Real-Time Vision-Aided Localization and Navigation Based on Three-View Geometry

VADIM INDELMAN
PINI GURFIL
EHUD RIVLIN
Technion—Israel Institute of Technology

HECTOR ROTSTEIN
RAFAEL

A new method for vision-aided navigation based on three-view geometry is presented. The main goal of the proposed method is to provide position estimation in GPS-denied environments for vehicles equipped with a standard inertial navigation system (INS) and a single camera only, without using any a priori information. Images taken along the trajectory are stored and associated with partial navigation data. By using sets of three overlapping images and the concomitant navigation data, constraints relating the motion between the time instances of the three images are developed. These constraints include, in addition to the well-known epipolar constraints, a new constraint related to the three-view geometry of a general scene. The scale ambiguity, inherent to pure computer vision-based motion estimation techniques, is resolved by utilizing the navigation data attached to each image. The developed constraints are fused with an INS using an implicit extended Kalman filter. The new method reduces position errors in all axes to the levels present while the first two images were captured. Navigation errors in other parameters are also reduced, including velocity errors in all axes. Reduced computational resources are required compared with bundle adjustment and simultaneous localization and mapping (SLAM). The proposed method was experimentally validated using real navigation and imagery data. A statistical study based on simulated navigation and synthetic images is presented as well.

Manuscript received June 10, 2010; revised February 25, 2011; released for publication August 2, 2011.

IEEE Log No. T-AES/48/3/944013.

Refereeing of this contribution was handled by M. Braasch.

Authors' address: V. Indelman and P. Gurfil, Faculty of Aerospace Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel, E-mail: (ivadim@tx.technion.ac.il); E. Rivlin, Department of Computer Science, Technion–Israel Institute of Technology, Haifa 32000, Israel; H. Rotstein, RAFAEL–Advanced Defense Systems Limited, Israel.

I. INTRODUCTION

Inertial navigation systems (INS) develop navigation errors over time due to the imperfectness of the inertial sensors. Over the past few decades, many methods have been proposed for restraining or eliminating these errors, assuming various types of additional sensors and a priori information. The majority of modern navigation systems rely on the Global Positioning System (GPS) as the primary means for mitigating the inertial measurement errors. However, GPS might be unavailable or unreliable; this happens when operating indoors, under water, or on other planets. In these scenarios, vision-based methods constitute an attractive alternative for navigation aiding due to their relatively low cost and autonomous nature. Vision-aided navigation has indeed become an active research field alongside the rapid development of computational power.

The current work is concerned with vision-aided navigation for a vehicle equipped with a standard INS and a single camera only, a setup that has been studied in a number of previous works. Existing methods vary by the number of overlapping images and by the techniques used for fusing the imagery data with the navigation system. Two related issues that have drawn much attention are computational requirements and the ability to handle loops, i.e., how the navigation solution is updated when the platform revisits some area.

Given two overlapping images, it is only possible to determine camera rotation and up-to-scale translation [1]. Therefore, two-view based methods for navigation aiding [2–6] are incapable of eliminating the developing navigation errors in all states. With no additional information or sensors for resolving the scale ambiguity, such as range sensors or stereo vision, the vehicle states are only partially observable (e.g. for an airborne platform, position and velocity along the flight heading are unobservable [3, 4]).

Imagery information stemming from multiple images (≥ 3) with a common overlapping region enables to determine the camera motion up to a common scale [1]. Indeed, several multi-view methods for navigation aiding have been already proposed [7, 8]. In [7] features that are observed within multiple images and the platform pose are related using an augmented state vector: The state vector contains the current platform pose and the platform pose for each previously-captured image that has at least one feature that appears in the current image. Once a certain feature, observed in the previous images, is no longer present in the currently-captured image, all the stored information for this feature is used for estimating the platform parameters, and the pose entries that belong to these past images are discarded. However, should the same feature be reobserved at some later time instant (e.g. a loop in a trajectory),

the method will be unable to use the data for the feature's first appearance. It was later proposed [9] to cope with loops using bundle adjustment [1]. However, this process involves processing all the images that are part of the loop sequence, and therefore real-time performance is hardly possible. Furthermore, the method contains an intermediate phase of structure reconstruction. In [8] the authors use the rank condition on the multiple-view-matrix [10] for simultaneously recovering 3D motion and structure during a landing process of an unmanned aerial vehicle, assuming a planar ground scene is observed.

The augmented state technique is also found in other works. The state vector may be augmented with the platform pose each time an image is obtained [4, 9], or with the 3D coordinates of the observed features in these images, an approach commonly referred to as SLAM [11–17]. While SLAM methods are naturally capable of handling loops, they present increasing computational requirements at each update step, due to the state vector augmentation approach undertaken for consistently maintaining the cross-correlation between the vehicle and map. Thus, real-time performance over prolonged periods of time is difficult.

Another approach for coping with trajectories containing loops is to apply smoothing on the image sequence, thereby improving the consistency of the environment representation (e.g. mosaic image), and then update the navigation system [18, 19]. However, in [18] a stereo rig is used, allowing computation of depth, while in the current work a single camera is utilized. In [19] a quasi-planar scene is assumed and the motion estimation is based on a homography matrix refined during the smoothing process. Since a two-view-based method is applied, only up-to-scale translation can be estimated. Therefore, this method is incapable of updating the whole position state.

In contrast to SLAM, the approach proposed herein is based on decoupling the navigation-aiding process from the process of constructing a representation of the observed environment [2]. While the former should be performed in real time, the latter may not be required in real time. There are many applications that settle for obtaining the environment representation with some time delay, or alternatively, that prefer to obtain the raw imagery data and construct the environment representation on their own. Thus, the state vector is constant in size and is comprised only of the vehicle's parameters, while the captured imagery and some associated navigation data are stored and maintained outside the filter. In our previous work [2], this approach was used for vision-aided navigation based on two-view geometry. Here, we extend this framework to three-view geometry.

In the newly-proposed approach, each update step requires only three images with an overlapping area and some stored navigation data in order to estimate the vehicle's parameters, which can be performed in real time. The refinement of the environment representation, which is no longer coupled to the parameter estimation, may be performed in a background process by applying various algorithms (e.g. smoothing, bundle adjustment). Moreover, the newly-suggested approach eliminates the need for an intermediate phase of structure reconstruction.

To obtain real-time vision-aided localization and navigation, this work suggests a new formulation of the constraints related to the three-view geometry of a general scene. These constraints, developed following the rank condition approach [1, 10], combine imagery and navigation data at the time instances of the three images. The said constraints and the well-known trifocal tensor [1] are both constituted assuming a general three-view geometry. However, while the trifocal tensor utilizes only features that are observed from all the three images, the developed constraints may also be separately applied using features that are observed in each pair of images of the given three images. It should be noted that the trifocal tensor has been suggested for camera motion estimation [21, 22], and for localization of a robot and observed landmarks while performing a planar motion [23]. However, the trifocal tensor and in particular the constraints developed herein, have not been proposed so far for navigation aiding.

The constrains are fused with an INS using an implicit extended Kalman filter (IEKF), allowing estimation of the position vector by reducing the position errors to the levels present while the first two images were taken. The proposed method is also capable of estimating other states, such as the velocity. A sequential application of the new algorithm to the incoming image sequence and the stored navigation data yields reduced navigation errors. Loops in the trajectory are handled naturally, requiring a reduced computational load compared with state-of-the-art techniques for handling loops such as SLAM and bundle adjustment.

Consequently, the main contributions of this work are: 1) a new formulation of the constraints stemming from a general static scene captured by three views; 2) application of the developed three-view constraints for navigation aiding, thereby allowing to efficiently handle loops, and 3) reduced computational requirements compared with other methods mentioned above.

II. METHOD OVERVIEW

A simplified diagram of the proposed method for navigation aiding is given in Fig. 1. The vehicle is equipped with a standard INS and a camera (which

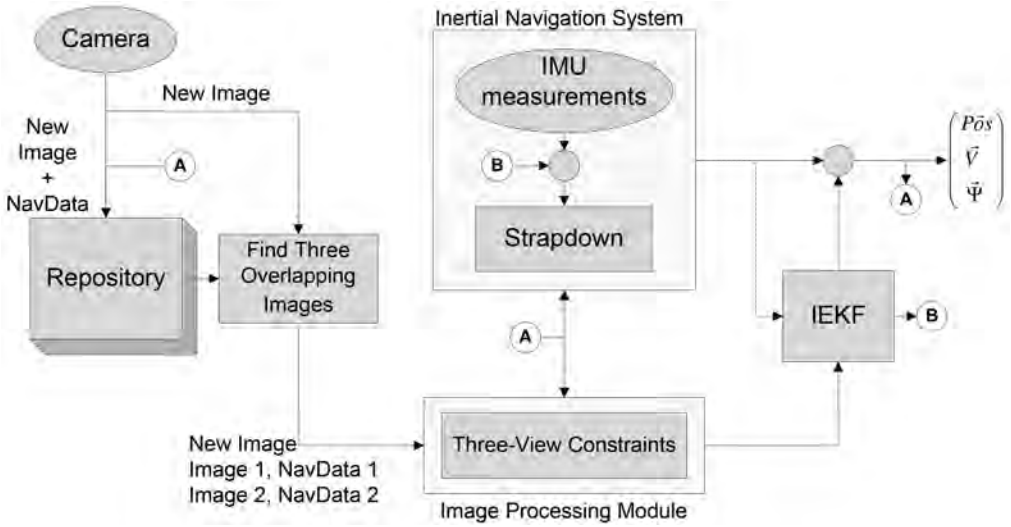

Fig. 1. Aiding an INS with three-view geometry constraints. Based on three-view geometry constraints, the filter estimates the navigation errors and a parametrization of IMU errors, which are used for correcting the navigation solution and subsequent IMU readings, respectively. In the figure, "A" denotes the corrected navigation solution, while "B" represents the estimated parametrization of IMU errors.

may be mounted on gimbals). The INS is comprised of an inertial measurement unit (IMU) whose readings are processed by the strapdown algorithm into a navigation solution.

During motion, the camera-captured images and partial navigation data, to be defined in the sequel, are stored and maintained. When a new image is captured, it is checked whether this image has a common overlapping area with two previously stored images.[1] One possible outcome of this step is a set of three overlapping images captured in close timing. Another possibility is a loop in the vehicle's trajectory, in which case the new image overlaps two stored images captured while the vehicle visited the region previously.

Once a set of three images containing a common overlapping area has been identified, the images and the navigation data associated to each image are used for calculating the constraints developed in Section III. These constraints are then reformulated into measurements and injected into an IEKF for estimating the developed navigation error and IMU errors (see Section IV). These estimates are ultimately used for correcting the navigation solution and the IMU measurements.

While some of the images in the repository are eventually used for navigation aiding, the overall set of stored images may be used for constructing a representation of the observed environment, e.g. a mosaic. The mosaic may be just a single image constructed from the set of camera-captured images (e.g. [2]), or alternatively, the mosaic may be represented by the original images accompanied by homography matrices that relate each image to a common reference frame [19]. In any case, since the navigation aiding step does not rely on the mosaic, but rather on the original images and the concomitant navigation data, the mosaic image construction may be performed in a background (low-priority) process [2]. A somewhat similar concept can be found in [20], where an architecture was developed allowing access to a repository constantly updated with readings from different sensors.

Throughout this paper, the following coordinate systems are used.

1) L—Local-level, local-north (LLLN) reference frame, also known as a North-East-Down (NED) coordinate system. Its origin is set at the location of the navigation system. X_L points North, Y_L points East, and Z_L completes a Cartesian right hand system.

2) B—Body-fixed reference frame. Its origin is set at the vehicle's center-of-mass. X_B points towards the vehicle's front, Y_B points right when viewed from above, and Z_B completes the setup to yield a Cartesian right hand system.

3) C—Camera-fixed reference frame. Its origin is set at the camera center-of-projection. Z_C points toward the field-of-view (FOV) center, X_C points toward the right half of the FOV when viewed from the camera center-of-projection, and Y_C completes the setup to yield a Cartesian right hand system.

