag}\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\mathbf{R}_{\mathcal{C}}^{(k)}\mathbf{R}_{\mathcal{C}}^{(k)^{\top}}\right) \quad (22)$$

$$\mathbf{U}_t\mathbf{\Sigma}_t\mathbf{V}_t^{\top} \overset{\mathrm{SVD}}{=} \sum_{c}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}_{c,t}}^{\top}\mathbf{R}_{\mathcal{C}_c}^{(k)},\ t\in\mathcal{T} \quad (23)$$

$$\mathbf{\Lambda}_{\mathcal{T}_t}^{(k)} = \mathbf{U}_t\mathbf{\Sigma}_t\mathbf{U}_t^{\top},\ t\in\mathcal{T}. \quad (24)$$

To see why this is a reasonable approximation note that, given $\mathbf{R}_{\mathcal{C}}^{(k)}$, the Frank-Wolfe, or GPM iteration is

$$\mathbf{R}_{\mathcal{T}}^{(k)} = \underset{\mathbf{R}_{\mathcal{T}}\in\mathrm{SO}(3)^T}{\mathrm{argmin}} -\mathrm{Tr}\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{R}_{\mathcal{C}}^{(k)}\mathbf{R}_{\mathcal{T}}^{\top}\right), \quad (25)$$

which has a closed form solution given by the orthogonal projection $\mathbf{R}_{\mathcal{T}_t}^{(k)} = \mathbf{U}_t\mathbf{V}_t^{\top}$. Close to the optimum, we must have $\mathbf{R}_{\mathcal{C}}^{(k)} \approx \mathbf{\Lambda}_{\mathcal{C}}^{(k)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\mathbf{R}_{\mathcal{T}}^{(k)}$, from the KKT condition. Hence, $\mathbf{\Lambda}_{\mathcal{T}_t}^{(k)} \approx \mathbf{U}_t\mathbf{\Sigma}_t\mathbf{U}_t^{\top}$, with equality at the optimum.

In order to set $\mathbf{\Lambda}_{\mathcal{T}}^{(0)}$, we leverage the spectral initialization [7], [16], whose distance to the optimum is dependent on the connectivity of the graph and the noise level [17]. Given the diagonal graph degree matrix $\mathbf{\Delta} = \mathrm{Diag}(\mathbf{\Delta}_{\mathcal{C}}, \mathbf{\Delta}_{\mathcal{T}})$, where

$$\mathbf{\Delta}_{i,i} = \sum_{j}\mathbf{A}_{i,j}, \quad (26)$$

this approximation consists of projecting the eigenspace of $\mathbf{\Delta}\otimes\mathbf{I}_3 - \tilde{\mathbf{R}}$ corresponding to the three smallest eigenvalues, to $\mathrm{SO}(3)$. From the KKT condition, [15] interpreted the degree matrix as a dual variable approximation, where $\mathbf{\Lambda}^* \approx \mathbf{\Delta}\otimes\mathbf{I}_3$. It can be shown that $\mathbf{\Delta}\otimes\mathbf{I}_3$ is, in fact, dual feasible. Thus, we set $\mathbf{\Lambda}_{\mathcal{T}}^{(0)}$ to $\mathbf{\Delta}_{\mathcal{T}}\otimes\mathbf{I}_3$.

In this approach, the bulk of the optimization is carried out in update (21), which can be interpreted as a rotations synchronization problem in the 2-th power graph, wherein there are edges between any two cameras of $\mathcal{C}$ which are connected via a third vertex in $\mathcal{T}$. The pairwise *measurements* (no longer rotations) of this power graph are the block entries of the block matrix $\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}$ and the adjacency matrix is taken to be $\mathbf{A}_{\mathcal{C}\mathcal{T}}\mathbf{\Delta}_{\mathcal{T}}^{-1}\mathbf{A}_{\mathcal{C}\mathcal{T}}^{\top}$.

