Averaging Essential and Fundamental Matrices in Collinear Camera Settings

Amnon Geifman* Yoni Kasten* Meirav Galun Ronen Basri
Weizmann Institute of Science
{amnon.geifman,yoni.kasten,meirav.galun,ronen.basri}@weizmann.ac.il

Abstract

Global methods to Structure from Motion have gained popularity in recent years. A significant drawback of global methods is their sensitivity to collinear camera settings. In this paper, we introduce an analysis and algorithms for averaging bifocal tensors (essential or fundamental matrices) when either subsets or all of the camera centers are collinear. We provide a complete spectral characterization of bifocal tensors in collinear scenarios and further propose two averaging algorithms. The first algorithm uses rank constrained minimization to recover camera matrices in fully collinear settings. The second algorithm enriches the set of possibly mixed collinear and non-collinear cameras with additional, "virtual cameras," which are placed in general position, enabling the application of existing averaging methods to the enriched set of bifocal tensors. Our algorithms are shown to achieve state of the art results on various benchmarks that include autonomous car datasets and unordered image collections in both calibrated and unclibrated settings.

1. Introduction

Global approaches to Structure from Motion (SfM) use bifocal tensors (essential or fundamental matrices) between pairs of images to recover camera parameters in multiview settings. These methods have gained popularity in recent years due to their high accuracy and improved efficiency. In contrast to incremental methods, which recover camera parameters for one image at a time and thus involve repeated application of bundle adjustment (BA) for each handled image, global algorithms apply BA only once, considerably reducing execution time. Existing global algorithms largely proceed in two steps, applying rotation averaging followed by translation averaging. Recent algorithms further improve accuracy by directly averaging essential and fundamental matrices in one step [13, 14].

*Equal contributors

A significant drawback of global methods is their sensitivity to collinear camera settings. When all camera centers in a scene lie along a line, bifocal tensors do not determine camera locations along this line, and point matches across three or more images must be utilized. Moreover, with only bifocal tensors, subsets of collinear cameras can lead to reconstructions of scene parts that are attached nonrigidly. Finally, the averaging algorithms of [13, 14] critically base their recovery on triplet sub-collections of images whose cameras must lie in general position. This severely limits the applicability of these algorithms, requiring in many cases to remove many images from the input datasets. Handling collinear camera settings is critical to many SLAM applications, including autonomous driving [8].

This paper introduces an analysis and novel solutions to 3D reconstruction problems involving cameras with collinear centers in the context of bifocal tensor averaging. We note that to date this problem has been addressed only in the context of translation averaging [5, 11, 34]. We introduce a complete algebraic characterization of bifocal tensors in collinear scenarios, providing both necessary and sufficient conditions that bifocal tensors can be realized by cameras with collinear centers. Our analysis complements the conditions derived for cameras in general position in [13, 14] and the partial conditions for collinear settings derived in [25]. Specifically, adopting the definitions of n-view bifocal matrices introduced in those papers, we provide a full characterization in terms of spectral decomposition and rank patterns of these matrices.

We build upon this characterization to design state of the art algorithms for global SfM that are applicable in both calibrated and uncalibrated settings. We first introduce a method that, given possibly erroneous bifocal tensors, enforces our spectral constraints. This algorithm is suitable for image collections captured in fully collinear settings.

We subsequently present a second algorithm for bifocal tensor averaging that can incorporate both collinear cameras and cameras in general position. This algorithm is based on the following novel observation. Given a point match across three views, it is possible to define a *virtual* camera centered at the unknown 3D location of this point and subse-

quently construct bifocal tensors relating this virtual camera with any of the three cameras corresponding to these views. Choosing this point so that its projections lie away from the epipoles ensures that the center of the virtual camera is non-collinear with the real cameras. We can therefore augment the set of bifocal tensors with the newly constructed matrices and then feed them to standard bifocal tensors averaging schemes, allowing us to obtain solutions in both fully and partly collinear camera settings.

We demonstrate state of the art results in various applications, and specifically improve over the recent results of [13, 14] by allowing to incorporate collinear camera triplets in the optimization process. We evaluate our proposed algorithms on four benchmarks: autonomous car datasets [8] and unordered collections of images [34, 19, 33] in both calibrated and uncalibrated settings.

2. Related work

Incremental approaches for calibrated [24, 15, 27, 1, 35] and uncalibrated SfM settings [17, 22] use two images to obtain an initial reconstruction and then incrementally use camera resectioning methods [7, 12, 21, 36, 10, 3], adding one image at a time to expand the reconstruction. Bundle Adjustment [31] is performed with every additional image to prevent error drift of camera parameters, rendering this process computationally demanding.

Global approaches to SfM use collections of bifocal tensors to simultaneously solve for the parameters of all cameras, and subsequently perform a single round of Bundle Adjustment. Most existing global approaches for calibrated settings first extract the pairwise rotations from the essential matrices, then perform rotation averaging [2, 18, 32, 9, 4], and finally solve for camera locations [2, 34, 20, 11, 5, 6]. Kasten et al. [13] introduced a method for averaging essential matrices, allowing for solving for camera location and orientation in a single optimization framework. In uncalibrated settings, Sweeney et al. [29] presented a method that first improves the measured fundamental matrices, and then after self-calibration, apply rotation averaging followed by translation averaging. More recently, [14] introduced an averaging algorithm for fundamental matrices that yields a unique projective reconstruction.

Global methods rely on collections of bifocal tensors, but those cannot determine the magnitudes of translation in collinear camera settings, and 3D points recovered from point tracks in three or more images must be used. This problem has been addressed in the context of translation averaging. Jiang et al. [11] recover translation magnitudes in collinear triplets of cameras by registering the depth of 3D points triangulated independently from each pair of cameras. Akin to our second algorithm, Wilson et al. [34] use unknown 3D points as additional (but not collinear) cameras in translation averaging. Cui et al. [5] extend [11] to cope with points tracks.

A number of papers analyze the solvability of SfM by investigating its corresponding viewing graph, in which each node represents a camera and edges represent available fundamental matrices [16, 20, 23, 29, 30]. These approaches, however, assume that the cameras are in general position and hence do not determine which viewing graphs are solvable in (possibly partly) collinear settings.

3. Characterization of collinear settings

Let $I_1, ..., I_n$ denote a collection of n images of a static scene captured respectively by cameras $P_1, ..., P_n$. Each camera P_i is represented by a 3×4 matrix $P_i = K_i R_i^T [I, -t_i]$ where K_i is a 3×3 calibration matrix, $\mathbf{t_i} \in \mathbb{R}^\mathbf{3}$ and $R_i \in SO(3)$ denote the position and orientation of P_i, respectively, in some global coordinate system. We further denote $V_i = K_i^{-T} R_i^T$, therefore the camera projection matrix can be expressed as

$$P_i = V_i^{-T} [I, -\mathbf{t}_i] \tag{1}$$

Consequently, let $\mathbf{X} = (X, Y, Z)^T$ be a scene point in the global coordinate system. Its projection onto I_i is given by $\mathbf{x}_i = \mathbf{X}_i / Z_i$, where $\mathbf{X}_i = (X_i, Y_i, Z_i)^T = K_i R_i^T (\mathbf{X} - \mathbf{t}_i)$.

We denote the fundamental matrix and the essential matrix between images I_i and I_j by F_{ij} and E_{ij} respectively. It was shown in [2] that E_{ij} and F_{ij} can be written a

$$E_{ij} = R_i^T (T_i - T_j) R_j \tag{2}$$

$$F_{ij} = K_i^{-T} E_{ij} K_j^{-1} = V_i (T_i - T_j) V_j^T \tag{3}$$

where $T_i = [\mathbf{t}_i]_\times$.