III. THREE-VIEW GEOMETRY CONSTRAINTS DEVELOPMENT

We begin by presenting a development of constraints based on a general three-view geometry. Figure 2 shows the considered scenario, in which a single ground landmark p is observed in three images

[1]The term common overlapping area refers in this work to an area that is present in all the three images.

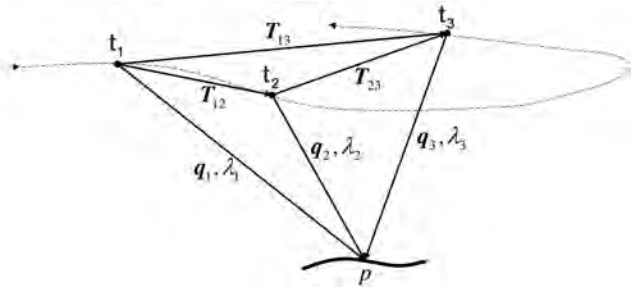

Fig. 2. Three-view geometry: ground landmark observed in three different images.

captured at time instances t_1, t_2, and t_3, where $t_1 < t_2 < t_3$. Denote by $\mathbf{T}_{ij}$ the camera translational motion from the ith to the jth view, with $i,j \in \{1,2,3\}$ and $i \neq j$. Let also $\mathbf{q}_i$ and λ_i be a line-of-sight (LOS) vector and a scale parameter, respectively, to the ground landmark p at time t_i, such that $\|\lambda_i \mathbf{q}_i\|$ is the range to this landmark. In particular, if $\mathbf{q}_i$ is a unit LOS vector, then λ_i is the range to the ground landmark.

Assuming $t_3 - t_2 > t_2 - t_1$, the translation vectors between the different views, when calculated solely based on the navigation data, will be obtained with different accuracy due to the developing inertial navigation errors: $\mathbf{T}_{12}$ contains navigation errors developed from t_1 to t_2, while $\mathbf{T}_{23}$ (and $\mathbf{T}_{13}$) is mainly affected by position errors developed from t_2 (or t_1) to t_3. Since $t_3 - t_2 > t_2 - t_1$, the accuracy of $\mathbf{T}_{23}$ is deteriorated compared with the accuracy of $\mathbf{T}_{12}$. The purpose of this section is to formulate constraints for determining $\mathbf{T}_{23}$ based on information extracted from the three images and partial navigation information (from which $\mathbf{T}_{12}$ may be calculated), thereby improving the accuracy of $\mathbf{T}_{23}$, bringing it to the accuracy levels of$\mathbf{T}_{12}$.

The position of a ground landmark p relative to the camera position at t_1, expressed in the LLLN system of t_2, can be written as

$$\lambda_1 C_{L_2}^{C_1} \mathbf{q}_1^{C_1} = C_{L_2}^{C_1} \mathbf{T}_{12}^{C_1} + \lambda_2 C_{L_2}^{C_2} \mathbf{q}_2^{C_2} \tag{1}$$

$$\lambda_1 C_{L_2}^{C_1} \mathbf{q}_1^{C_1} = C_{L_2}^{C_1} \mathbf{T}_{12}^{C_1} + C_{L_2}^{C_2} \mathbf{T}_{23}^{C_2} + \lambda_3 C_{L_2}^{C_3} \mathbf{q}_3^{C_3} \tag{2}$$

where $\mathbf{q}_i^{C_i}$ is a LOS vector to the ground feature at t_i, expressed in a camera system at t_i; $C_{L_2}^{C_i}$ is a directional cosine matrix (DCM) transforming from the camera system at t_i to the LLLN system at t_2; and $\mathbf{T}_{ij}^{C_i}$ is the platform translation from time t_i to t_j, expressed in the camera system at t_i. Here $i,j \in \{1,2,3\}$, $i \neq j$.

Subtraction of (1) from (2) and some basic algebraic manipulations give

$$\mathbf{0} = \lambda_1 C_{L_2}^{C_1} \mathbf{q}_1^{C_1} - \lambda_2 C_{L_2}^{C_2} \mathbf{q}_2^{C_2} - C_{L_2}^{C_1} \mathbf{T}_{12}^{C_1} \tag{3a}$$

$$\mathbf{0} = \lambda_2 C_{L_2}^{C_2} \mathbf{q}_2^{C_2} - \lambda_3 C_{L_2}^{C_3} \mathbf{q}_3^{C_3} - C_{L_2}^{C_2} \mathbf{T}_{23}^{C_2}. \tag{3b}$$

Since the scale parameters $\lambda_1, \lambda_2, \lambda_3$ are neither required nor known, we wish to form constraints on $\mathbf{T}_{23}$ without using these parameters, or in other words, avoid structure reconstruction. For this purpose, (3) is rewritten into the matrix form

$$\begin{bmatrix} \mathbf{q}_1 & -\mathbf{q}_2 & \mathbf{0}_{3\times 1} & -\mathbf{T}_{12} \\ \mathbf{0}_{3\times 1} & \mathbf{q}_2 & -\mathbf{q}_3 & -\mathbf{T}_{23} \end{bmatrix}_{6\times 4} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ 1 \end{bmatrix}_{4\times 1} = \mathbf{0}_{6\times 1}. \tag{4}$$

For the sake of brevity, the superscript L_2 was omitted, e.g. $\mathbf{q}_1 \equiv \mathbf{q}_1^{L_2} = C_{L_2}^{C_1} \mathbf{q}_1^{C_1}$.

Let

$$A = \begin{bmatrix} \mathbf{q}_1 & -\mathbf{q}_2 & \mathbf{0}_{3\times 1} & -\mathbf{T}_{12} \\ \mathbf{0}_{3\times 1} & \mathbf{q}_2 & -\mathbf{q}_3 & -\mathbf{T}_{23} \end{bmatrix} \in \mathbb{R}^{6\times 4}. \tag{5}$$

In a similar manner to [1] and [10], since all the components in $[\lambda_1 \ \lambda_2 \ \lambda_3 \ 1]^T$ are non-zero, it follows that $\text{rank}(A) < 4$. The following theorem provides necessary and sufficient conditions for rank deficiency of A.

THEOREM 1 $\text{rank}(A) < 4$ *if and only if all the following conditions are satisfied:*

$$\mathbf{q}_1^T (\mathbf{T}_{12} \times \mathbf{q}_2) = 0 \tag{6a}$$

$$\mathbf{q}_2^T (\mathbf{T}_{23} \times \mathbf{q}_3) = 0 \tag{6b}$$

$$(\mathbf{q}_2 \times \mathbf{q}_1)^T (\mathbf{q}_3 \times \mathbf{T}_{23}) = (\mathbf{q}_1 \times \mathbf{T}_{12})^T (\mathbf{q}_3 \times \mathbf{q}_2). \tag{6c}$$

The proof of Theorem 1 is provided in Appendix I.

The first two constraints in (6) are the well-known epipolar constraints, which force the translation vectors to be co-planar with the LOS vectors. Given multiple matching features, one can determine from (6a) and (6b) the translation vectors $\mathbf{T}_{12}$ and $\mathbf{T}_{23}$, respectively, up to scale. In general, these two scale unknowns are different. The two scales are connected through (6c), which relates between the magnitudes of $\mathbf{T}_{23}$ and $\mathbf{T}_{12}$. Consequently, if the magnitude of $\mathbf{T}_{12}$ is known, it is possible to calculate both the direction and the magnitude of $\mathbf{T}_{23}$, given multiple matching features. To the best of the authors' knowledge, the constraint (6c) has not appeared in previous publications.

Choosing the time intervals $t_2 - t_1$ and $t_3 - t_2$ should be made while considering several aspects. Most importantly, when calculating the translation vectors $\mathbf{T}_{12}$ and $\mathbf{T}_{23}$ based on data taken from the navigation system, the navigation error in $\mathbf{T}_{23}$ should be larger than the navigation error in $\mathbf{T}_{12}$. It is recommended to choose the time intervals $t_2 - t_1$ and $t_3 - t_2$ such that $\|\mathbf{T}_{12}\|$ and $\|\mathbf{T}_{23}\|$ are of the same order of magnitude. This also applies to trajectories that contain loops; however, one should note that in these cases $t_3 - t_2$ will be typically much larger than $t_2 - t_1$. In addition, choosing too small time intervals can render the constraints (6) ill-conditioned.

Several remarks are in order. First, (6) also contains rotation parameters, since all the quantities are assumed to be expressed in the LLLN system at t_2. Second, structure reconstruction is not required. As shown in the sequel, this allows to maintain a constant-size state vector comprising the vehicle's parameters only, resulting in a reduced computational load.

A. Multiple Features Formulation

In typical scenarios there is a set of matching pairs of features between the first two views, another set between the second and third view, and a set of matching triplets between all the three views, which is the intersection of the previous two sets. These sets are denoted by $\{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}\}_{i=1}^{N_{12}}, \{\mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}_{i=1}^{N_{23}}$, and $\{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}_{i=1}^{N_{123}}$, respectively, where N_{12}, N_{23}, and N_{123} are the number of matching features in each set, and $\mathbf{q}_{j_i}^{C_j}$ is the ith LOS vector in the jth view, $j \in (1,2,3)$. Note that each LOS vector is expressed in its own camera system. These LOS vectors can be expressed in the LLLN system at t_2, as was assumed in the development leading to (6), using rotation matrices whose entries are taken from the navigation system. Thus, omitting again the explicit notation of the LLLN system at t_2, we have the matching sets $\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}\}_{i=1}^{N_{12}}, \{\mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{23}}$, and $\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{123}}$. Obviously,

$$
\begin{aligned}
(\mathbf{q}_1, \mathbf{q}_2) \in \{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{123}} \\
\rightarrow (\mathbf{q}_1, \mathbf{q}_2) \in \{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}\}_{i=1}^{N_{12}} \\
(\mathbf{q}_2, \mathbf{q}_3) \in \{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{123}} \\
\rightarrow (\mathbf{q}_2, \mathbf{q}_3) \in \{\mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{23}}.
\end{aligned}
$$

The matching sets are assumed to be consistent in the following sense. Denote by $(\mathbf{q}_1^*, \mathbf{q}_2^*, \mathbf{q}_3^*)$the jth element in $\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{123}}$. Then, the matching pairs $(\mathbf{q}_1^*, \mathbf{q}_2^*)$and $(\mathbf{q}_2^*, \mathbf{q}_3^*)$ appear in the matching pairs sets $\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}\}_{i=1}^{N_{12}}$ and $\{\mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{23}}$, respectively, in the jth position as well.