The optimization in (21) fits therefore the criteria of the primal-dual method from [15]. Given $\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}$ and $\mathbf{A}_{\mathcal{C}\mathcal{T}}\mathbf{\Delta}_{\mathcal{T}}^{-1}\mathbf{A}_{\mathcal{C}\mathcal{T}}^{\top}$, the dual $\mathbf{\Lambda}_{\mathcal{C}}^{(k)}$ is initialized as the graph degree matrix of the power graph *i.e.*, $\forall c\in\mathcal{C}\ \mathbf{\Lambda}_{\mathcal{C}_{i,i}}^{(k)} \leftarrow \sum_j(\mathbf{A}_{\mathcal{C}\mathcal{T}}\mathbf{\Delta}_{\mathcal{T}}^{-1}\mathbf{A}_{\mathcal{C}\mathcal{T}}^{\top})_{i,j}$. $\mathbf{R}_{\mathcal{C}}^{(k)}$ is then obtained by orthogonally projecting the eigenspace spanned by the eigenvectors of $\mathbf{\Lambda}_{\mathcal{C}}^{(k)} - \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}$ corresponding to the 3 smallest eigenvalues to $\mathrm{SO}(3)^C$. $\mathbf{\Lambda}_{\mathcal{C}}^{(k)}$ is updated according to

$$\mathbf{U}_i\mathbf{\Sigma}_i\mathbf{V}_i^{\top} \overset{\mathrm{SVD}}{\leftarrow} \sum_j\left(\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{(k-1)^{-1}}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\right)_{i,j}\mathbf{R}_{\mathcal{C}_j}^{(k)} \quad (27)$$

$$\mathbf{\Lambda}_{\mathcal{C}_i}^{(k)} \leftarrow \mathbf{U}_i\mathbf{\Sigma}_i\mathbf{U}_i^{\top}. \quad (28)$$

and the iterations repeat until convergence, upon which (22) is automatically verified. The entire method is laid out in Algorithm (1).

C. Solution for translations

Minimizing the translations term (3) is equivalent to a PGO translation problem with pairwise measurements

$$\tilde{\mathbf{t}}_{c,m^{(t)}} := \tau_{c,m^{(t)}}(\tilde{\mathbf{t}}_{c,m^{(t)}} + \mathbf{R}_c^*\mathbf{R}_{m^{(t)}}^{*^{\top}}\bar{\mathbf{t}}_{m,m_1}). \quad (29)$$

Let $\tilde{\mathbf{T}}\in\mathbb{R}^{3|\mathcal{E}|}$ be a vector containing all the pairwise edges stacked. Define the vector of variables $\mathbf{T}\in\mathbb{R}^{3(C+T)}$

$$\mathbf{T} := \begin{bmatrix}\mathbf{t}_{c_1}^{\top} & \dots & \mathbf{t}_{c_C}^{\top} & \mathbf{t}_{m_1^{(1)}}^{\top} & \dots & \mathbf{t}_{m_1^{(T)}}^{\top}\end{bmatrix}^{\top}, \quad (30)$$

and the incidence block matrix $\mathbf{J}\in\mathbb{R}^{3|\mathcal{E}|\times 3(C+T)}$ such that if e indexes $(c, m^{(t)})\in\mathcal{E}$ then

$$\mathbf{J}_{e,c} := \tau_{c,m}\mathbf{I}_3 \quad (31)$$

$$\mathbf{J}_{e,t} := -\tau_{c,m}\mathbf{R}_c^*\mathbf{R}_m^{*^{\top}}\bar{\mathbf{R}}_{m,m_1}. \quad (32)$$

The translation problem has the least-squares formulation

$$\min_{\mathbf{T}\in\mathbb{R}^{3(C+T)}}\|\tilde{\mathbf{T}} - \mathbf{J}\mathbf{T}\|_2^2. \quad (33)$$

Due to the size of the matrices involved, the conjugate gradient method is used to solve (33).

We conclude this section with a note on the feasibility of a streaming version of Algorithm 1. At $T+1$, suppose the object moves to a new pose. For all cameras $c\in\mathcal{C}$ with the object in their field-of-view, the relative poses to each of the object nodes, $\tilde{\mathbf{P}}_{c,m^{(T+1)}}$, can be estimated in parallel. Instead of then starting the algorithm again, the initializations in lines 1-5, which would involve increasingly larger matrices for large T, can be replaced by updating the initializations of $\mathbf{L}\in\mathbb{R}^{3C\times 3C}$ and $\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{-1}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\in\mathbb{R}^{3C\times 3C}$ directly. The optimization problem that follows is only dependent on C and the object poses update is linear in T.