Recently, [13, 14] established a set of algebraic constraints characterizing the consistency of bifocal tensors for cameras whose center lie in general position. In this paper we complement these characterizations by handling collinear camera centers. We first repeat the following definitions made in [13, 14]. Denote by $\mathbb{S}^{3n}$ the set of all the $3n \times 3n$ symmetric matrices.

Definition 1. *A matrix $F \in \mathbb{S}^{3n}$, whose 3×3 blocks are denoted by F_{ij}, is called an* **n-view fundamental matrix** *if $\forall i \neq j \in [n]$, rank$(F_{ij}) = 2$ and $F_{ii} = 0$. We denote the set of all such matrices by $\mathcal{F}$.*

Definition 2. *An n-view fundamental matrix F is called* **consistent** *if there exist camera matrices $P_1, ..., P_n$ of the form $P_i = V_i^{-T} [I, \mathbf{t}_i]$ such that $F_{ij} = V_i ([\mathbf{t}_i]_\times - [\mathbf{t}_j]_\times) V_j^T$.*

Definition 3. *A matrix $E \in \mathbb{S}^{3n}$, whose 3×3 blocks are denoted by E_{ij}, is called an* **n-view essential matrix** *if $\forall i \neq j$, rank$(E_{ij}) = 2$, the two singular values of E_{ij} are equal, and $E_{ii} = 0$. We denote the set of all such matrices by $\mathcal{E}$.*

Definition 4. *An n-view essential matrix E is called* **consistent** *if there exist n rotation matrices $\{R_i\}_{i=1}^n$ and n vectors $\{\mathbf{t}_i\}_{i=1}^n$ such that $E_{ij} = R_i^T([\mathbf{t}_i]_\times - [\mathbf{t}_j]_\times)R_j$.*

We next derive necessary and sufficient conditions for the consistency of essential and fundamental matrices in collinear camera settings.

Theorem 1. *Let $E \in \mathcal{E}$. Then, E is consistent and can be realized by cameras with collinear centers if and only if E satisfies the following two conditions:*

1. *The eigenvalues of E are $\lambda, \lambda, -\lambda, -\lambda$, where $\lambda > 0$.*
2. *The corresponding eigenvectors, $X, Y \in \mathbb{R}^{3n\times 2}$, are such that each 3×2 sub-block, V_i, of $\sqrt{0.5}(X + Y)$ satisfies $V_i^T V_i = \frac{1}{n} I_{2\times 2}$.*

Theorem 2. *Let $F \in \mathcal{F}$. Then, F is consistent and can be realized by cameras with collinear centers if and only if the following conditions hold*

1. *rank$(F) = 4$ and F has exactly 2 positive and 2 negative eigenvalues.*
2. *rank$(F_i) = 2$, where F_i denotes the i^{th} block-row of F, $i \in [n]$.*

The proofs of both theorems are given in the Appendix.

4. Method

In this section we present algorithms for bifocal tensor averaging when either subsets or all of the camera centers are collinear. We assume we are given images $I_1, ..., I_n$ along with a (possibly partial and erroneous) collection of measured bifocal tensors, denoted by $\{\hat{F}_{ij}\}$ if cameras are uncalibrated or$\{\hat{E}_{ij}\}$ if they are calibrated. Our aim is to find a *consistent* n-view bifocal matrix $F \in \mathbb{S}^{3n}$ (resp. $E \in \mathbb{S}^{3n}$) whose 3×3 blocks are as close as possible to the measured tensors.

Similar to [13, 14], our algorithms rely on constructing a triplet cover of the viewing graph that satisfies certain rigidity-like constraints. Specifically, let $G = (V, W)$ be a viewing graph whose vertices $v_1, ..., v_n \in V$ represent the n cameras, and edges $w_{ij} \in W$ represent pairs of images for which bifocal tensors are measured ($|W| \leq \binom{n}{2}$). The information captured in G is summarized in the n-view bifocal matrix $\hat{F} \in \mathbb{S}^{3n}$ (resp. $\hat{E}$).

A triplet cover is a *connected* dual graph $\bar{G}$, whose nodes represent (possibly a subset of) the 3-cliques in G and edges connect each two vertices whose corresponding 3-cliques in G share an edge (i.e., triplets that share two cameras). Configurations that are represented by such a connected dual graph satisfy a rigidity-like condition, according to which, as is proved in [14] for uncalibrated cameras in general position, if each 9×9 submatrix of F corresponding to a vertex in $\bar{G}$ is consistent then F determines the parameters of all cameras uniquely (up to a global projective transformation in $\mathcal{P}^4$). Moreover, enforcing the consistency of triplets is easier than that of larger sets of cameras since for camera triplets consistency is independent of the scale of the estimated bifocal tensors. Below we denote the number of vertices in $\bar{G}$ by $m \leq \binom{n}{3}$ and index them by $\tau(1), ..., \tau(m)$. We further denote by $E_{\tau(k)}$ (resp. $F_{\tau(k)}$) 9×9 submatrices of E (resp. F) corresponding to a triplet $\tau(k)$.

We next present two averaging algorithms. The first algorithm handles image collections taken with cameras whose centers are *all* near collinear. The second algorithm also allows for partial collinearity. We further show how both algorithms can be applied in both calibrated and uncalibrated settings. In each case we formulate the problem as a rank constraint optimization, which we solve using ADMM in a manner similar to [13, 14].

4.1. Fully collinear setups

Our first algorithm applies Thms. 1 and 2 to handle fully collinear setups.

4.1.1 Calibrated setting

Given the measurement matrix $\hat{E}$ and triplet cover $\bar{G}$ we seek to solve

$$\begin{aligned} \min_{E\in\mathcal{E}} \quad & \sum_{k=1}^{m} ||E_{\tau(k)} - \hat{E}_{\tau(k)}||_F^2 \qquad (4) \\ \text{s.t.} \quad & \text{rank}(E_{\tau(k)}) = 4 \\ & \lambda_1(E_{\tau(k)}) = -\lambda_4(E_{\tau(k)}),\ \lambda_2(E_{\tau(k)}) = -\lambda_3(E_{\tau(k)}) \end{aligned}$$

where $\lambda_i(.)$ denote the non-zero eigenvalues of a matrix, $i \in [4]$ and $k \in [m]$. We note that in (4) we excluded condition 2 of Thm. 1 to simplify the optimization. Our experiments converged in all cases to solutions that satisfy all the conditions of Thm. 1.

Recovery of camera parameters. Once we obtain an n-view essential matrix whose triplets are consistent, we proceed to determine the corresponding n camera matrices. There are two obstacles in this process. First, the obtained essential matrices do not determine the rotations uniquely, and secondly, in collinear settings essential matrices can only determine the direction of the line connecting camera centers, but not positions along this line. Due to the ambiguity of essential matrices three views give rise to eight possible rotation configurations, of which typically four give rise to cyclic consistent configurations (i.e., that satisfy $R_{12}R_{23}R_{31} = I$). To select the appropriate configuration we first use 2-view correspondences as in [10] to determine the pairwise rotations and then recover the absolute orientation of the three cameras, R_1, R_2, R_3, using the eigenvalues decomposition method of [2].