Since the constraints in (6) are linear in $\mathbf{T}_{12}$ and $\mathbf{T}_{23}$, it is convenient to reorganize the equations into the following form:

$$(\mathbf{q}_1 \times \mathbf{q}_2)^T [\mathbf{q}_3]_\times \mathbf{T}_{23} = (\mathbf{q}_2 \times \mathbf{q}_3)^T [\mathbf{q}_1]_\times \mathbf{T}_{12} \tag{7}$$

$$(\mathbf{q}_2 \times \mathbf{q}_3)^T \mathbf{T}_{23} = 0 \tag{8}$$

$$(\mathbf{q}_1 \times \mathbf{q}_2)^T \mathbf{T}_{12} = 0. \tag{9}$$

Here $[.]_\times$ is the operator defined for some vector $\mathbf{a} = [a_1 \ a_2 \ a_3]^T$ as

$$[\mathbf{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{10}$$

Defining the vectors $\mathbf{f}, \mathbf{g}, \mathbf{u}, \mathbf{w} \in \mathbb{R}^{3\times 1}$ as

$$\mathbf{f}^T \doteq (\mathbf{q}_2 \times \mathbf{q}_3)^T \tag{11}$$

$$\mathbf{g}^T \doteq (\mathbf{q}_1 \times \mathbf{q}_2)^T \tag{12}$$

$$\mathbf{u}^T \doteq (\mathbf{q}_1 \times \mathbf{q}_2)^T [\mathbf{q}_3]_\times = \mathbf{g}^T [\mathbf{q}_3]_\times \tag{13}$$

$$\mathbf{w}^T \doteq (\mathbf{q}_2 \times \mathbf{q}_3)^T [\mathbf{q}_1]_\times = \mathbf{f}^T [\mathbf{q}_1]_\times \tag{14}$$

and considering all the matching pairs and triplets, (7)–(9) turn into

$$[\mathbf{u}_i^T]_{1\times 3} \mathbf{T}_{23} = [\mathbf{w}_i^T]_{1\times 3} \mathbf{T}_{12} \tag{15}$$

$$[\mathbf{f}_j^T]_{1\times 3} \mathbf{T}_{23} = 0 \tag{16}$$

$$[\mathbf{g}_k^T]_{1\times 3} \mathbf{T}_{12} = 0 \tag{17}$$

with $i = 1 \ldots N_{123}$, $j = 1 \ldots N_{23}$, $k = 1 \ldots N_{12}$. Stacking these equations together yields

$$\begin{bmatrix} U \\ F \\ 0 \end{bmatrix}_{N\times 3} \mathbf{T}_{23} = \begin{bmatrix} W \\ 0 \\ G \end{bmatrix}_{N\times 3} \mathbf{T}_{12} \tag{18}$$

where $N \doteq N_{12} + N_{23} + N_{123}$ and

$$U = [\mathbf{u}_1 \cdots \mathbf{u}_{N_{123}}]^T \tag{19}$$

$$W = [\mathbf{w}_1 \cdots \mathbf{w}_{N_{123}}]^T \tag{20}$$

$$F = [\mathbf{f}_1 \cdots \mathbf{f}_{N_{23}}]^T \tag{21}$$

$$G = [\mathbf{g}_1 \cdots \mathbf{g}_{N_{12}}]^T. \tag{22}$$

If $\mathbf{T}_{12}$ and the rotation matrices are given (e.g. by the navigation system), the minimum number of matching features required for determining the vector $\mathbf{T}_{23}$ are a single matching pair between the second and the third views, and one matching triplet that may be utilized both in the trifocal constraint (7) and in the epipolar constraint (8). Moreover, since $\mathbf{T}_{12}$ is known with a certain level of accuracy, it is not essential to use the epipolar constraint for the first two views. Application of this constraint, however, is expected to improve the a priori accuracy of $\mathbf{T}_{12}$, and therefore reduce the estimation error of $\mathbf{T}_{23}$.

An alternative formulation of the constraints induced by three-view geometry of a general scene is described by the trifocal tensor [1]. Indeed, the application of the trifocal tensor was already suggested for estimating the camera motion [21, 22]. However, three-view geometry, and in particular the trifocal tensor and the constraints proposed herein, have not been used thus far for navigation aiding. Moreover, while the trifocal tensor approach is solely based on matching triplets, the constraints formulation presented in (18) allows using matching pairs as well. This is expected to improve the state estimation accuracy, since in typical applications the

cardinality of the sets of matching pairs $\{\mathbf{q}_{1_i},\mathbf{q}_{2_i}\}_{i=1}^{N_{12}}$ and $\{\mathbf{q}_{2_i},\mathbf{q}_{3_i}\}_{i=1}^{N_{23}}$ is much larger than the cardinality of the set of matching triplets $\{\mathbf{q}_{1_i},\mathbf{q}_{2_i},\mathbf{q}_{3_i}\}_{i=1}^{N_{123}}$.

While the development of the constraints in (18) assumed a general ground scene, when a planar scene is under consideration, an additional constraint, expressing the fact that all the observed features are located on the same plane [10, 8], may be incorporated.

One may estimate $\mathbf{T}_{23}$ based on (18) using standard techniques (e.g. singular value decomposition (SVD)) and then fuse $\mathbf{T}_{23}$ with the INS. However, a better alternative is to utilize the implicit nature of (18) using an IEKF [24], as discussed in the next section.

IV. FUSION WITH A NAVIGATION SYSTEM

In this section we present a technique for fusing the three-view geometry constraints with a standard navigation system, assuming three images with a common overlapping area had been identified. The data fusion is performed using an indirect IEKF that estimates the navigation parameter errors instead of the parameters themselves. These estimated errors are then used for correcting the navigation solution computed by the navigation system (see Fig. 1).

When real imagery and navigation data are considered, the existence of navigation errors and image noise renders the constraints of (18) inaccurate. Thus, the following residual measurement is defined:

$$\mathbf{z} \doteq \begin{bmatrix} U \\ F \\ 0 \end{bmatrix}_{N\times 3} \mathbf{T}_{23} - \begin{bmatrix} W \\ 0 \\ G \end{bmatrix}_{N\times 3} \mathbf{T}_{12} \doteq \mathcal{A}\mathbf{T}_{23} - \mathcal{B}\mathbf{T}_{12}. \tag{23}$$

Since $\mathbf{T}_{12} = \mathbf{Pos}(t_2) - \mathbf{Pos}(t_1)$, $\mathbf{T}_{23} = \mathbf{Pos}(t_3) - \mathbf{Pos}(t_2)$, and the matrices F,G,U,W are functions of the LOS vectors, the residual measurement $\mathbf{z}$ is a nonlinear function of the following parameters[2]:

$$\begin{aligned}\mathbf{z} = \mathbf{h}(&\mathbf{Pos}(t_3),\boldsymbol{\Psi}(t_3),\mathbf{Pos}(t_2),\boldsymbol{\Psi}(t_2),\\ &\mathbf{Pos}(t_1),\boldsymbol{\Psi}(t_1),\{\mathbf{q}_{1_i}^{C_1},\mathbf{q}_{2_i}^{C_2},\mathbf{q}_{3_i}^{C_3}\}).\end{aligned} \tag{24}$$

Here (t_3,t_2,t_1) denote the time instances in which the three overlapping images were captured, with t_3 being the current time.

We now define the state vector as

$$\mathbf{X} = [\Delta\mathbf{P}^T \quad \Delta\mathbf{V}^T \quad \Delta\boldsymbol{\Psi}^T \quad \mathbf{d}^T \quad \mathbf{b}^T]^T \tag{25}$$

where $\Delta\mathbf{P}\in\mathbb{R}^3$, $\Delta\mathbf{V}\in\mathbb{R}^3$, $\Delta\boldsymbol{\Psi} = (\Delta\phi,\Delta\theta,\Delta\psi)^T \in [0,2\pi]\times[0,\pi]\times[0,2\pi]$ are the position, velocity and attitude errors, respectively, and $(\mathbf{d},\mathbf{b})$ is the parameterization of errors in the inertial sensor measurements: $\mathbf{d}\in\mathbb{R}^3$ is the gyro drift, and $\mathbf{b}\in\mathbb{R}^3$ is the accelerometer bias. The first 9 components of $\mathbf{X}$ are given in LLLN coordinates, while the last 6 are written in a body-fixed reference frame. The corresponding transition matrix $\Phi_d(t_b,t_a)$ satisfying $\mathbf{X}(t_b) = \Phi_d(t_b,t_a)\mathbf{X}(t_a)$ is given in [3].

Since it is unknown a priori which three images will have a common overlapping area, and in order to maintain a constant-size state vector, each captured image should be stored and associated with the relevant navigation information. The navigation data that should be attached to each image are the platform position, attitude, gimbal angles, and the filter's covariance matrix.

Linearizing $\mathbf{h}$ about $\mathbf{Pos}(t_3)$, $\boldsymbol{\Psi}(t_3)$, $\mathbf{Pos}(t_2)$, $\boldsymbol{\Psi}(t_2)$, $\mathbf{Pos}(t_1)$, $\boldsymbol{\Psi}(t_1)$, and $\{\mathbf{q}_{1_i}^{C_1},\mathbf{q}_{2_i}^{C_2},\mathbf{q}_{3_i}^{C_3}\}$, and keeping the first-order terms yields

$$\mathbf{z} \approx H_3\mathbf{X}(t_3) + H_2\mathbf{X}(t_2) + H_1\mathbf{X}(t_1) + D\mathbf{v} \tag{26}$$

where $H_3,H_2,H_1\in\mathbb{R}^{N\times 15}$ are defined as

$$H_3 \doteq \nabla_{\boldsymbol{\zeta}(t_3)}\mathbf{h}, \qquad H_2 \doteq \nabla_{\boldsymbol{\zeta}(t_2)}\mathbf{h}, \qquad H_1 \doteq \nabla_{\boldsymbol{\zeta}(t_1)}\mathbf{h} \tag{27}$$

while $\boldsymbol{\zeta}$ is composed of the navigation solution and IMU errors parametrization:

$$\boldsymbol{\zeta} \doteq [\mathbf{Pos}^T \quad \mathbf{V}^T \quad \boldsymbol{\Psi}^T \quad \mathbf{d}^T \quad \mathbf{b}^T] \tag{28}$$

with $\mathbf{Pos}$, $\mathbf{V}$, and $\boldsymbol{\Psi}$ representing position, velocity, and attitude calculated by the INS, respectively.

The terms $\mathbf{X}(t_3)$, $\mathbf{X}(t_2)$, and $\mathbf{X}(t_1)$ in (26) are the navigation errors at the three time instances; in general, $\mathbf{X}(t_1)$, $\mathbf{X}(t_2)$, and $\mathbf{X}(t_3)$ may be correlated.

Noting that we are only interested in estimating the navigation errors at the current time instant $\mathbf{X}(t_3)$, the navigation errors at the first two time instances are considered as random parameters in the measurement equation. Therefore, since $\mathbf{X}(t_2)$ and $\mathbf{X}(t_1)$ are not estimated, the estimation error $\tilde{\mathbf{X}} \doteq \mathbf{X} - \hat{\mathbf{X}}$ in these two time instances is $\tilde{\mathbf{X}}(t_2) \equiv \mathbf{X}(t_2)$ and $\tilde{\mathbf{X}}(t_1) \equiv \mathbf{X}(t_1)$, respectively. These errors are represented by the filter covariance matrices $P(t_1)$, $P(t_2)$, respectively, which are attached to the first two images.

The matrix D in (26) is the gradient of $\mathbf{h}$ with respect to the LOS vectors, i.e., $D \doteq \nabla_{\{\mathbf{q}_{1_i}^{C_1},\mathbf{q}_{2_i}^{C_2},\mathbf{q}_{3_i}^{C_3}\}}\mathbf{h}$, and $\mathbf{v}$ is the image noise associated with the LOS vectors, having a covariance matrix R. Thus, the measurement noise is modeled as a combination of image noise, with the appropriate Jacobian matrix D, and the estimation errors $\tilde{\mathbf{X}}(t_2)$ and $\tilde{\mathbf{X}}(t_1)$ with the Jacobian matrices H_2 and H_1, respectively. The development of the matrices H_3,H_2,H_1,D, and R is given in Appendix II.

[2] In (24), the notation $\{\mathbf{q}_{1_i}^{C_1},\mathbf{q}_{2_i}^{C_2},\mathbf{q}_{3_i}^{C_3}\}$ refers to the fact that LOS vectors from all the three images are used for calculating the residual measurement $\mathbf{z}$. Note that each of the matrices F,G,U,W is a function of a different set of matching points.

The propagation step of the filter is carried out using the matrix Φ_d and the state vector $\mathbf{X} \in \mathbb{R}^{15\times 1}$, as explained in [2]. The update step is executed only when a set of three overlapping images becomes available. In this step the current state vector $\mathbf{X}(t_3)$ is estimated based on the LOS vectors and the first two state vectors $\mathbf{X}(t_1)$, $\mathbf{X}(t_2)$, as explained next. This is in contrast to the SLAM approach, in which both the propagation and update steps of the filter are performed on a state vector that constantly increases in size.

The Kalman gain matrix is given by

$$\begin{aligned} K &= P_{\mathbf{X}(t_3)\mathbf{z}(t_3,t_2,t_1)} P^{-1}_{\mathbf{z}(t_3,t_2,t_1)} \\ &= E[\tilde{\mathbf{X}}^- \tilde{\mathbf{z}}^T] E[\tilde{\mathbf{z}}\tilde{\mathbf{z}}^T]^{-1} \\ &= E[(\mathbf{X}-\hat{\mathbf{X}}^-)(\mathbf{z}-\hat{\mathbf{z}})^T] E[(\mathbf{z}-\hat{\mathbf{z}})(\mathbf{z}-\hat{\mathbf{z}})^T]^{-1} \end{aligned} \quad (29)$$

where the explicit time notations were omitted for conciseness.