III. EXPERIMENTS

A. Indoor scenes dataset

Given that no existing camera network pose estimation dataset is suitable to benchmark the proposed method, we put forward a novel dataset of two camera pose estimation scenes, shown in Figs. 1 and 3. The former consists of a large L-shaped shop with different pieces of furniture and walls blocking the view, whereas the latter is a rectangular room free of occlusions. Images were ray-traced from 3D models of these indoor environments, which were fitted with camera arrays covering them entirely. Each camera has distinct intrinsic ground-truth calibration parameters, that we provide. Given an input value of T time steps, renders from cameras with the chosen object in their field-of-view were obtained procedurally by iteratively placing it

Algorithm 1 Bipartite rotations synchronization

Require: $\{\bar{\mathbf{P}}_{m,m'}\}_{(m,m')\in\mathcal{M}}, \{\tilde{\mathbf{P}}_{c,m^{(t)}}\}_{(c,m^{(t)})\in\mathcal{E}}, \delta, K$

$\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}_{c,t}} \leftarrow \sum_{m:\ m^{(t)}\in N(c)} k_{c,m^{(t)}} \tilde{\mathbf{R}}_{c,m^{(t)}} \mathbf{R}_{m,m_1}$
$\mathbf{A}_{\mathcal{C}\mathcal{T}_{c,t}} \leftarrow \sum_{m:\ m^{(t)}\in N(c)} k_{c,m^{(t)}}$ ▷ Adjacency
$\mathbf{\Delta}_{\mathcal{T}} \leftarrow \mathrm{Diag}\left(\mathbf{A}_{\mathcal{C}\mathcal{T}}^{\top}\mathbf{1}_C\right)$ ▷ Degree
$\mathbf{\Lambda}_{\mathcal{T}} \leftarrow \mathbf{\Delta}_{\mathcal{T}} \otimes \mathbf{I}_3$ ▷ Spectral init.
$\mathbf{\Lambda}_{\mathcal{C}} \leftarrow \mathrm{Diag}(\mathbf{A}_{\mathcal{C}\mathcal{T}}\mathbf{\Delta}_{\mathcal{T}}^{-1}\mathbf{A}_{\mathcal{C}\mathcal{T}}^{\top}\mathbf{1}_C) \otimes \mathbf{I}_3$ ▷ Spectral init.
$k \leftarrow 0$ ▷ Iteration counter
while $|\lambda_3| > \delta$ and $k < K$ **do**
  $\mathbf{L} \leftarrow \mathbf{\Lambda}_{\mathcal{C}} - \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{-1}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}$
  $\mathbf{V}, \{\lambda_i\}_{i=1,\dots,3} \leftarrow \mathrm{eigensolver}(\mathbf{L}, 3)$
  for $i < C$ **do** ▷ $\mathcal{C}$ primal update
    $\mathbf{J}_i, \mathbf{\Sigma}_i, \mathbf{H}_i^{\top} \leftarrow \mathrm{SVD}(\mathbf{V}_i)$
    $\mathbf{R}_{\mathcal{C}_i} \leftarrow \mathbf{J}\mathrm{diag}(1, 1, \det(\mathbf{J}_i\mathbf{H}_i^{\top}))\mathbf{H}_i^{\top}$
  end for
  $\mathbf{S} \leftarrow \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}\mathbf{\Lambda}_{\mathcal{T}}^{-1}\tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\mathbf{R}_{\mathcal{C}}$
  for $i < C$ **do** ▷ $\mathcal{C}$ dual update
    $\mathbf{J}_i, \mathbf{\Sigma}_i, \mathbf{H}_i^{\top} \leftarrow \mathrm{SVD}(\mathbf{S}_i)$
    $\mathbf{\Lambda}_{\mathcal{C}_{i,i}} \leftarrow \mathbf{J}_i\mathbf{\Sigma}_i\mathbf{J}_i^{\top}$
  end for
  $\mathbf{G} \leftarrow \tilde{\mathbf{R}}_{\mathcal{C}\mathcal{T}}^{\top}\mathbf{R}_{\mathcal{C}}$
  for $i < T$ **do** ▷ $\mathcal{T}$ dual update
    $\mathbf{J}_i, \mathbf{\Sigma}_i, \mathbf{H}_i^{\top} \leftarrow \mathrm{SVD}(\mathbf{G}_i)$
    $\mathbf{\Lambda}_{\mathcal{T}_{i,i}} \leftarrow \mathbf{J}_i\mathbf{\Sigma}_i\mathbf{J}_i^{\top}$
  end for
  $k \leftarrow k + 1$
end while