Next, we need to recover the absolute camera locations. Since our procedure enforces the conditions of Thm. 1 for each triplet and due to the rigidity-like structure of the triplet cover, all the recovered essential matrices agree on the direction of the line through the camera center. We therefore set $\mathbf{t}_1 = 0$, $\mathbf{t}_2 = -R_1\mathbf{t}_{12}$, and $\mathbf{t}_3 = \alpha\mathbf{t}_2 = -\alpha R_1\mathbf{t}_{12}$, where the relative translation $\mathbf{t}_{12}$ is extracted from $\hat{E}_{12}$ with magnitude 1 and sign determined using 2-view point correspondences. This yields the following camera matrices $P_1 = [R_1^T \ |0]$, $P_2 = [R_2^T \ |R_2^T R_1\mathbf{t}_{12}]$, and $P_3 = [R_3^T \ |\alpha R_3^T R_1\mathbf{t}_{12}]$. To determine α we must resort to 3-view correspondences. Let $\beta_i\mathbf{x}_i = P_i X$, $i \in [3]$, denote three projections of a 3D point X where β_i denote the projective depths of $\mathbf{x}_i$. As with the DLT algorithm [10], we use the first two equations to determine X and then α is determined from $P_3X \times \mathbf{x}_3 = 0$. Such an equation can be written for every 3-view correspondence, resulting in an over-constrained linear system of equations in α which we solve in least squares. We emphasize that the choice of rotations and α does not change E, and so it maintains its consistency and only resolves the ambiguity in reconstructing the underlying cameras. Finally, we use the method in [13], to traversing $\bar{G}$ and bring all the n cameras to a common Euclidean coordinate frame.

4.1.2 Uncalibrated setting

Given measurement matrix $\hat{F}$ and triplet cover $\bar{G}$ we solve

$$\min_{F\in\bar{\mathcal{F}}} \sum_{k=1}^{m} ||F_{\tau(k)} - \hat{F}_{\tau(k)}||_F^2 \qquad (5)$$
$$\text{s.t.} \quad \text{rank}(F_{\tau(k)}) = 4.$$

Here we denote by $\bar{\mathcal{F}}$ the set of n-view fundamental matrices where we relax the requirement that $\text{rank}(F_{ij}) = 2$. For simplicity of implementation we do not enforce the full set of constraints of Thm. 2. The solutions obtained in our experiments, however, always satisfied all of these conditions.

Recovery of camera parameters. Once we obtain an n-view fundamental matrix whose triplets are consistent we proceed to determine the corresponding n camera matrices. Here too, due to collinearity, reconstruction is not unique [10]. Formally, following [10, 16], given two fundamental matrices F_{12}, F_{23} there are four degrees of freedom in determine the three camera matrices that are compatible with F_{12} and F_{23}. The camera matrices can be expressed as $P_2 = [I \ |0]$, $P_1 = [[\mathbf{e}_{21}]_\times F_{12} \ |\mathbf{e}_{21}]$, and $P_3 = [[\mathbf{e}_{23}]_\times F_{23}^T \ |0] + \mathbf{e}_{23}\mathbf{a}^T]$, where $\mathbf{e}_{ij}$ is the null-space vector (epipole) of F_{ij}, and $\mathbf{a} \in \mathbb{R}^4$ can be set arbitrarily. For cameras in general position the remaining fundamental matrix F_{13} uniquely determines the entries of $\mathbf{a}$. When, however, the three cameras are collinear $\mathbf{a}$ is not determined by F_{13}. Similar to Sec. 4.1, we resolve $\mathbf{a}$ using 3-view correspondences. Using the first two view we recover the 3D point X and then obtain equations of the form $P_3X \times \mathbf{x}_3 = 0$ which provide two linear equations in $\mathbf{a}$ for every 3-view correspondence. In principle, two point correspondences suffice to determine $\mathbf{a}$, but for stability we incorporate all inlier 3-view correspondences.

This procedure is applied independently to each triplet of cameras, resulting in camera matrices defined up to a projective transformation. As with the calibrated case, the choice of $\mathbf{a}$ does not change F, and it only resolves the ambiguity in reconstructing the cameras. Finally, by traversing $\bar{G}$, as in [14], all the cameras are brought to a common projective coordinate frame.

4.2. Handling collinearity with virtual cameras

The algorithm presented in Sec. 4.1 handles datasets in which *all* cameras are (nearly) collinear. Many common datasets, however, contain both collinear cameras and cameras in general position. We next present a bifocal tensor averaging algorithm that can be applied to any such collection of cameras. Our algorithm extends the averaging algorithms of [13, 14] to these (partly) accidental settings. The main limitation of those previous algorithms is their reliance on constructing a triplet cover in which *every* triplet must include images captured by cameras in general position. This limits the applicability of the algorithm in datasets that include collinear camera sets and often results in discarding many of the input images. Below we propose a novel approach that overcomes this limitation.

Our approach is based on augmenting collinear triplets of cameras by constructing virtual cameras centered around 3D points corresponding to 3-view point matches that are not collinear with the real cameras. Let P_1, P_2, and P_3 be three cameras in a triplet. Recall (Eqs. (1)-(3)) that each camera can be parameterized by $P_i = [V_i^{-T}| - V_i^{-T}\mathbf{t}_i] \in \mathbb{R}^{3\times 4}$, $i \in [3]$, where in calibrated settings $V_i = R_i^T \in SO(3)$, and the associated bifocal tensors are then given by $F_{ij} = V_i[\mathbf{t}_i - \mathbf{t}_j]_\times V_j^T$, $i, j \in [3]$.

Let $X \in \mathbb{R}^3$ be a 3D point seen by the three cameras. We aim to construct bifocal tensors relating the virtual camera centered at X with the three real cameras P_1, P_2 and P_3. We further choose an "orientation" for the virtual camera that coincides with the orientations of one of the real cameras, say, $V_X = V_2$. The bifocal tensor F_{iX} for $i \in [3]$ can then be expressed as

$$\begin{aligned} F_{iX} &= V_i[\mathbf{t}_i - X]_\times V_2^T \\ &= \frac{1}{\det(V_i^{-1})}[V_i^{-T}(\mathbf{t}_i - X)]_\times V_i^{-T} V_2^T, \end{aligned}$$

where for the latter equality we use the identity $B^{-1}[\mathbf{a}]_\times = \frac{1}{\det(B)}[B^T\mathbf{a}]_\times B^T$. Let $\mathbf{x}_i = [x_i, y_i, 1]^T \in \mathbb{R}^3$ be the projection of X onto frame i. Then it holds that $s_i\mathbf{x}_i =$

$P_i[X^T, 1]^T = V_i^{-T}(X - \mathbf{t}_i)$, where s_i is the projective depth of X with respect to camera i. Therefore,

$$F_{iX} = \frac{-s_i}{\det(V_i^{-1})}[\mathbf{x}_i]_\times V_{i2}, \tag{6}$$

where $V_{i2} = V_i^{-T} V_2^T$. By construction, the matrix

$$\begin{bmatrix} 0 & F_{12} & F_{13} & F_{1X} \\ F_{12}^T & 0 & F_{23} & F_{2X} \\ F_{13}^T & F_{23}^T & 0 & F_{3X} \\ F_{1X}^T & F_{2X}^T & F_{3X}^T & 0 \end{bmatrix} \tag{7}$$

is a consistent 4-view bifocal matrix.