Since $\hat{\mathbf{z}} = H_3 \hat{\mathbf{X}}^-(t_3)$

$$\begin{aligned} \tilde{\mathbf{z}} &= \mathbf{z} - \hat{\mathbf{z}} \\ &= H_3 \tilde{\mathbf{X}}^-(t_3) + H_2 \tilde{\mathbf{X}}(t_2) + H_1 \tilde{\mathbf{X}}(t_1) + D\mathbf{v}. \end{aligned} \quad (30)$$

Hence

$$P_{\mathbf{X}(t_3)\mathbf{z}(t_3,t_2,t_1)} = P_3^- H_3^T + P_{32}^- H_2^T + P_{31}^- H_1^T \quad (31)$$

$$\begin{aligned} P_{\mathbf{z}(t_3,t_2,t_1)} &= H_3 P_3^- H_3^T \\ &\quad + [H_2 \quad H_1] \begin{bmatrix} P_2 & P_{21} \\ P_{21}^T & P_1 \end{bmatrix} [H_2 \quad H_1]^T + DRD^T \end{aligned} \quad (32)$$

where $P_i = E[\tilde{\mathbf{X}}_i \tilde{\mathbf{X}}_i^T]$ and $P_{ij} = E[\tilde{\mathbf{X}}_i \tilde{\mathbf{X}}_j^T]$.

As the measurement noise $H_2\mathbf{X}(t_2) + H_1\mathbf{X}(t_1) + D\mathbf{v}$ is statistically dependent with the state vector to be estimated, $\mathbf{X}(t_3)$, the basic assumption of the Kalman filter is contradicted. Equations (31) and (32) are an ad-hoc approach for taking into consideration this dependence within the Kalman filter framework, that has given good results. Note that if all the three state vectors, $\mathbf{X}(t_3)$, $\mathbf{X}(t_2)$ and $\mathbf{X}(t_1)$, were to be estimated, the measurement noise in (26) would be $D\mathbf{v}$, which is still statistically dependent with the state vectors. However, this dependence would only be due to the Jacobian D, as modeled by a standard IEKF formulation [24, 31]. Explicit equations in such case are given, for example, in [32].

Referring to (31) and (32), while the matrices P_3^-, P_2, and P_1 are known, the cross-correlation matrices P_{32}^-, P_{31}^-, and P_{21} are unknown, and therefore need to be calculated. However, since $\mathbf{X}(t_2)$ and $\mathbf{X}(t_1)$ are stored outside the filter, these terms cannot be calculated without additional information or assumptions.

This issue is handled as follows. Inertial navigation between t_1 and t_2 is assumed. Denoting by $\Phi_d(t_2,t_1)$ the transition matrix between $\mathbf{X}(t_1)$ and $\mathbf{X}(t_2)$, the term P_{21} may be calculated as

$$P_{21} = E[\tilde{\mathbf{X}}(t_2)\tilde{\mathbf{X}}^T(t_1)] = \Phi_d(t_2,t_1) P_1. \quad (33)$$

The other two cross-correlation terms, $P_{32}^- = E[\tilde{\mathbf{X}}^-(t_3)\tilde{\mathbf{X}}^T(t_2)]$ and $P_{31}^- = E[\tilde{\mathbf{X}}^-(t_3)\tilde{\mathbf{X}}^T(t_1)]$, may be neglected if $t_3 \gg t_2$ (e.g. loops), or when the first two images and their associated navigation data have been received from an external source (e.g. some other vehicle).

Several approaches exist for handling all the other cases in which $t_3 - t_2$ is not considerably large. One possible approach is to keep a limited history of the platform navigation parameters by incorporating these parameters into the state vector each time a new image is captured within a certain sliding window [7]. This approach is capable of handling scenarios in which all the three images are captured within the assumed sliding window. Another alternative would be to develop a bound on $t_3 - t_2$ under which the cross-correlation terms P_{32}^- and P_{31}^- can be considered negligible, and select sets of overlapping images accordingly. These two approaches may also be jointly applied. Covariance intersection (CI) [26, 27] could also be potentially used to deal with the cross-correlation terms. However, CI is incapable of handling cases in which the measurement matrix contains only a partial representation of the state vector [27, 28], which is the situation in the present case.

In this work it is assumed that the current navigation parameters are not correlated with the navigation parameters that are associated with the first two images, i.e., $P_{32}^- = 0$ and $P_{31}^- = 0$.

In case the above assumptions regarding P_{32}^-, P_{31}^-, and P_{21} are not satisfied, these terms can be explicitly calculated using the method developed in [33]. This method allows calculating the cross-covariance terms for a general multi-platform measurement model assuming all the thus-far performed multi-platform measurement updates are stored in a graph. As described in [32], the method can be adjusted in a straightforward manner to the three-view constraints measurement model (26) considered in this paper.

After the residual measurement and the gain matrix have been computed using (24) and (29), respectively, the state vector and the covariance matrix can be updated based on the standard equations of the IEKF.

A. Computational Requirements

A single filter update step, given three images with a common overlapping area, involves computation of the matrices $\mathcal{A}, \mathcal{B}$ and the Jacobian matrices H_3, L_2, L_1, and D. These calculations are linear in N, the overall size of the matching sets

$\{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}_{i=1}^{N_{123}}$, $\{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}\}_{i=1}^{N_{12}}$, and $\{\mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}_{i=1}^{N_{23}}$. Noting that the state vector is constant in size, the most computationally expensive operation in the filter update step is the inversion of an $N \times N$ matrix required for the calculation of the gain matrix.

The computational load of the proposed method does not change significantly over time (depending on the variation of N), regardless of the scenarios in which the algorithm is applied to (including loops). Moreover, if the computational capability is limited, it is possible to utilize only part of the available matching pairs and triplets (see Section III-A), or eliminate the epipolar constraint for the first two views, thus reducing the computational load even further.

In contrast to the above, the computational requirements of other methods capable of handling trajectory loops, are much higher. Conventional SLAM entails constantly-increasing computational requirements, due to the augmentation of the state vector. Furthermore, the high computational load is induced in each filter propagation step. For example, denote by d the number of elements added to the state vector each time a new image is captured. After using n images, the state vector in SLAM will consist of nd elements, representing the observed scene, and of navigation parameters. In contrast to this, in our approach the state vector is a fixed-size 15-element vector, being propagated and updated by the filter. Methods that perform state augmentation until a certain size of the state vector is reached (e.g. [7]), handle loops in the trajectory by applying bundle adjustment over all the images that have been captured during the loop chain, as opposed to processing only three images as done in our approach.

B. Extensions

It is straightforward to extend the developed method for handling more than three overlapping images, which may improve robustness to noise. In the general case, assume k given images, such that each three neighboring images are overlapping (a common overlapping area for all the k images is not required). Assume also that all these images are associated with the required navigation data. In the spirit of (6), we write an epipolar constraint for each pair of consecutive images, and a constraint for relating the magnitudes of the translation vectors (similar to (6c)) for each three adjacent overlapping images. Next, the residual measurement $\mathbf{z}$ is redefined and the calculations of the required Jacobian matrices in the IEKF formulation are repeated.

For example, consider the case of four images captured at time instances $t_1,\ldots,t_4$, with t_4 being the current time, and assume existence of common overlapping areas for the first three images and for the last three images. One possible formulation of the constraints is

$$(\mathbf{q}_1 \times \mathbf{q}_2)^T [\mathbf{q}_3]_\times \mathbf{T}_{23} = (\mathbf{q}_2 \times \mathbf{q}_3)^T [\mathbf{q}_1]_\times \mathbf{T}_{12} \quad (34)$$

$$(\mathbf{q}_2 \times \mathbf{q}_3)^T \mathbf{T}_{23} = 0 \quad (35)$$

$$(\mathbf{q}_1 \times \mathbf{q}_2)^T \mathbf{T}_{12} = 0 \quad (36)$$

$$(\mathbf{q}_2 \times \mathbf{q}_3)^T [\mathbf{q}_4]_\times \mathbf{T}_{34} = (\mathbf{q}_3 \times \mathbf{q}_4)^T [\mathbf{q}_2]_\times \mathbf{T}_{23} \quad (37)$$

$$(\mathbf{q}_3 \times \mathbf{q}_4)^T \mathbf{T}_{34} = 0. \quad (38)$$

Considering all the available matches and following the same procedure as in Section IV, the residual measurement $\mathbf{z}$ will assume the form

$$\mathbf{z} = \mathcal{J}\mathbf{T}_{34} - \mathcal{V}\mathbf{T}_{23} - \mathcal{L}\mathbf{T}_{12}$$

where the matrices $\mathcal{J}, \mathcal{V}, \mathcal{L}$ are constructed based on (34)–(38).

Since $\mathbf{T}_{12}$, $\mathbf{T}_{23}$ and all the rotation matrices that implicitly appear in (34)–(38) can be calculated based on the navigation data associated with the images, the residual measurement $\mathbf{z}$ is given by

$$\mathbf{z} = \mathbf{h}(\mathbf{Pos}(t_4), \mathbf{\Psi}(t_4), \mathbf{Pos}(t_3), \mathbf{\Psi}(t_3), \mathbf{Pos}(t_2), \mathbf{\Psi}(t_2), \mathbf{Pos}(t_1), \mathbf{\Psi}(t_1), \{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}, \mathbf{q}_{4_i}^{C_4}\})$$

in which $\mathbf{Pos}(t_4), \mathbf{\Psi}(t_4)$ are part of the current navigation solution. This measurement may be utilized for estimating the developed navigation errors in the same manner as discussed in Section IV. The involved computational requirements will increase only in the update step, according to the total size of the matching sets. The propagation step of the filter remains the same.

V. SIMULATION AND EXPERIMENTAL RESULTS

This section presents statistical results obtained from simulated navigation data and synthetic imagery data, as well as experimental results utilizing real navigation and imagery data.

A. Implementation Details

1) *Navigation Simulation*: The navigation simulation consists of the following steps [2]: a) trajectory generation; b) velocity and angular velocity increments extraction from the created trajectory; c) IMU error definition and contamination of pure increments by noise; and d) strapdown calculations. The strapdown mechanism provides, at each time step, the calculated position, velocity, and attitude of the vehicle. Once a set of three images with a common overlapping area is available, the developed algorithm is executed: the state vector is estimated based on the developed algorithm using IEKF, which is then used for updating the navigation solution (see Fig. 1). The estimated bias and drift are used for correcting the IMU measurements.

2) *Image Processing Module*: Given three images with a common overlapping area, the image processing phase includes features extraction from each image using the SIFT algorithm [29] and computation of sets of matching pairs between the first two images $\{\mathbf{x}_1^i,\mathbf{x}_2^i\}_{i=1}^{N_{12}}$, and between the last two images $\{\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{23}}$, where $\mathbf{x}^i=(x^i,y^i)^T$ are the image coordinates of the ith feature. This computation proceeds as follows. First, the features are matched based on their descriptor vectors (that were computed as part of the SIFT algorithm), yielding the sets $\{\mathbf{x}_1^i,\mathbf{x}_2^i\}_{i=1}^{\tilde{N}_{12}},\{\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{\tilde{N}_{23}}$. Since this step occasionally produces false matches (outliers), the RANSAC algorithm [30] is applied over the fundamental matrix [1] model in order to reject the existing false matches, thus obtaining the refined sets $\{\mathbf{x}_1^i,\mathbf{x}_2^i\}_{i=1}^{N_{12}}$ and $\{\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{23}}$. The fundamental matrices are not used in further computations.

The next step is to use these two sets for calculating matching triplet features, i.e., matching features in the three given images. This step is performed by matching all $\mathbf{x}_1\in\{\mathbf{x}_1^i,\mathbf{x}_2^i\}_{i=1}^{N_{12}}$ with all $\mathbf{x}_3\in\{\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{23}}$, yielding a set of matching triplets $\{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}}$. The matching process includes the same steps as described above.