T times in random poses, inside the scene. The pose of the object at each time step was sampled uniformly, ensuring no intersections with the environment. An overview of the datasets is shown in Table I. The 3D models, images and the procedural rendering script are available online.

B. Camera-object transformations and graph construction

The object used in all datasets was a cube with side 0.575m, covered in 24 arUco markers, each with side 0.276m, as represented in Fig. 4. Given the set of images obtained as described above, the camera-object edges of the pose graph were obtained via OpenCV's arUco library for corner detection and the implementation of the P4P method provided therein. Further refinement of the camera-marker transformations was done via Levenberg-Marquardt. Finally, edges were filtered according to the average reprojection error. The noise models used for the concentrations $k_{c,m}$ and precisions $\tau_{c,m}$ were derived empirically, based on the detected arUco area in the respective image, as detection accuracy wanes for larger distances and angles.

C. Object calibration

In order to estimate the relative transformations between any two markers m and m', the cube was fixed in place and 1000 images from a single camera moving around it were captured, from which the pairwise measurements were computed as described above. The estimates $\{\mathbf{P}_m\mathbf{P}_{m'}^{-1}\}_{(m,m')\in\mathcal{M}}$ were obtained by minimizing the NLL (1). Since this graph of the pairwise measurements is bipartite as well, Algorithm 1 was employed, with the edges reversed *i.e.*, the set of the static cameras $\mathcal{C}$ is the set of 24 static markers and the set of markers of the moving cube $\mathcal{M}\times\mathcal{T}$ becomes the mobile camera. Our algorithm converges in 2 iterations, with an average time per iteration of $0.07s$ in Python.

TABLE I
DATASET CHARACTERISTICS

Dataset	Area (m^2)	Cams $\lvert\mathcal{C}\rvert$	Time $\lvert\mathcal{T}\rvert$	Edges $\lvert\mathcal{E}\rvert$
SmallRoom50			50	768
SmallRoom500	72	25	500	8067
SmallRoom5K			5000	80036
LargeShop500			500	43983
LargeShop5K	358	342	5000	438906
LargeShop10K			10000	874700

TABLE II
POSE ESTIMATION RESULTS: ERRORS W.R.T. GROUND-TRUTH

Dataset	avg δ_R	max δ_R	avg δ_t	max δ_t	t (s/it)
SmallRoom50	0.54	5.33	0.036	0.285	0.02
SmallRoom500	0.09	0.21	0.008	0.016	0.04
SmallRoom5K	**0.07**	**0.13**	**0.007**	**0.012**	**0.22**
LargeShop500	0.09	0.46	0.040	0.093	0.16
LargeShop5K	0.05	0.15	0.031	0.064	0.48
LargeShop10K	**0.04**	**0.13**	**0.030**	**0.064**	**0.79**

D. Camera pose estimation results

Due to the problem's inherent *gauge symmetry*, given a set of camera pose estimates $\{\mathbf{P}_c\}_{c\in\mathcal{C}}$, for any $\mathbf{H} \in \mathrm{SE}(3)$ the set $\{\mathbf{P}_c\mathbf{H}\}_{c\in\mathcal{C}}$ is also a solution. In order to establish comparisons with the latent camera poses, we consider the equivalence relation $\{\mathbf{P}_c\}_{c\in\mathcal{C}} \sim \{\mathbf{P}'_c\}_{c\in\mathcal{C}} \Leftrightarrow \exists\mathbf{H} \in \mathrm{SE}(3) : \mathbf{P}_c = \mathbf{P}'_c\mathbf{H}\ \forall c \in \mathcal{C}$. We define the orbit distance in the quotient manifold $\mathrm{SE}(3)^C/\sim$ as in [17] *i.e.*,

$$d_{\mathrm{SE}(3)^C/\sim}\left(\{\mathbf{P}_c\}_{c\in\mathcal{C}}, \{\mathbf{P'}_c\}_{c\in\mathcal{C}}\right) = \min_{\mathbf{H}\in\mathrm{SE}(3)} d_{\mathrm{SE}(3)^C}\left(\{\mathbf{P}_c\}_{c\in\mathcal{C}}, \{\mathbf{P'}_c\mathbf{H}\}_{c\in\mathcal{C}}\right). \quad (34)$$