Note that F_{iX} in (6) can be estimated from the input images, since V_{i2} can be estimated from $\hat{F}_{12}$ and the scale $-s_i/\det(V_i^{-1})$ can be discarded. Specifically, in a calibrated setting we estimate $V_{i2} = R_{i2}$ from $\hat{E}_{i2}$. Two rotations are obtained, and we use pairwise correspondences to select the correct one. In an uncalibrated setting, following the recovery of cameras and the use of 3-view correspondences described in Sec. 4.1.2, we obtain that $V_{12} = V_1^{-T} V_2 = [\mathbf{e}_{21}]_\times F_{12}$, $V_{22} = I$, and $V_{32} = V_3^{-T} V_2 = [\mathbf{e}_{23}]_\times F_{23}^T + \mathbf{e}_{23}[a_1, a_2, a_3]$. Finally, X can be selected to be non-collinear with the centers of the three real cameras. Consequently, the estimated elements of (7) can be used to augment the viewing graph G and then used in the averaging algorithms of [13, 14], which are applicable and stable in general position scenarios. These algorithms use ADMM to solve constrained optimization problems, which for completeness we summarize below.

Averaging essential matrices [13]. Given a measurement matrix $\hat{E}$ and triplet cover $\bar{G}$, we solve

$$\begin{aligned} \min_{E \in \mathcal{E}} \quad & \sum_{k=1}^{m} ||E_{\tau(k)} - \hat{E}_{\tau(k)}||_F^2 \\ \text{s.t.} \quad & \text{rank}(E_{\tau(k)}) = 6 \\ & \lambda_i(E_{\tau(k)}) = -\lambda_{7-i}(E_{\tau(k)}), \;\; i = 1, 2, 3 \\ & X(E_{\tau(k)}) + Y(E_{\tau(k)}) \;\; \text{is block rotation,} \end{aligned} \tag{8}$$

where the columns of $X(E_{\tau(k)}), Y(E_{\tau(k)}) \in \mathbb{R}^{9\times 3}$ include the eigenvectors of $E_{\tau(k)}$ corresponding respectively to positive and negative eigenvalues.

Averaging fundamental matrices [14]. Given a measurement matrix $\hat{F}$ and triplet cover $\bar{G}$, we solve

$$\begin{aligned} \min_{F \in \mathcal{F}} \quad & \sum_{k=1}^{m} ||F_{\tau(k)} - \hat{F}_{\tau(k)}||_F^2 \\ \text{s.t.} \quad & \text{rank}(F_{\tau(k)}) = 6. \end{aligned} \tag{9}$$

5. Experiments

We evaluate our algorithms on several datasets, including nearly collinear video sequences taken from the KITTI Visual Odometry datasets [8] and unordered collections of calibrated [34] and uncalibrated [19, 33] images. Since our first algorithm (Sec. 4.1), which we denote by 'R4,' is applicable only to fully collinear settings we apply it only to the KITTI sequences. The second algorithm (Sec. 4.2), denoted 'VC,' is applied to all datasets. We compare our algorithms to several recent methods, including[13] , LUD [20] and 1DSFM [34] for the calibrated datasets and GPSFM [14] and PPSFM [17] for the uncalibrated ones. For the calibrated settings we compare the mean and median translation errors and for non-calibrated settings we compare the mean reprojection error.

5.1. Datasets

Driving Car Image Collections. The KITTI visual odometry [8] benchmark includes 11 video sequences captured by moving cars with ground truth camera positions and orientations. As is typical for driving, these sequences often contain stretches of near collinear motion. For our experiments we randomly selected for each of 11 datasets three near collinear subsequences (identified by applying PCA to the ground truth camera locations), each includes 100 frames. We then used each sequence to produce three collections of non-overlapping subsequences, each of length 5, 10 or 20 frames, yielding a total of 1155 sequences over 3300 frames.

Unordered Internet Photos. We further test our VC algorithm on calibrated unordered internet photo collections, collected by [34]. We note that the "ground truth" camera matrices for this dataset include an estimate obtained by an incremental method [26] . This dataset includes many outlier photographs, and consequently, in addition to maximizing accuracy, our goal is to maximize the number of cameras handled by the method. Additional datasets include uncalibrated photos [19, 33]. As "ground truth" these datasets include a list of inliers 2D projections of unknown 3D points by unknown cameras, allowing us to evaluate reconstruction accuracy via the mean reprojection error.

5.2. Constructing a triplet cover

Both our R4 and VC algorithms require a triplet cover graph $\bar{G}$ as input. We produce this cover by applying the following three steps.

Initial triplet cover. For our R4 algorithm, we initialize $\bar{G}$ simply using consecutive camera triplets $(i-1, i, i+1)$, $2 \le i < n-1$. For our VC algorithm we construct an initial cover using the heuristics of [13, 14], where for the Kitti datasets we do not filter collinear triplets.

Enrichment. For the internet photo collection datasets we next enrich the initial triplet covers as follows. For the calibrated dataset of [34], the initial cover is typically disconnected. Rather than using just the largest connected component, as in [13], we augment the set of triplets with collinear

ones that make the graph connected. To that end, we also keep the full cover $\bar{G}'$, a graph that includes all the 3-cliques in $\bar{G}$ (so the vertices of $\bar{G}'$ form a superset of the vertices of $\bar{G}$). We then iteratively select pairs of nodes in $\bar{G}$ from different connected components and use shortest path (measured by the number of edges) to connect them in $\bar{G}'$.

For the uncalibrated datasets [19, 33] the initial triplet cover produced with [14] forms graphs with single connected components. Although the triplets in these graphs pass the non-collinearlity test applied in [14], we add a virtual camera to each triplet, in order to improve the stability of the averaging algorithm.

Adding virtual cameras and filtering. Next, we identify collinear triplets (using the collinearity measures of [13, 14]) and for each such triplet produce a virtual camera. To that end, we consider the set of 3-view correspondences that are not collinear with the 3 camera centers (we avoid points whose too close to the epipole), and select the match that minimizes the sum of symmetric epipolar distances in all three images. We then produce the three bifocal tensors relating the virtual camera to the triplet. As this results in a cover that is not minimal (for example, only two of the three triplets that involve a virtual camera are needed to produce a valid cover), we further remove superfluous triplets from cover as in [13, 14].

5.3. Results

Tables 1-7 show the accuracy and execution times obtained with our algorithms on all datasets. Our algorithms are further compared to state of the art methods including essential matrix averaging [13], LUD [20], and 1DSFM [34] for calibrated images and GPSFM [14] and PPSFM [17] for uncalibrated ones. Tables 1-4 show the results of applying our algorithms on the nearly collinear Kitti sequences under calibrated and uncalibrated settings. These experiments demonstrate that our pipelines are faster than the other methods except [13, 14], which completely fail to reconstruct the scene due to their sensitivity to collinearity. Also, our algorithms were more accurate than the other methods in most of the runs.

Tables 5-6 show the results of applying our VC algorithm to the calibrated internet photo collection [34]. The results demonstrate the benefit of adding virtual cameras for collinear triplets, which has led to increasing the number of reconstructed cameras compared to [13] while maintaining comparable accuracy. Our method runs as fast as LUD and is much faster than 1DSFM.

Table 7 shows reprojection error and execution time of our VC algorithm on the uncalibrated internet photo benchmark. Compared to GPSFM and PPSFM, our reconstruction is more accurate than the other methods in 6 out of the 8 datasets and is on par with GPSFM on the remaining 17 datasets. Our method is slower than GPSFM due to the additional virtual cameras. Yet it is faster than PPSFM in most runs and more accurate in all runs. Additional results are provided in the supplementary material.