When using synthetic imagery data, a set of points in the real-world are randomly drawn. Then, taking into account the camera motion, known from the true vehicle trajectory, and assuming specific camera calibration parameters, the image coordinates of the observed real-world points are calculated using a pinhole projection [1] at the appropriate time instances. See, for example, [31] for further details. Consequently, a list of features for each time instant of the three time instances, which are manually specified, is obtained: $\{\mathbf{x}_1^i\}$, $\{\mathbf{x}_2^i\}$, and $\{\mathbf{x}_3^i\}$. The mapping between these three sets is known, since these sets were calculated using the pinhole projection based on the same real-world points. Thus, in order to find the matching sets $\{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}},\{\mathbf{x}_1^i,\mathbf{x}_2^i\}_{i=1}^{N_{12}}$, and $\{\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{23}}$ it is only required to check which features are within the camera FOV at all the three time instances.

Finally, the calculated sets of matching features are transformed into sets of matching LOS vectors. A LOS vector, expressed in the camera system for some feature $\mathbf{x}=(x,y)^T$, is calculated as $\mathbf{q}^C=(x,y,f)^T$, where f is the camera focal length. As a result, three matching LOS sets are obtained: $\{\mathbf{q}_{1_i}^{C_1},\mathbf{q}_{2_i}^{C_2},\mathbf{q}_{3_i}^{C_3}\}_{i=1}^{N_{123}},\{\mathbf{q}_{1_i}^{C_1},\mathbf{q}_{2_i}^{C_2}\}_{i=1}^{N_{12}}$, and $\{\mathbf{q}_{2_i}^{C_2},\mathbf{q}_{3_i}^{C_3}\}_{i=1}^{N_{23}}$. When handling real imagery, the camera focal length, as well as other camera parameters, are found during the camera calibration process. In addition, a radial distortion correction [1] was applied to camera-captured images, or alternatively, to the extracted feature coordinates.

TABLE I
Initial Navigation Errors and IMU Errors

Parameter	Description	Value	Units
$\Delta\mathbf{P}$	Initial position error (1σ)	$(100,100,100)^T$	m
$\Delta\mathbf{V}$	Initial velocity error (1σ)	$(0.3,0.3,0.3)^T$	m/s
$\Delta\boldsymbol{\Psi}$	Initial attitude error (1σ)	$(0.1,0.1,0.1)^T$	deg
$\mathbf{d}$	IMU drift (1σ)	$(10,10,10)^T$	deg/hr
$\mathbf{b}$	IMU bias (1σ)	$(10,10,10)^T$	mg

B. Statistical Results based on Simulated Navigation and Synthetic Imagery

In this section we present statistical results obtained by applying the developed algorithm to a trajectory containing a loop based on a simulated navigation system and synthetic imagery data. The assumed initial navigation errors and IMU errors are summarized in Table I. The synthetic imagery data was obtained by assuming a $20^\circ\times30^\circ$ camera FOV, focal length of 1570 pixels, and image noise of 1 pixel. The assumed trajectory, shown in Fig. 3(a), includes a loop that is repeated twice (see also Fig. 3(b)).

In order to demonstrate the performance of the algorithm in loop scenarios, the three-view navigation-aiding algorithm was applied twice, at $t=427$ s and at $t=830$ s, each time a specific point along the trajectory was revisited. The true translation vectors are $\mathbf{T}_{12}^L=[100\ 0\ 0]^T$ and $\mathbf{T}_{23}^L=[500\ 0\ 0]^T$. No other updates of the navigation system were performed, i.e., inertial navigation was applied elsewhere.

Figure 4 provides the Monte-Carlo results (100 runs). As seen, with the help of the three-view update, the position error (which has grown to several kilometers because of the inertial navigation phase) is reset in all axes to the levels of errors at t_1 and t_2 (see Fig. 4(b)). The velocity error is also considerably reduced in all axes as a result of the algorithm activation, while the accelerometer bias is estimated mainly in the z axis (see Fig. 4(d)).

Assuming at least three matching triplets of features exist, the proposed method can be applied without using the epipolar constraints, utilizing only the constraint relating the magnitudes of translation vectors (15). In this case the accuracy of the method will degrade, mainly in a direction normal to the motion heading, as shown in Fig. 5. The position error in the North direction, which is the motion heading at the time of the algorithm activation, is roughly the same as in the case where all the constraints in (18) are applied. However, in the East direction the accuracy of the position state is considerably degraded, with an error of around 900 m, compared with an error of about 100 m (Fig. 4(b)), which is the initial position error (see Table I). Observe also that although the error in the down direction has

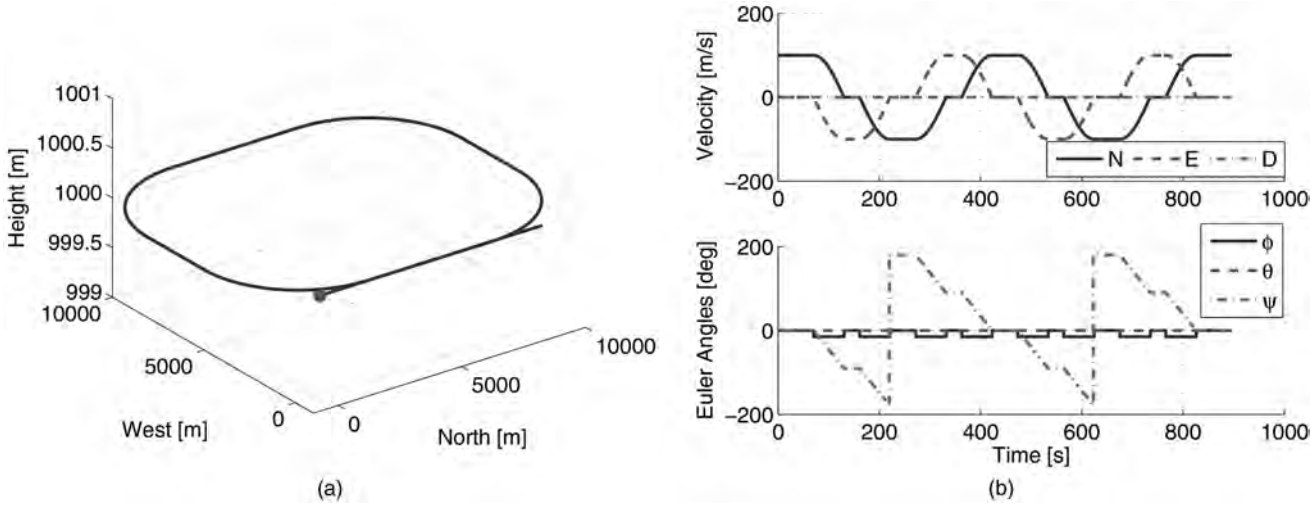

Fig. 3. Trajectory used in statistical study. Vehicle performs loop twice. (a) Position. Filled circle indicates initial position. (b) Velocity and Euler angles.

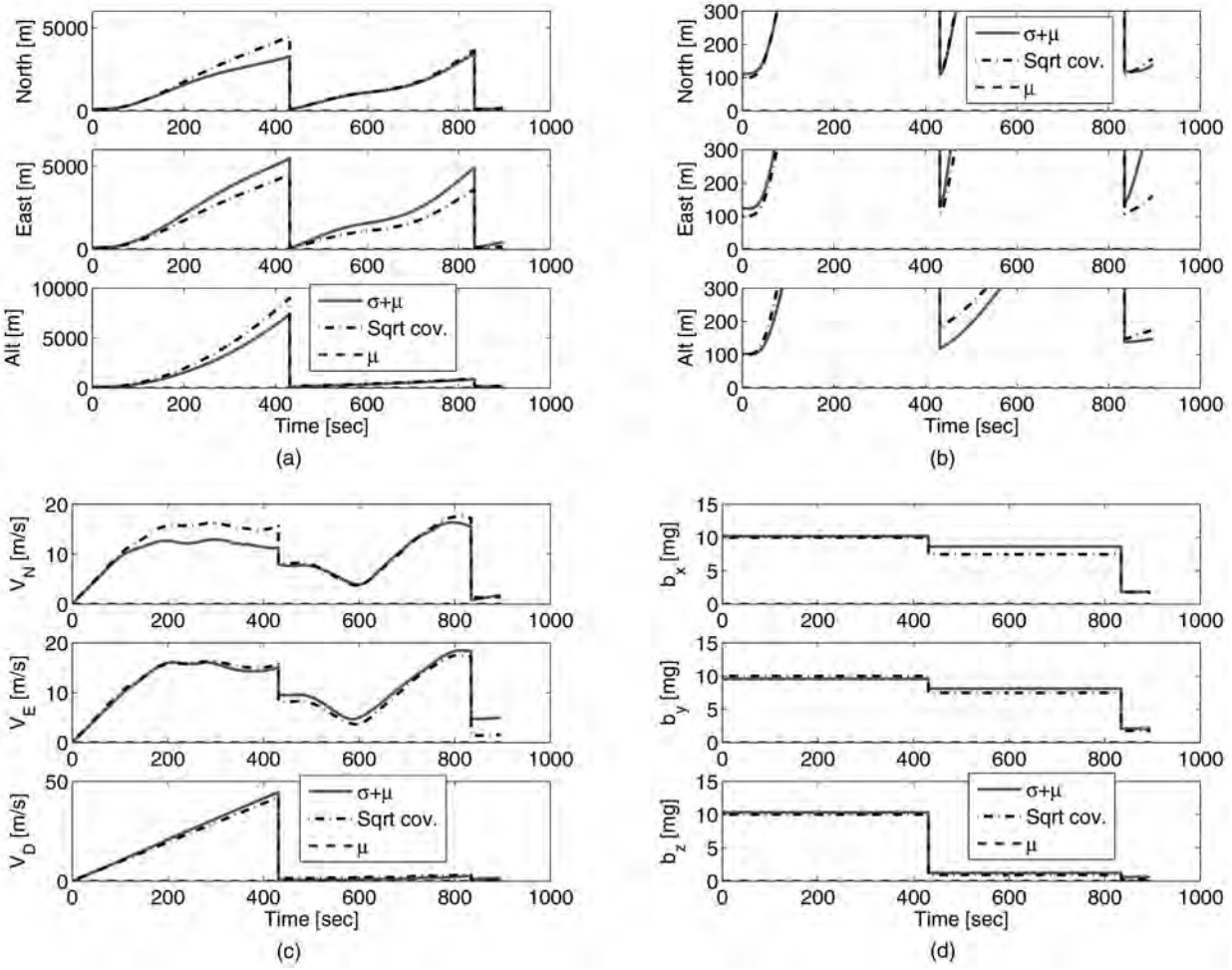

Fig. 4. Monte-Carlo results of three-view navigation-aiding algorithm based on navigation simulation and synthetic imagery data. (a) Position errors. (b) Position errors—zoom. (c) Velocity errors. (d) Bias estimation errors.

not significantly changed, the filter covariance is no longer consistent (the same filter tuning was used in both cases). The absolute reduction of position and velocity errors in all axes is not possible when applying two-view based techniques for navigation aiding, since the position and velocity along the motion direction are unobservable [2, 3]. In practical applications each of the two approaches may be applied, depending on the number of available overlapping images. Whenever a set of three images

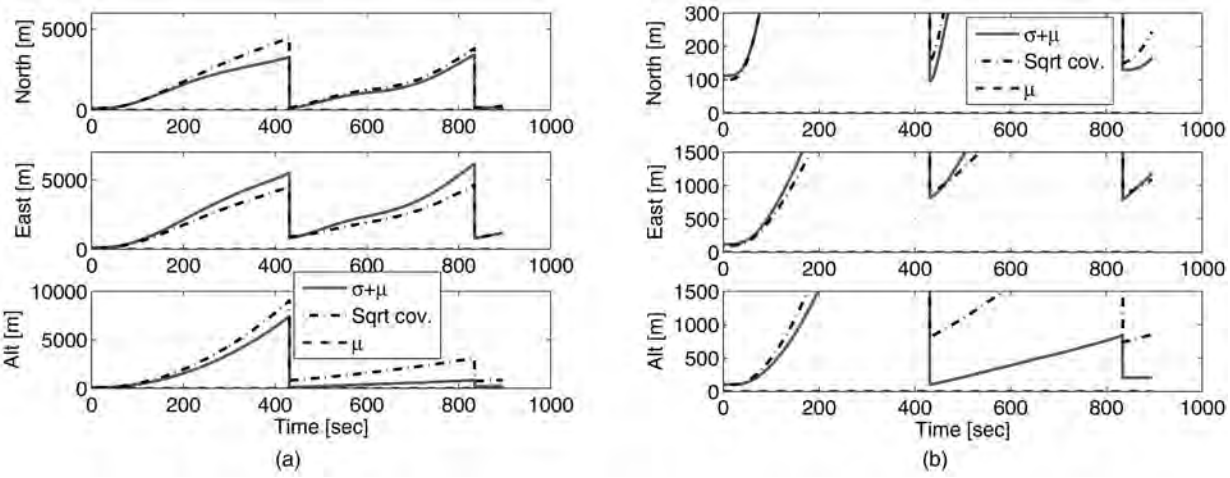

Fig. 5. Monte-Carlo results of the three-view navigation-aiding algorithm based on navigation simulation and synthetic imagery data without applying epipolar constraints. (a) Position errors. (b) Position errors—zoom.