For this choice of gauge, we computed the mean and maximum translation and rotation errors, defined as $\delta_R := \angle(\bar{\mathbf{R}}_c, \hat{\mathbf{R}}_c)$ in degrees and $\delta_t := \|\bar{\mathbf{t}}_c - \hat{\mathbf{t}}_c\|_2$ in meters, respectively. Results from our Python implementation of Algorithm 1, using SciPy sparse matrices and the sparse eigensolver from NumPy (`eigs`), are shown in Table II. Convergence was achieved after two iterations in all scenes. The best results are highlighted.

The different network topologies lead to different graph connectivities. As expected, better results are consistently obtained for better connected graphs. Specifically, the small room scene, devoid of occlusions, lends itself to highly precise camera pose estimation. As anticipated, within both datasets, we note substantial reductions in pose estimation errors with the increase in the number of object poses, without a dramatic rise in Algorithm 1's time per iteration.

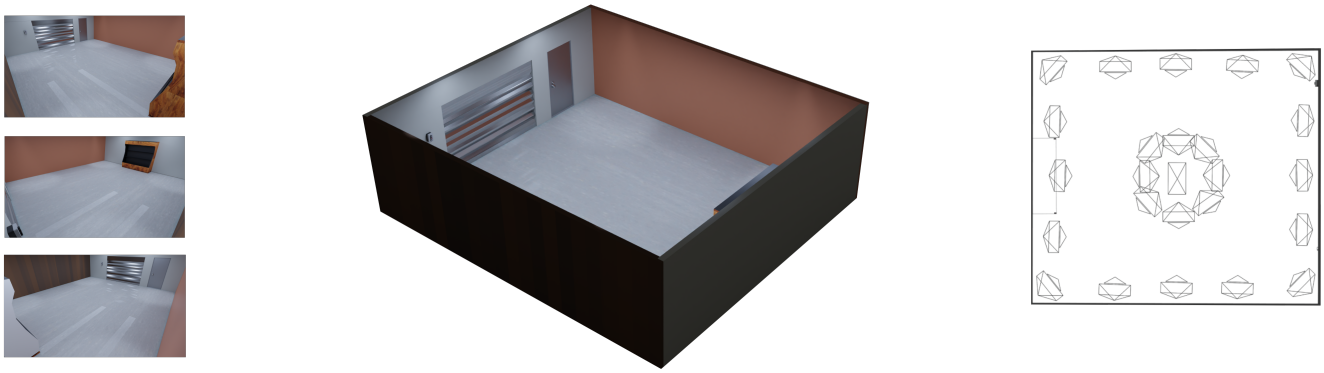

Fig. 3. Small room scene: array composed of 25 cameras mounted on the ceiling of a 72m^2 room. Left: image examples; Middle: 3D model of the room; Right: top-view of the room with camera locations.

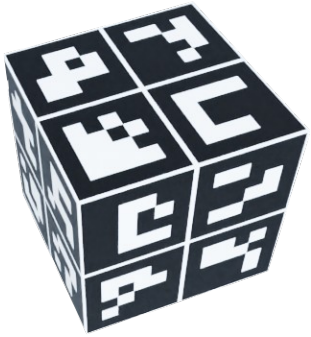

Fig. 4. Cube with side 0.575m covered in 24 ArUco markers with side 0.276m, used as the dynamic object in the bipartite pose graph.

IV. CONCLUSIONS

In this paper, we presented a new approach for pose estimation in large camera networks. Leveraging the well-established duality-based MLE methodology commonly employed in the PGO literature, we laid out a new formulation of the problem as a bipartite pose graph, comprising both camera and object nodes. This allowed us to derive an efficient method to carry out camera pose optimization, with accuracy improving with additional object poses, but without compromising computational efficiency. In addition, we put forward a new 3D dataset of indoor scenes, for the purpose of camera calibration with and without objects. We believe that this dataset can serve as a benchmark for future works on object and camera pose estimation. Our algorithm stands as a competitive baseline for the latter.