Technical details. We ran our experiments on an Intel(R)-i7 3.20GHz with Windows. For BA we used the Theia SfM library [28], which we run on a Linux Intel(R) Xeon(R) CPU@2.30GHz with 16 cores. Camera position results for [13, 20, 34] in Tables 5-6 are taken from these papers.

Table 1. KITTI, calibrated: Mean position error in meters before BA.

DS	5 Cameras					10 Cameras					20 Cameras				
	VC	R4	[13]	[20]	[34]	VP	R4	[13]	[20]	[34]	VP	R4	[13]	[20]	[34]
00	**0.02**	0.06	0.36	0.05	299.50	**0.04**	0.10	1.34	0.08	0.94	0.13	0.35	2.69	**0.11**	1.89
01	**0.73**	1.41	2.36	1.16	1.88	**1.09**	3.22	5.47	1.86	4.26	3.57	5.21	11.97	**2.57**	5.87
02	**0.03**	0.04	0.65	0.11	0.89	**0.09**	0.12	1.89	0.13	1.68	0.34	**0.20**	4.20	0.29	1.97
03	**0.07**	0.23	0.28	0.09	0.55	0.27	0.41	0.64	**0.08**	0.97	0.45	1.52	2.33	**0.12**	1.83
04	**0.09**	0.26	0.90	0.18	0.84	**0.07**	0.34	2.65	0.18	1.74	**0.15**	0.60	6.22	0.39	4.21
05	**0.02**	0.06	0.49	0.07	1166.56	**0.03**	0.10	1.37	0.13	30.14	**0.09**	0.24	3.24	0.19	2.02
06	**0.03**	0.16	0.94	0.14	1.37	**0.10**	0.28	2.43	0.17	1.52	**0.21**	0.76	5.73	0.39	3.89
07	**0.02**	0.08	0.30	0.05	7.95	0.04	0.18	0.80	0.06	442.08	**0.09**	0.34	1.74	0.10	1.79
08	**0.02**	0.04	0.40	0.06	0.62	**0.04**	**0.16**	0.92	0.07	49.13	**0.11**	0.25	2.37	0.15	2.33
09	**0.02**	0.15	0.62	0.12	21.07	**0.06**	0.24	1.96	0.10	0.96	0.18	0.53	3.93	**0.14**	2.97
10	**0.02**	0.04	0.61	0.07	0.86	**0.05**	0.14	1.65	0.11	1.42	**0.15**	0.28	3.01	0.17	2.54

Table 2. KITTI, calibrated: average execution time in seconds.

5 Cameras				10 Cameras				20 Cameras			
VP	R4	[13]	[20]	VP	R4	[13]	[20]	VP	R4	[13]	[20]
0.42	0.47	**0.22**	0.75	1.21	1.07	**0.58**	2.34	3.06	2.15	**1.47**	6.51

Table 3. KITTI, uncalibrated: Mean reprojection error in pixels after BA, averaged per dataset.

	5 Cameras				10 Cameras				20 Cameras			
	VC	R4	[14]	[17]	VP	R4	[14]	[17]	VP	R4	[14]	[17]
00	**0.12**	**0.12**	0.40	0.13	**0.15**	**0.15**	4.63	0.16	**0.16**	**0.16**	8.71	0.18
01	0.90	0.85	2.43	**0.22**	6.09	1.49	7.09	**0.32**	5.84	6.77	16.45	**0.41**
02	**0.12**	**0.12**	2.84	0.13	**0.15**	**0.15**	7.58	0.16	**0.16**	**0.16**	14.20	0.19
03	**0.14**	**0.14**	0.91	0.15	**0.17**	**0.17**	0.84	0.21	**0.19**	0.21	4.60	0.31
04	**0.12**	**0.12**	0.69	0.14	**0.15**	**0.15**	11.68	0.18	**0.18**	**0.18**	39.80	0.22
05	**0.13**	**0.13**	1.72	0.15	**0.16**	**0.16**	9.89	0.19	**0.18**	**0.18**	20.61	0.27
06	**0.12**	**0.12**	1.92	0.13	**0.15**	**0.15**	18.74	0.17	**0.17**	**0.17**	42.83	0.21
07	0.84	**0.16**	1.02	**0.16**	**0.18**	0.23	3.80	0.22	**0.21**	3.15	11.28	0.37
08	**0.13**	**0.13**	0.90	0.14	**0.16**	0.18	4.16	0.18	**0.18**	0.21	9.42	0.22
09	**0.12**	**0.12**	1.44	0.13	**0.14**	**0.14**	8.78	0.16	**0.16**	**0.16**	12.62	0.19
10	**0.12**	**0.12**	1.91	0.13	**0.15**	**0.15**	4.55	0.17	**0.17**	0.20	10.51	0.20

Table 4. KITTI, uncalibrated: average execution time in seconds

5 Cameras				10 Cameras				20 Cameras			
VP	R4	[14]	[17]	VP	R4	[14]	[17]	VP	R4	[14]	[17]
0.66	1.24	0.67	2.04	1.70	2.56	**1.68**	5.60	4.37	5.59	**3.89**	11.63

Acknowledgment This research is supported in part by the U.S.-Israel Binational Science Foundation, grant number 2018680 and by the Minerva Foundation with funding from the German Federal Ministry for Education and Research .

Appendix

Our proofs rely on lemmas provided in the supplementary material.

A. Proof of Theorem 1

Proof. $\Rightarrow$: Assume $E \in \mathcal{E}$ is consistent and realized by cameras with collinear centers. Using Thm. 2 in [13] we have

i. E can be formulated as $E = A + A^T$ where $A = UV^T$ and $U, V \in \mathbb{R}^{3n\times 3}$

$$U = \begin{bmatrix} \alpha_1 R_1^T \\ \vdots \\ \alpha_n R_n^T \end{bmatrix} T, \quad V = \begin{bmatrix} R_1^T \\ \vdots \\ R_n^T \end{bmatrix} \tag{10}$$

Table 5. Unordered internet photos, calibrated: Mean ($\bar{x}$) and median ($\tilde{x}$) camera position error in meters before and after bundle adjustment.

		Our Method					[13]					LUD [20]					1DSFM [34]			
Data set	N_c	$\bar{x}$	$\tilde{x}$	$\bar{x}_{BA}$	$\tilde{x}_{BA}$	N_r	$\bar{x}$	$\tilde{x}$	$\bar{x}_{BA}$	$\tilde{x}_{BA}$	N_r	$\bar{x}$	$\tilde{x}$	$\bar{x}_{BA}$	$\tilde{x}_{BA}$	N_r	$\bar{x}$	$\bar{x}_{BA}$	$\tilde{x}_{BA}$	N_r
Vienna Cathedral	836	14.9	4.9	11.1	2.0	715	9.6	4.2	5.4	1.2	674	10	5.4	10	4.4	750	6.6	2e4	0.5	757
Piazza del Popolo	328	6.7	3.1	3.2	0.8	280	7.2	3.5	2.5	0.8	275	5	1.5	4	1.0	305	3.1	200	2.6	303
NYC Library	332	4.2	2.7	2.3	0.8	281	3.3	2.2	1.1	0.47	277	6	2.0	7	1.4	320	2.5	20	0.4	292
Alamo	577	2.6	1.1	1.4	0.3	502	2.5	1.2	0.8	0.35	482	2	0.4	2	0.3	547	1.1	2e7	0.3	521
Yorkminster	437	11.6	3.1	9.4	1.0	367	5.6	2.7	1.9	0.8	341	5	2.7	4	1.3	404	3.4	500	0.2	395
Montreal ND	450	2.0	1	1.2	0.7	433	1.9	1.0	0.6	0.4	416	1	0.5	1	0.4	435	2.5	1	0.9	425
Tower of London	572	11.2	4.6	6.9	1.4	422	11.6	5.0	4	1.0	414	20	4.7	10	3.3	425	11	40	0.4	414
Ellis Island	227	11.0	5.3	4.4	1.8	214	14.1	6.1	5.3	1.7	211	-	-	-	-	-	3.7	40	0.4	213
Notre Dame	553	1.7	0.7	0.5	0.2	536	1.8	0.8	0.4	0.2	529	0.8	0.3	0.7	0.2	536	10	7	2.1	500

Table 6. Unordered internet photos, calibrated: Execution time in seconds. T_{R+T} denote the time for motion averaging (either tensor averaging or rotation and translation). T_{BA} the time for bundle adjustment and T_{Tot} is the total running time of the method, including the additional time for building the triangle cover. Empty cells represent image collections not tested by the authors.