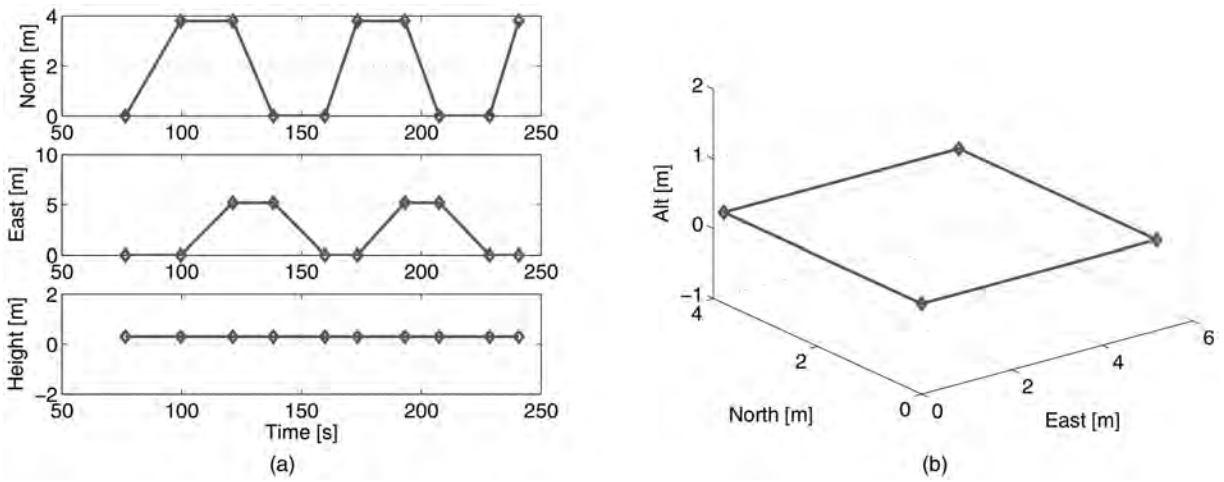

Fig. 6. Trajectory performed in experiment. (a) True trajectory. (b) True trajectory—3D view.

with a common overlapping area becomes available, the proposed method will reduce the navigation errors that two-view navigation aiding methods were unable to estimate (e.g. errors along motion heading) in accordance with the quality of navigation data attached to the first two images in the set.

C. Experiment Results

An experiment was carried out for validating the proposed method. The experimental setup contained an MTi-G Xsens[3] IMU/INS and a 207MW Axis network camera[4] that were mounted on top of a ground vehicle. The vehicle was manually commanded using a joystick, while the camera captured images perpendicular to the motion heading. During the experiment, the inertial sensor measurements and camera images were recorded for postprocessing at 100 Hz and 15 Hz, respectively. In addition, these two data sources were synchronized by associating to each image a time stamp from the navigation timeline.

Since the experiment was carried out indoors, GPS was unavailable, and therefore the MTi-G could not supply a valid navigation solution for reference. However, the true vehicle trajectory was manually measured during the experiment and associated with a timeline by postprocessing the inertial sensors readings. The reference trajectory is shown in Fig. 6. The diamond markers denote the manual measurements of the vehicle position, while the solid line represents a linear interpolation between each two markers. The vehicle began its motion at $t \approx 76$ s. As can be seen in Fig. 6(a), the vehicle performed the same closed trajectory twice (see also Fig. 6(b)).

The recorded inertial sensor measurements were processed by the strapdown block yielding an inertial navigation solution. Sets of three images with a

[3]http://www.xsens.com/en/general/mti-g.

[4]http://www.axis.com/products/cam_207mw/index.htm.

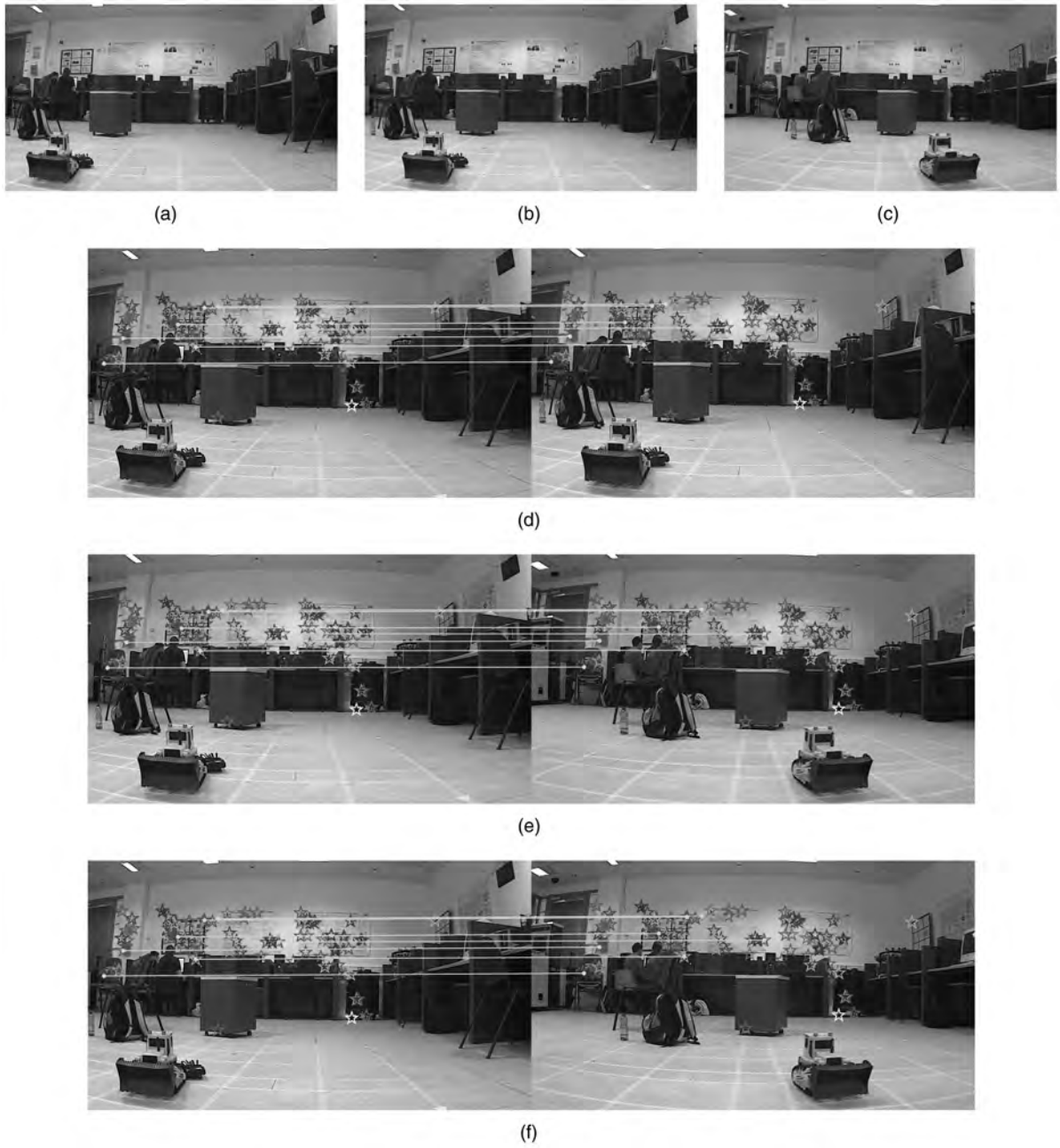

Fig. 7. Image matching process. (a)–(c) Three camera-captured images used in first sequential update in experiment. (d) Matching triplets between image 1 and 2: $(\mathbf{x}_1,\mathbf{x}_2) \in \{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}}$. (e) Matching triplets between image 2 and 3: $(\mathbf{x}_2,\mathbf{x}_3) \in \{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}}$. (f) Matching triplets between image 1 and 3: $(\mathbf{x}_1,\mathbf{x}_3) \in \{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}}$. For clarity, only first few matches are explicitly shown; rest of matches are denoted by marks in each image.

common overlapping area were identified and chosen. The proposed algorithm was applied for each such set and used for updating the navigation system. Two different update modes are demonstrated in this experiment: 1) "sequential update," in which all the three images are acquired closely to each other, and 2) "loop update," in which the first two images are captured while the platform passes a given region for the first time, whereas the third image is obtained at the second passing of the same region. The algorithm application is the same in both cases.

The image matching process for the first set of three overlapping images is shown in Fig. 7. The camera-captured images are given in Figs. 7(a)–(c). The set of matching triplets $\{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}}$ is provided in Figs. 7(d)–(f), showing matches between each pair of images. For example, Fig. 7(d) shows the matches between the first and second image, such that $(\mathbf{x}_1,\mathbf{x}_2) \in \{\mathbf{x}_1^i,\mathbf{x}_2^i,\mathbf{x}_3^i\}_{i=1}^{N_{123}}$. As seen, the three images have a significant common overlapping area, and thus it is possible to obtain a large number of matching triplets. About 140 matching triplets were

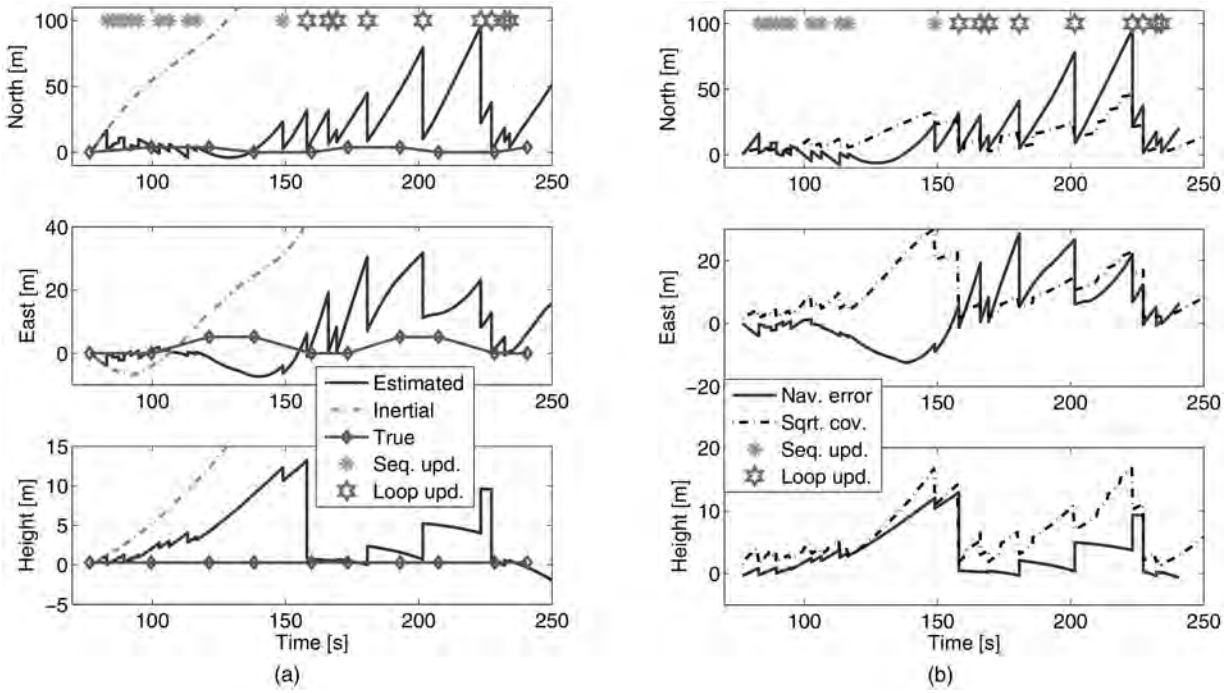

Fig. 8. Experiment results. Small position error (several meters) is obtained while sequentially activating algorithm. Position error is reset to its prior levels each time loop update is applied. (a) Estimated position. (b) Position estimation error versus filter uncertainty covariance.

found for the three images shown in Figs. 7(a)–(c); however, only a few of them are explicitly shown in Figs. 7(d)–(f), while the rest of the matches are denoted by various markers.