ACKNOWLEDGEMENT

This work was supported by LARSyS funding (DOI: 10.54499/LA/P/0083/2020, 10.54499/UIDP/50009/2020, and 10.54499/UIDB/50009/2020], through Fundação para a Ciência e a Tecnologia. M Marques and J Costeira were also supported by the SmartRetail project [PRR - C645440011-00000062], through IAPMEI - Agência para a Competitividade e Inovação.

REFERENCES

[1] J. L. Schönberger and J.-M. Frahm, "Structure-from-Motion Revisited," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A General Framework for Graph Optimization," in *IEEE International Conference on Robotics and Automation,*. IEEE, 2011, pp. 3607–3613.

[3] L. Carlone, D. M. Rosen, G. Calafiore, J. J. Leonard, and F. Dellaert, "Lagrangian Duality in 3D SLAM: Verification Techniques and Optimal Solutions," Tech. Rep.

[4] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert, "Initialization techniques for 3D SLAM: A survey on rotation estimation and its use in pose graph optimization," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June. Institute of Electrical and Electronics Engineers Inc., 6 2015, pp. 4597–4604.

[5] F. Dellaert, "Factor Graphs and GTSAM: A Hands-on Introduction," Tech. Rep., 2012.

[6] D. Martinec and T. Pajdla, "Robust Rotation and Translation Estimation in Multiview Reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[7] F. Arrigoni, B. Rossi, and A. Fusiello, "Spectral synchronization of multiple views in SE(3)," *SIAM Journal on Imaging Sciences*, vol. 9, no. 4, pp. 1963–1990, 2016.

[8] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard, "SE-Sync: A Certifiably Correct Algorithm for Synchronization over the Special Euclidean Group," 12 2016.

[9] R. Hartley, J. Trumpf, Y. Dai, and H. Li, "Rotation averaging," *International Journal of Computer Vision*, vol. 103, no. 3, pp. 267–305, 7 2013.

[10] Álvaro Parra, S.-F. Chng, T.-J. Chin, A. Eriksson, and I. Reid, "Rotation Coordinate Descent for Fast Globally Optimal Rotation Averaging," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[11] R. Tron and R. Vidal, "Distributed image-based 3-D localization of camera sensor networks," in *IEEE Conference on Decision and Control*. Institute of Electrical and Electronics Engineers Inc., 2009, pp. 901–908.

[12] ——, "Distributed 3-D localization of camera sensor networks from 2-D image measurements," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3325–3340, 2014.

[13] A. Eriksson, C. Olsson, F. Kahl, and T.-J. Chin, "Rotation Averaging and Strong Duality," in *IEEE Conference on Computer Vision And Pattern Recognition*, 2018.

[14] F. Dellaert, D. M. Rosen, J. Wu, R. Mahony, and L. Carlone, "Shonan Rotation Averaging: Global Optimality by Surfing SO(p) n," in *European Conference on Computer Vision*.

[15] G. Moreira, M. Marques, and J. Paulo Costeira, "Rotation Averaging in a Split Second: A Primal-Dual Method and a Closed-Form for Cycle Graphs," in *IEEE/CVF International Conference on Computer Vision*, 2021.

[16] ——, "Fast Pose Graph Optimization via Krylov-Schur and Cholesky Factorization," in *Winter Conference on Applications of Computer Vision*, 2021.
[17] K. J. Doherty, D. M. Rosen, and J. J. Leonard, "Performance Guarantees for Spectral Initialization in Rotation Averaging and Pose-Graph SLAM," in *Proceedings - IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers Inc., 2022, pp. 5608–5614.
[18] N. Boumal, A. Singer, P. A. Absil, and V. D. Blondel, "Cramér-Rao bounds for synchronization of rotations," 11 2012.
[19] N. Boumal, "Nonconvex phase synchronization," *SIAM Journal on Optimization*, vol. 26, no. 4, pp. 2355–2377, 2016.