	Our Method			[13]			LUD [20]			1DSFM[34]		
Data set	T_{R+T}	T_{BA}	T_{Tot}	T_{R+T}	T_{BA}	T_{Tot}	T_{R+T}	T_{BA}	T_{Tot}	T_{R+T}	T_{BA}	T_{Tot}
Vienna Cathedral	145	262	930	68	293	566	787	208	1467	323	3611	3934
Piazza del Popolo	54	39	143	26	27	87	88	31	162	42	213	255
NYC Library	55	80	214	28	58	125	102	47	200	47	382	429
Alamo	96	115	509	47	155	327	385	133	750	152	646	798
Yorkminster	67	100	296	33	116	207	103	148	297	71	955	1026
Montreal ND	80	216	626	41	170	494	271	167	553	93	1043	1136
Tower of London	89	132	280	41	120	241	88	86	228	61	750	811
Ellis Island	40	44	170	21	53	140	-	-	-	29	276	305
Notre Dame	100	419	1070	52	277	720	707	126	1047	205	2139	2344

Table 7. Unordered internet photos, uncalibrated: Mean reprojection error and execution times. m and n respectively denote the number of 3D points and cameras.

Dataset	m	n	Error (pixels)			Time (sec.)		
			VC	[14]	[17]	VC	[14]	[17]
Dino 4983	4983	36	0.43	**0.42**	0.47	15.75	**4.65**	13.00
Folke Filbyter	21150	40	**0.26**	0.82	0.31	14.30	**6.70**	102.77
Cherub	72784	65	0.75	**0.74**	0.81	48.52	**27.30**	101.64
Toronto University	7087	77	**0.24**	0.54	0.26	30.47	**26.59**	91.26
Sri Thendayuthapani	88849	98	**0.31**	0.51	0.33	**219.11**	220.25	325.58
Tsar Nikolai I	37857	98	**0.29**	0.32	0.31	89.93	**70.79**	101.01
Smolny Cathedral	51115	131	**0.46**	0.48	0.50	303.62	**210.75**	263.60
Skansen Kronan	28371	131	**0.41**	0.44	0.44	118.60	**83.43**	161.81

with $R_i \in SO(3)$, $\mathbf{t}_i = \alpha_i \mathbf{t}$, $i \in [n]$, $\mathbf{t} \in \mathbb{R}^3$, $T = [\mathbf{t}]_\times$, such that $\sum_{i=1}^n \mathbf{t}_i = \sum_{i=1}^n \alpha_i \mathbf{t} = 0$.

ii. Each column of U is orthogonal to each column of V, i.e., $V^T U = 0_{3\times 3}$.

To prove condition 1, we first examine the eigenvalues of

$$A^T A = V U^T U V^T. \tag{11}$$

Due to the orthogonality of the R_i's

$$U^T U = \left(\sum_{i=1}^n \alpha_i^2 \right) T^T T. \tag{12}$$

T is 3×3 skew-symmetric, therefore it has 2 identical singular values (while the third one is zero). Consequently, the (symmetric) matrix $U^T U$ has two identical eigenvalues, with eigenvalue decomposition of the form $U^T U = P \tilde{\Lambda} P^T$ with $\tilde{\Lambda} = \text{diag}[\tilde{\lambda}, \tilde{\lambda}, 0]$ and $P \in SO(3)$. Therefore $\text{rank}(U) = 2$ and, using (11),

$$A^T A = V P \tilde{\Lambda} P^T V^T. \tag{13}$$

Note that $(VP)^T (VP) = P^T V^T V P = n I_{3\times 3}$, therefore if we scale V by $1/\sqrt{n}$ and scale $\tilde{\lambda}$ by n (13) becomes the eigen-decomposition of $A^T A$. Thus, $A^T A$ has two equal eigenvalues, and since $\lambda(A^T A) = (\sigma(A))^2$ it follows that A has two equal singular values (and $\text{rank}(A) = 2$). Consequently, the full rank factorization of A is of the form $A = \tilde{U} \tilde{V}^T$, where $\tilde{U}, \tilde{V} \in \mathbb{R}^{3n\times 2}$.

Next, we construct a full-rank factorization of A based on the columns of $U, V \in \mathbb{R}^{3n\times 3}$. Based on the observation that $\text{rank}(U) = 2$, without loss of generality we assume that the third column of U is a linear combination of the first two columns, i.e., $\mathbf{u}_3 = \alpha \mathbf{u}_1 + \beta \mathbf{u}_2$ for some $\alpha, \beta \in \mathbb{R}$. Now, A can be decomposed as $A = UV^T = \sum_{i=1}^3 \mathbf{u}_i \mathbf{v}_i^T = \mathbf{u}_1(\mathbf{v}_1^T + \alpha \mathbf{v}_3^T) + \mathbf{u}_2(\mathbf{v}_2^T + \beta \mathbf{v}_3^T)$. So a full-rank decomposition of A takes the form $A = \tilde{U}\tilde{V}^T$, with $\tilde{U} = [\mathbf{u}_1, \mathbf{u}_2]$ and $\tilde{V} = [\mathbf{v}_1 + \alpha \mathbf{v}_3, \mathbf{v}_2 + \beta \mathbf{v}_3]$. Note that $\text{span}(\tilde{U}) = \text{span}(U)$ and $\text{span}(\tilde{V}) \subseteq \text{span}(V)$, implying that the columns of $\tilde{U}$ are orthogonal to the columns of $\tilde{V}$.

Let $A = \hat{U} \begin{bmatrix} \sigma & 0 \\ 0 & \sigma \end{bmatrix} \hat{V}^T$ be the SVD decomposition of A, where $\hat{U}, \hat{V} \in \mathbb{R}^{3n\times 2}$, then

$$E = A + A^T = \begin{bmatrix} \hat{U} & \hat{V} \end{bmatrix} \Sigma \begin{bmatrix} \hat{V}^T \\ \hat{U}^T \end{bmatrix}, \tag{14}$$

where $\Sigma = \sigma I \in \mathbb{S}^4$. We next claim that (14) is the SVD decomposition of E. To that end, we need to show that $\text{span}(\hat{U})$ and $\text{span}(\hat{V})$ are orthogonal sub-spaces. Recall that A can be factorized into $A = \tilde{U}\tilde{V}^T$, such that $\text{span}(\tilde{U})$ and $\text{span}(\tilde{V})$ are orthogonal sub-spaces. Due to the ambiguity of the full-rank factorization, there exists an invertible matrix $B \in \mathbb{R}^{2\times 2}$ such that $\hat{U} = \tilde{U}B$ and $\hat{V} = \tilde{V}B^{-T}$ which implies that $\text{span}(\hat{U})$ is orthogonal to $\text{span}(\hat{V})$. This derivation implies that $\text{rank}(E) = 4$ with 4 identical singular values. Finally, using Lemma 1, due to the form of singular vectors the eigenvalues of E are of the form $\lambda, \lambda, -\lambda, -\lambda$, where $\lambda = \sigma > 0$.