The localization results are shown in Fig. 8. Figure 8(a) presents the estimated position compared with the true position. In addition, inertial-navigation-based position estimation is shown for comparison. Figure 8(b) depicts the position estimation errors (computed by subtracting the true position from the estimated position) and the square root of the filter covariance. The update mode is presented in both figures until $t \approx 150$ s sequential updates were performed, while loop updates were applied after the platform has completed a loop, starting from $t \approx 158$ s.

During the sequential updates phase, the time instances (t_1, t_2, t_3) were chosen such that $t_2 - t_1 \approx 1$ s and $t_3 - t_2 \approx 5$ s. As seen in Fig. 8, while sequential updates are active, the position is estimated with an accuracy of several meters, whereas the inertial solution rapidly diverges. Increasing the filter's frequency, given that the appropriate triplets of images are available, is expected to reduce the position error even further. The consistent behavior of the filter covariance indicates that the correlation between $\mathbf{X}(t_3)$ and $\mathbf{X}(t_2)$, which is not accounted for in the current filter formulation (Section IV), is not significant.

Although the position error is significantly reduced during the sequential updates of the algorithm (until $t \approx 120$ s), its development is mitigated during this phase but not entirely eliminated, as clearly evident in the height error. Two main reasons for this phenomenon are: 1) imperfect estimation of the actual IMU errors, and 2) in each update, the algorithm allows reducing current position errors only to the level of errors that were present while the first two images of the three were taken. Because each update in the sequential mode uses a different set of three images, and because the development of inertial error between these images, the error—although considerably mitigated—will continue to develop.

After the vehicle had completed its first loop, it became possible to apply the algorithm in a "loop update" mode. As seen in Fig. 8, the loop updates were applied at a varying frequency, which was typically lower than the frequency of sequential updates. Referring to Fig. 6, the vehicle completed its first loop at $t \approx 158$ s and performed the same trajectory once again, completing the second loop at $t \approx 230$ and afterwards continuing the same basic trajectory for another 10 s. In these last 10 s the vehicle began performing a third loop.

Each loop update significantly reduces the inertially-accumulated position error, yielding a small error of several meters after over 150 s of operation. For comparison, the inertial error approaches 1100 m (in the North axis) over this period of time, indicating the low quality of the inertial sensors. Note that the position error is reduced in all axes, including along the motion direction, which is not possible in two-view methods for navigation aiding [2].

As seen in Fig. 8, although each loop update drastically reduces the developed position error, the rate of the inertially-developing position error between each two loop updates has not been arrested compared with the pure inertial scenario (Fig. 8(a)), leading to the conclusion that the IMU errors parametrization (drift and bias) were not estimated well in the experiment.

Note also that as additional loop updates are applied and until reaching $t \approx 230$ s, the update accuracy deteriorates. For example, the East position error is reduced to -1.5 m at the first loop update ($t = 158$ s), while in the loop update at $t = 201$ s the East position error was reduced only to 6 m. The reason for this accuracy deterioration is that each loop update is performed using the current image and two images of the same scene that had been captured while the platform visited the area for the first time. As already mentioned, each update allows to reduce the current position error to the level of errors that were present while the first two images were captured. However, as can be seen from Fig. 8, the position error in the sequential updates phase, although considerably arrested, gradually increases over time, and hence the loop updates are capable of reducing the position error to the level of errors that increase with time. For example, the first two images participating in the first loop update at $t = 158$ s were captured at $t = 77$ and $t = 78$ s, while the first two images participating in the loop update at $t = 201$ s were captured at $t = 131$ and $t = 132$ s. Since the position error at $t = 131$ and $t = 132$ s was larger than the position error at $t = 77$ s and $t = 78$ s (Fig. 8(b)), the position error after the loop update at $t = 201$ s was accordingly larger than the position error after the first loop update (at $t = 158$ s).

After $t \approx 230$ s, the platform began its third loop and thus the loop updates from $t \approx 230$ s and on were performed using images (and the attached navigation data) captured at the beginning of the platform's trajectory (around $t = 80$ s). Therefore, the obtained position error at these loop updates is of accuracy comparable to the accuracy of the first loop updates (starting from $t = 158$ s), and hence to the accuracy of the navigation solution calculated in the beginning of the trajectory.

Analyzing the experiment results, it is also tempting to compare the performance obtained in the sequential and loop update modes. However, because these two modes of algorithm activation were not applied in the same phase, a quantitative analysis cannot be performed. Nevertheless, regardless of the sequential update mode, it is safe to state that activation of the algorithm in a loop update mode reduces the position errors in all axes to prior values while processing only three images.

VI. CONCLUSIONS

This paper presented a new method for vision-aided navigation based on three-view geometry. Camera-captured images were stored and designated by partial navigation data taken from the INS. These images were used for constructing a representation of the observed environment, while some of them were also incorporated for navigation aiding. The proposed method utilized three overlapping images to formulate constraints relating between the platform motion at the time instances of the three images. The associated navigation data for each of the three images allowed to determine the scale ambiguity inherent to all pure computer vision techniques for motion estimation. The constraints were further reformulated and fused with an INS using an IEKF. A single activation of the method over a set of three overlapping images reduces the inertially developed position errors to the levels present while the first two images were captured.

The developed method for vision-aided navigation may be used in various applications in which three overlapping images, and the required navigation data, are available. In this paper the method was applied to maintaining small navigation errors, while operating in a GPS-denied environment, accomplished by engaging the algorithm over sequential overlapping imagery, and utilizing the overlapping images in case a loop in the trajectory occurs. In contrast to the existing methods for vision-aided navigation, which are also capable of handling loops, such as bundle adjustment and SLAM, the computational requirements of the proposed algorithm allow real-time navigation aiding, since a constant-size state vector is used, and only three images are processed at each update step of the IEKF. The refinement process of the environment representation, such as mosaic image construction, may be performed in a background process.

The method was examined based on real imagery and navigation data, obtained in an experiment, and in a statistical study using simulated navigation and synthetic imagery. The results showed that reduced position and velocity errors can be maintained over time, thus allowing operation without relying on the GPS signal. Specifically, the position errors obtained in the experiment, in which a low-grade IMU was used, were reduced to several meters each time the algorithm was applied, while the inertial position error has reached over 1000 m in 150 s of operation. The implication of this result is important for various applications, in which the GPS signal is unavailable or unreliable. Among these is a holding pattern mission, in which the platform has to perform the same loop trajectory numerous times. Satellite orbit determination is another possible application.

APPENDIX I. PROOF OF THEOREM 1

Recall the matrix A,

$$A = \begin{bmatrix} \mathbf{q}_1 & -\mathbf{q}_2 & \mathbf{0}_{3\times 1} & -\mathbf{T}_{12} \\ \mathbf{0}_{3\times 1} & \mathbf{q}_2 & -\mathbf{q}_3 & -\mathbf{T}_{23} \end{bmatrix} \in \mathbb{R}^{6\times 4} \quad (39)$$

and the constraints

$$\mathbf{q}_1^T(\mathbf{T}_{12}\times\mathbf{q}_2) = 0 \quad (40)$$

$$\mathbf{q}_2^T(\mathbf{T}_{23}\times\mathbf{q}_3) = 0 \quad (41)$$

$$(\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{T}_{23}) = (\mathbf{q}_1\times\mathbf{T}_{12})^T(\mathbf{q}_3\times\mathbf{q}_2). \quad (42)$$

Next we prove that the constraints (40)–(42) hold if and only if $\text{rank}(A) < 4$.

A. $\text{rank}(A) < 4 \Rightarrow$ Equations (40)–(42)

Since $\text{rank}(A) < 4$, there exists a non-zero vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)^T$ such that $A\boldsymbol{\beta} = \mathbf{0}$. The explicit equations stemming from $A\boldsymbol{\beta} = \mathbf{0}$ are

$$\mathbf{q}_1\beta_1 - \mathbf{q}_2\beta_2 - \mathbf{T}_{12}\beta_4 = \mathbf{0} \quad (43)$$

$$\mathbf{q}_2\beta_2 - \mathbf{q}_3\beta_3 - \mathbf{T}_{23}\beta_4 = \mathbf{0}. \quad (44)$$

Cross-multiplying (43) by $\mathbf{q}_1$ and (44) by $\mathbf{q}_3$ yields

$$(\mathbf{q}_1\times\mathbf{q}_2)\beta_2 + (\mathbf{q}_1\times\mathbf{T}_{12})\beta_4 = \mathbf{0} \quad (45)$$

$$(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 - (\mathbf{q}_3\times\mathbf{T}_{23})\beta_4 = \mathbf{0}. \quad (46)$$

If $\mathbf{q}_1\times\mathbf{q}_2 \neq \mathbf{0}$ and $\mathbf{q}_3\times\mathbf{q}_2 \neq \mathbf{0}$, then performing an inner product of (45) with $(\mathbf{q}_3\times\mathbf{q}_2)$ and of (46) with $(\mathbf{q}_1\times\mathbf{q}_2)$ yields

$$(\mathbf{q}_3\times\mathbf{q}_2)^T(\mathbf{q}_1\times\mathbf{q}_2)\beta_2 + (\mathbf{q}_3\times\mathbf{q}_2)^T(\mathbf{q}_1\times\mathbf{T}_{12})\beta_4 = 0 \quad (47)$$

$$(\mathbf{q}_1\times\mathbf{q}_2)^T(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 - (\mathbf{q}_1\times\mathbf{q}_2)^T(\mathbf{q}_3\times\mathbf{T}_{23})\beta_4 = 0. \quad (48)$$

Noting that $(\mathbf{q}_3\times\mathbf{q}_2)^T(\mathbf{q}_1\times\mathbf{q}_2) = -(\mathbf{q}_1\times\mathbf{q}_2)^T(\mathbf{q}_3\times\mathbf{q}_2)$ and adding (47) and (48) gives the constraint (42).

The first two constraints may be obtained similarly: cross-multiplying (43) by $\mathbf{q}_2$ and then taking an inner product with $\mathbf{q}_1$ gives the constraint (40). Cross-multiplying from the right (44) by $\mathbf{q}_3$ and then taking an inner product with $\mathbf{q}_2$ gives the constraint (41).

Degenerate Cases: $\mathbf{q}_1\times\mathbf{q}_2 = \mathbf{0}$ or $\mathbf{q}_3\times\mathbf{q}_2 = \mathbf{0}$, or both, i.e., $\mathbf{q}_1\|\mathbf{q}_2$ or $\mathbf{q}_2\|\mathbf{q}_3$, or $\mathbf{q}_1\|\mathbf{q}_2\|\mathbf{q}_3$. Consider the case $\mathbf{q}_1\|\mathbf{q}_2$. Since both $\mathbf{q}_1$ and $\mathbf{q}_2$ point to the same ground point, it may be concluded that $\mathbf{T}_{12}$ is parallel to $\mathbf{q}_1$ and $\mathbf{q}_2$. More formally, if r_1 and r_2 are the scale parameters such that $\|r_i\mathbf{q}_i\|$ is the range to the ground point, then $\mathbf{T}_{12} = r_2\mathbf{q}_2 - r_1\mathbf{q}_1 = r_2 a\mathbf{q}_1 - r_1\mathbf{q}_1 = (r_2 a - r_1)\mathbf{q}_1$, where a is a constant. Hence $\mathbf{T}_{12}\|\mathbf{q}_1\|\mathbf{q}_2$. Consequently, (41) is the only constraint from the three constraints in (40)–(42) that is not degenerate. This constraint may be obtained as explained above. The case $\mathbf{q}_2\|\mathbf{q}_3$ is handled in a similar manner.