Next, to prove the second condition, we first construct an SVD decomposition of A, using $A = UV^T$ in (i) and the thin SVD decomposition of $T = P \begin{bmatrix} \sigma & 0 \\ 0 & \sigma \end{bmatrix} Q^T$ with $P, Q \in \mathbb{R}^{3\times 2}$ such that $P^T P = Q^T Q = I_{2\times 2}$. Let $\hat{U} = \frac{1}{\sqrt{\sum \alpha_i^2}} \begin{bmatrix} \alpha_1 R_1^T \\ \vdots \\ \alpha_n R_n^T \end{bmatrix} P$ and $\hat{V} = \frac{1}{\sqrt{n}} \begin{bmatrix} R_1^T \\ \vdots \\ R_n^T \end{bmatrix} Q$ so that $\hat{U}^T \hat{U} =$

$I_{2\times2}$ and $\hat{V}^T\hat{V} = I_{2\times2}$. Then, using (10),

$$A = UV^T = \left(n\sum {\alpha_i}^2\right)^{\frac{1}{2}}\ \hat{U}\begin{bmatrix}\sigma & 0\\ 0 & \sigma\end{bmatrix}\hat{V}^T.$$

This thin SVD decomposition of A implies that the thin SVD decomposition of E is of the form (14). Using Lemma 1 we obtain that $\hat{V} = \sqrt{0.5}(X+Y)$ where X and Y include the eigenvectors of E, and the sub-blocks of $\hat{V}$ are orthogonal, since $\hat{V}_i = \frac{1}{\sqrt{n}}R_i^TQ$ satisfies $\hat{V}_i^T\hat{V}_i = \frac{1}{n}I_{2\times2}$, $i \in [n]$, implying that the second condition is satisfied.

$\Leftarrow$: We show that if $E \in \mathcal{E}$ satisfies conditions 1-2, E is consistent. Following condition 1, E is of rank 4 with spectral decomposition of the form as in Lemma 1, and therefore, due to this lemma, its SVD takes the form $E = \lambda\hat{U}\hat{V}^T + \lambda\hat{V}\hat{U}^T$, where $\hat{U}, \hat{V} \in \mathbb{R}^{3n\times2}$ are given by $\hat{U} = \sqrt{0.5}(X-Y)$ and $\hat{V} = \sqrt{0.5}(X+Y)$. Moreover, due to condition (2), every 3×2 block $\hat{V}_i$ satisfies $\hat{V}_i^T\hat{V}_i = \frac{1}{n}I_{2\times2}$.

Our next aim is to use $\hat{U}$ and $\hat{V}$ to define $U, V \in \mathbb{R}^{3n\times3}$ in a form that implies the consistency of E. For each $i \in [n]$ we construct a 3×3 block $\tilde{V}_i$ by adding a third column to $\hat{V}_i$ that is set to be the cross product of its two columns, scaled by $\sqrt{n}$. The sign of this vector is further selected such that $\det(\tilde{V}_i) > 0$. Consequently, $R_i = \sqrt{n}\tilde{V}_i$ is a rotation matrix, and we let

$$V = \left[R_1, ..., R_n\right]^T. \tag{15}$$

Additionally, we pad $\hat{U}$ with a third column of zeros, and then scale the obtained matrix by $\lambda/\sqrt{n}$, yielding $U \in \mathbb{R}^{3n\times3}$. It can be readily verified that $E = UV^T + VU^T$.

Next, since E is an n-view essential matrix, $E_{ii} = 0$, implying that $U_iV_i^T$ is skew symmetric. This means that there are two possible options: either $\text{rank}(U_iV_i^T) = 2$ or $\text{rank}(U_iV_i^T) = 0$. Since $\text{rank}(V_i) = 3$ this respectively implies that either $\text{rank}(U_i) = 2$ or $U_i = 0$. By Lemma 3, when $\text{rank}(U_i) = 2$ and $\text{rank}(V_i) = 3$ then

$$T_i = V_i^TU_i \tag{16}$$

is skew-symmetric. Moreover, since the third column of U_i is identically zero, then $T_i = \alpha_iT$ for some $\alpha_i \in \mathbb{R}$ and $T = \begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}$. This also includes the case that $U_i = 0$ by setting $\alpha_i = 0$. Combining (15) and (16), $U_i = V_i^{-T}T_i = \alpha_iR_i^TT$, implying that $U = \begin{bmatrix}\alpha_1R_1^T\\ \vdots\\ \alpha_nR_n^T\end{bmatrix}T$. Finally, $E = UV^T + VU^T$ and the forms of U and V imply that E is consistent and realized by cameras with collinear centers. □

B. Proof of Theorem 2

Proof. $\Rightarrow$: Assume $F \in \mathcal{F}$ is consistent and realized by cameras with collinear centers. Then, by definition and due to collinearity F can be formulated as $F = A + A^T$ where $A = UV^T$ and $U, V \in \mathbb{R}^{3n\times3}$,

$$U = \begin{bmatrix}\alpha_1V_1\\ \vdots\\ \alpha_nV_n\end{bmatrix}T, \quad V = \begin{bmatrix}V_1\\ \vdots\\ V_n\end{bmatrix},$$

where $V_i \in \mathbb{R}^{3\times3}$ are full rank, $T = [\mathbf{t}]_\times$ and $\mathbf{t} \in \mathbb{R}^3$.

Next, we show that $\text{rank}(F) = 4$. Using QR decomposition we can write $V_i = K_i^{-T}R_i^T$, where K_i is invertible and upper triangular and $R_i \in SO(3)$. This decomposition is unique due to the full rank of V_i. We can therefore write $F = K^TEK$, where the $3n\times3n$ matrix K is block diagonal with 3×3 blocks formed by $\{K_i^{-1}\}_{i=1}^n$ and thus has full rank. As a consequence $\text{rank}(F) = \text{rank}(E)$. By construction E has the same form as in Thm. 1, and therefore $\text{rank}(E) = 4$, implying also that $\text{rank}(F) = 4$.

Next, since $\text{rank}(F) = 4$ and $\text{rank}(U) = 2$ we have that $\text{rank}(A) = 2$ and we can express its full-rank decomposition as $A = \tilde{U}\tilde{V}^T$, where $\tilde{U}, \tilde{V} \in \mathbb{R}^{3n\times2}$ and $\text{rank}(\tilde{U}) = \text{rank}(\tilde{V}) = 2$. Using Lemma 2 F has two positive and two negative eigenvalues. This concludes the proof of the first condition.

The second condition is justified as follows. Due to collinearity, $\exists\mathbf{e} \in \mathbb{R}^3, \mathbf{e} \neq 0$, s.t. $F_{ji}\mathbf{e} = 0$ for some camera $\exists i \in [n]$ and for all other cameras $j \in [n]$, implying that $F_i^T\mathbf{e} = 0$. Consequently, for all $i \in [n]$, $\text{rank}(F_i) \leq 2$, implying that $\text{rank}(F_i) = 2$, since otherwise if $\text{rank}(F_i) < 2$ it contradicts the assumption that $\text{rank}(F_{ji}) = 2$, for all j.

$\Leftarrow$: Let $F \in \mathcal{F}$ then if F satisfies conditions 1 and 2. We prove that it is consistent and realized by cameras with collinear centers. Condition 1 and Lemmas 2 and 8 imply that F can be decomposed into $F = \hat{U}\hat{V}^T + \hat{V}\hat{U}^T$, where $\hat{U}, \hat{V} \in \mathbb{R}^{3n\times2}$ and $\text{rank}(\hat{U}) = \text{rank}(\hat{V}) = 2$ and all 3×2 blocks are full rank.

We next use $\hat{U}$ and $\hat{V}$ to define $U, V \in \mathbb{R}^{3n\times3}$ in a form that implies the consistency of F as follows. $U = \left[\hat{U}, 0\right]$ and for each $i \in [n]$ we construct a 3×3 block V_i by appending a third column to $\hat{V}_i$, setting it to be the cross product of its two columns. Since $F_{ii} = 0$ for all i, then $\hat{U}_i\hat{V}_i^T = U_iV_i^T$ is skew-symmetric, and by construction $\text{rank}(U_i) = 2$ and $\text{rank}(V_i) = 3$. Therefore by Lemma 3, $T_i = V_i^{-1}U_i$ is also skew-symmetric. Therefore, $T_i = [\mathbf{t}_i]_\times$, for some $\mathbf{t}_i \in \mathbb{R}^3$, implying that $F_{ij} = V_i(T_i - T_j)V_j^T$ and therefore that F is an n-view fundamental matrix.

Finally, collinearity follows from the second condition, which implies that $\exists\mathbf{e} \neq 0$ s.t. $F_i^T\mathbf{e} = 0$. This implies that $F_{ji}\mathbf{e} = 0$ for all $j \in [n]$, asserting that all n cameras are collinear.

□

References

[1] Sameer Agarwal, Noah Snavely, Ian Simon, Steven M Seitz, and Richard Szeliski. Building rome in a day. In *2009 IEEE 12th international conference on computer vision*, pages 72–79. IEEE, 2009. 2

[2] Mica Arie-Nachimson, Shahar Z Kovalsky, Ira Kemelmacher-Shlizerman, Amit Singer, and Ronen Basri. Global motion estimation from point matches. In *2012 Second Int. Conf. on 3D Imaging, Modeling, Processing, Visualization & Transmission*, pages 81–88. IEEE, 2012. 2, 3

[3] Federico Camposeco, Andrea Cohen, Marc Pollefeys, and Torsten Sattler. Hybrid camera pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 136–144, 2018. 2

[4] Avishek Chatterjee and Venu Madhav Govindu. Robust relative rotation averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):958–972, 2018. 2

[5] Zhaopeng Cui, Nianjuan Jiang, Chengzhou Tang, and Ping Tan. Linear global translation estimation with feature tracks. *ArXiv preprint, ArXiv:1503.01832*, 2015. 1, 2

[6] Zhaopeng Cui and Ping Tan. Global structure-from-motion by similarity averaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 864–872, 2015. 2

[7] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 25(8):930–943, 2003. 2

[8] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1, 2, 5

[9] Richard Hartley, Jochen Trumpf, Yuchao Dai, and Hongdong Li. Rotation averaging. *International Journal of Computer Vision*, 103(3):267–305, 2013. 2

[10] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003. 2, 3, 4

[11] Nianjuan Jiang, Zhaopeng Cui, and Ping Tan. A global linear method for camera pose registration. In *IEEE Int. Conf. on Computer Vision*, pages 481–488, 2013. 1, 2

[12] Yoni Kasten, Meirav Galun, and Ronen Basri. Resultant based incremental recovery of camera pose from pairwise matches. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1080–1088. IEEE, 2019. 2

[13] Yoni Kasten, Amnon Geifman, Meirav Galun, and Ronen Basri. Algebraic characterization of essential matrices and their averaging in multiview settings. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 1, 2, 3, 4, 5, 6, 7

[14] Yoni Kasten, Amnon Geifman, Meirav Galun, and Ronen Basri. Gpsfm: Global projective sfm using algebraic constraints on multi-view fundamental matrices. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 3, 4, 5, 6, 7

[15] Manfred Klopschitz, Arnold Irschara, Gerhard Reitmayr, and Dieter Schmalstieg. Robust incremental structure from motion. In *Proc. 3DPVT*, volume 2, pages 1–8, 2010. 2

[16] Noam Levi and Michael Werman. The viewing graph. In *Computer Vision and Pattern Recognition*, volume 1, pages I–I. IEEE, 2003. 2, 4

[17] Ludovic Magerand and Alessio Del Bue. Practical projective structure from motion (p2sfm). 2, 5, 6, 7

[18] Daniel Martinec and Tomas Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007. 2

[19] Carl Olsson and Olof Enqvist. Stable structure from motion for unordered image collections. In *Scandinavian Conf. on Image Analysis*, pages 524–535. Springer, 2011. 2, 5, 6

[20] Onur Ozyesil and Amit Singer. Robust camera location estimation by convex programming. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2674–2683, 2015. 2, 5, 6, 7

[21] Mikael Persson and Klas Nordberg. Lambda twist: an accurate fast robust perspective three point (p3p) solver. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–332, 2018. 2

[22] Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops, and Reinhard Koch. Visual modeling with a hand-held camera. *Int. Journal of Computer Vision*, 59(3):207–232, 2004. 2

[23] Alessandro Rudi, Matia Pizzoli, and Fiora Pirri. Linear solvability in the viewing graph. In *Asian Conf. on Computer Vision*, pages 369–381. Springer, 2010. 2

[24] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016. 2

[25] Soumyadip Sengupta, Tal Amir, Meirav Galun, Tom Goldstein, David W Jacobs, Amit Singer, and Ronen Basri. A new rank constraint on multi-view fundamental matrices, and its application to camera location recovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4798–4806, 2017. 1

[26] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM transactions on graphics (TOG)*, volume 25, pages 835–846. ACM, 2006. 5

[27] Noah Snavely, Steven M Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80(2):189–210, 2008. 2

[28] Chris Sweeney. Theia multiview geometry library: Tutorial & reference. http://theia-sfm.org. 6

[29] Chris Sweeney, Torsten Sattler, Tobias Hollerer, Matthew Turk, and Marc Pollefeys. Optimizing the viewing graph for structure-from-motion. In *IEEE Int. Conf. on Computer Vision*, pages 801–809, 2015. 2

[30] Matthew Trager, Brian Osserman, and Jean Ponce. On the solvability of viewing graphs. In *European Conf. on Computer Vision*, pages 335–350. Springer, Cham, 2018. 2

[31] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustmentâa modern synthesis. In *Int. Workshop on Vision Algorithms*, pages 298–372. Springer, 1999. 2

[32] Roberto Tron and René Vidal. Distributed image-based 3-d localization of camera sensor networks. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 901–908. IEEE, 2009. 2

[33] Oxford VGG. Multiview datasets. http://www.robots.ox.ac.uk/~vgg/data/. 2, 5, 6

[34] Kyle Wilson and Noah Snavely. Robust global translations with 1dsfm. In *European Conf. on Computer Vision*, pages 61–75. Springer, 2014. 1, 2, 5, 6, 7

[35] Changchang Wu. Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pages 127–134. IEEE, 2013. 2

[36] Enliang Zheng and Changchang Wu. Structure from motion using structure-less resection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2075–2083, 2015. 2