The last degenerated case is $\mathbf{q}_1\|\mathbf{q}_2\|\mathbf{q}_3$, which occurs when the vehicle moves along the LOS vectors. In this case all the constraints in (40)–(42) are degenerate.

Note that in the first two degenerate cases ($\mathbf{q}_1\|\mathbf{q}_2$ or $\mathbf{q}_2\|\mathbf{q}_3$), it is possible to write another set of three constraints. For example, if $\mathbf{q}_1\|\mathbf{q}_2$ (but not to $\mathbf{q}_3$), we can formulate two epipolar constraints between views 1 and 3, and between views 2 and 3, and provide the equivalent constraint to (42) relating between $\mathbf{T}_{13}$ and $\mathbf{T}_{23}$.

B. Equations (40)–(42) $\Rightarrow \text{rank}(A) < 4$

The proof is based on steps similar to the previous section, in a reverse order. Recall the constraint (42), multiplied by some constant $\beta_4 \neq 0$:

$$(\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{T}_{23})\beta_4 = (\mathbf{q}_1\times\mathbf{T}_{12})^T(\mathbf{q}_3\times\mathbf{q}_2)\beta_4. \quad (49)$$

Since $(\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{q}_2)$ is a scalar and (49) is a scalar equation, there exists some $\beta_2 \neq 0$ such that

$$(\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 = (\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{T}_{23})\beta_4$$

$$(\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 = (\mathbf{q}_1\times\mathbf{T}_{12})^T(\mathbf{q}_3\times\mathbf{q}_2)\beta_4.$$

The above equation may be rewritten into

$$(\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 - (\mathbf{q}_2\times\mathbf{q}_1)^T(\mathbf{q}_3\times\mathbf{T}_{23})\beta_4 = 0 \quad (50)$$

$$(\mathbf{q}_3\times\mathbf{q}_2)^T(\mathbf{q}_1\times\mathbf{q}_2)\beta_2 + (\mathbf{q}_3\times\mathbf{q}_2)^T(\mathbf{q}_1\times\mathbf{T}_{12})\beta_4 = 0 \quad (51)$$

or equivalently

$$(\mathbf{q}_2\times\mathbf{q}_1)^T[(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 - (\mathbf{q}_3\times\mathbf{T}_{23})\beta_4] = 0 \quad (52)$$

$$(\mathbf{q}_3\times\mathbf{q}_2)^T[(\mathbf{q}_1\times\mathbf{q}_2)\beta_2 + (\mathbf{q}_1\times\mathbf{T}_{12})\beta_4] = 0. \quad (53)$$

At this point it is assumed that $\mathbf{q}_1\times\mathbf{q}_2 \neq \mathbf{0}$ and $\mathbf{q}_3\times\mathbf{q}_2 \neq \mathbf{0}$. The proof for cases in which this assumption does not hold is given in the sequel.

Noting that $\mathbf{q}_2^T(\mathbf{q}_3\times\mathbf{q}_2) \equiv 0$, and since the constraint (41) is satisfied, the vectors $(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 - (\mathbf{q}_3\times\mathbf{T}_{23})\beta_4$ and $(\mathbf{q}_2\times\mathbf{q}_1)$ are not perpendicular. In the same manner, since $\mathbf{q}_2^T(\mathbf{q}_1\times\mathbf{q}_2) = 0$ and the constraint (40) is met, the vectors $(\mathbf{q}_1\times\mathbf{q}_2)\beta_2 + (\mathbf{q}_1\times\mathbf{T}_{12})\beta_4$ and $(\mathbf{q}_3\times\mathbf{q}_2)$ are not perpendicular as well. Therefore the last two equations lead to

$$(\mathbf{q}_3\times\mathbf{q}_2)\beta_2 - (\mathbf{q}_3\times\mathbf{T}_{23})\beta_4 = \mathbf{0} \quad (54)$$

$$(\mathbf{q}_1\times\mathbf{q}_2)\beta_2 + (\mathbf{q}_1\times\mathbf{T}_{12})\beta_4 = \mathbf{0} \quad (55)$$

that may by rewritten as

$$\mathbf{q}_3\times(\mathbf{q}_2\beta_2 - \mathbf{T}_{23}\beta_4 + \mathbf{q}_3\beta_3) = \mathbf{0} \quad (56)$$

$$\mathbf{q}_1\times(\mathbf{q}_2\beta_2 + \mathbf{T}_{12}\beta_4 + \mathbf{q}_1\beta_1) = \mathbf{0} \quad (57)$$

for any β_1, β_3. Consequently,

$$\mathbf{q}_2\beta_2 + \mathbf{q}_3\beta_3 - \mathbf{T}_{23}\beta_4 = \mathbf{0} \tag{58}$$

$$\mathbf{q}_1\beta_1 + \mathbf{q}_2\beta_2 + \mathbf{T}_{12}\beta_4 = \mathbf{0}. \tag{59}$$

In order to obtain the same expression for the matrix A, the vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3, \alpha_4)^T$ is defined as

$$\begin{aligned} \alpha_1 &\doteq \beta_1, \qquad \alpha_2 \doteq -\beta_2 \\ \alpha_3 &\doteq \beta_3, \qquad \alpha_4 \doteq -\beta_4 \end{aligned} \tag{60}$$

so that (58) and (59) turn into

$$-\mathbf{q}_2\alpha_2 + \mathbf{q}_3\alpha_3 + \mathbf{T}_{23}\alpha_4 = \mathbf{0} \tag{61}$$

$$\mathbf{q}_1\alpha_1 - \mathbf{q}_2\alpha_2 - \mathbf{T}_{12}\alpha_4 = \mathbf{0}. \tag{62}$$

The above may be rewritten as

$$A\boldsymbol{\alpha} = \mathbf{0} \tag{63}$$

and since $\boldsymbol{\alpha}$ is a non-zero vector, one may conclude that $\text{rank}(A) < 4$.

Note that the epipolar constraints (40) and (41) only guarantee that the matrices $[\mathbf{q}_1 \;\; -\mathbf{q}_2 \;\; -\mathbf{T}_{12}]$ and $[\mathbf{q}_2 \;\; -\mathbf{q}_3 \;\; -\mathbf{T}_{23}]$ are singular, which does not necessarily lead to $\text{rank}(A) < 4$.

Degenerate Cases: Next we prove that $\text{rank}(A) < 4$ also when $\mathbf{q}_1 \| \mathbf{q}_2$ or $\mathbf{q}_2 \| \mathbf{q}_3$, or $\mathbf{q}_1 \| \mathbf{q}_2 \| \mathbf{q}_3$.

Let $\mathbf{q}_1 \| \mathbf{q}_2$ while $\mathbf{q}_3$ is not parallel to $\mathbf{q}_1$. As proven above, $\mathbf{q}_1 \| \mathbf{q}_2 \| \mathbf{T}_{12}$, and thus, the matrix A is of the form

$$A = \begin{bmatrix} \mathbf{q}_1 & a\mathbf{q}_1 & \mathbf{0}_{3\times 1} & b\mathbf{q}_1 \\ \mathbf{0}_{3\times 1} & a\mathbf{q}_1 & \mathbf{q}_3 & \mathbf{T}_{23} \end{bmatrix} \tag{64}$$

for some scalars a, b. In order to prove that $\text{rank}(A) < 4$, we need to show that $A\boldsymbol{\beta} = \mathbf{0}$ for some non-zero vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)^T$. Assume a general vector $\boldsymbol{\beta}$ and explicitly write $A\boldsymbol{\beta} = 0$:

$$\mathbf{q}_1\beta_1 + a\mathbf{q}_1\beta_2 + b\mathbf{q}_1\beta_4 = 0 \tag{65}$$

$$a\mathbf{q}_1\beta_2 + \mathbf{q}_3\beta_3 + \mathbf{T}_{23}\beta_4 = 0. \tag{66}$$

Observe that the second equation leads to the epipolar constraint $\mathbf{q}_2^T(\mathbf{q}_3 \times \mathbf{T}_{23}) = 0$. Since the constraints (40)–(42) hold, it follows that the matrix $[\mathbf{q}_2 \;\; -\mathbf{q}_3 \;\; -\mathbf{T}_{23}]$ is singular, and since $\mathbf{q}_2 = a\mathbf{q}_1$, it is possible to find non-zero entries for β_2, β_3, and β_4 so that (66) is satisfied. From (65) it is easy to see that $\beta_1 = -a\beta_2 - b\beta_4$. Thus, a non-zero vector β was found such that $A\boldsymbol{\beta} = \mathbf{0}$, which leads to the conclusion that $\text{rank}(A) < 4$. A similar procedure may be applied when $\mathbf{q}_2 \| \mathbf{q}_3$ while $\mathbf{q}_1$ is not parallel to $\mathbf{q}_2$.

The last degenerate case is when all the three vectors are parallel. As already mentioned, both of the translation vectors in this case are parallel to the LOS vectors, i.e., $\mathbf{q}_1 \| \mathbf{q}_2 \| \mathbf{q}_3 \| \mathbf{T}_{12} \| \mathbf{T}_{23}$. The matrix A is then of the following form:

$$A = \begin{bmatrix} \mathbf{q}_1 & a\mathbf{q}_1 & \mathbf{0}_{3\times 1} & b\mathbf{q}_1 \\ \mathbf{0}_{3\times 1} & -a\mathbf{q}_1 & c\mathbf{q}_1 & d\mathbf{q}_1 \end{bmatrix} \tag{67}$$

where a, b, c, and d are some constants. Since one may find some non-zero vector $\boldsymbol{\beta}$ such that $A\boldsymbol{\beta} = \mathbf{0}$, (e.g. $\boldsymbol{\beta} = (b, 0, d/c, -1)^T$), the conclusion is that $\text{rank}(A) < 4$.

APPENDIX II. IEKF MATRICES

In this Appendix we present the development of the IEKF matrices H_3, H_2, H_1, D and R. Recall the residual measurement definition (see (23), (24), and (26))

$$\begin{aligned} \mathbf{z} &= \mathbf{h}(\mathbf{Pos}(t_3), \boldsymbol{\Psi}(t_3), \mathbf{Pos}(t_2), \boldsymbol{\Psi}(t_2), \mathbf{Pos}(t_1), \\ &\quad \boldsymbol{\Psi}(t_1), \{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}) = \mathcal{A}\mathbf{T}_{23} - \mathcal{B}\mathbf{T}_{12} \\ &\approx H_3\mathbf{X}(t_3) + H_2\mathbf{X}(t_2) + H_1\mathbf{X}(t_1) + D\mathbf{v} \end{aligned} \tag{68}$$

where

$$\mathcal{A} \doteq \begin{bmatrix} U \\ F \\ 0 \end{bmatrix}_{N\times 3}, \qquad \mathcal{B} \doteq \begin{bmatrix} W \\ 0 \\ G \end{bmatrix}_{N\times 3} \tag{69}$$

and $\mathbf{X}$ is the state vector defined in (25):

$$\mathbf{X}_{15\times 1} = [\Delta\mathbf{P}^T \quad \Delta\mathbf{V}^T \quad \Delta\boldsymbol{\Psi}^T \quad \mathbf{d}^T \quad \mathbf{b}^T]^T. \tag{70}$$

Recall also that the time instant of the third image t_3 of the three overlapping images is the current time. Therefore, in (68) $\mathbf{X}(t_3)$ is the state vector to be estimated, while $\tilde{\mathbf{X}}(t_2) = \mathbf{X}(t_2)$ and $\tilde{\mathbf{X}}(t_1) = \mathbf{X}(t_1)$ are the estimation errors at the first two time instances represented by the filter covariance attached to each image. These last two terms, accompanied by the Jacobian matrices H_2 and H_1 and the image noise $\mathbf{v}$ along with the Jacobian matrix D, constitute the measurement noise. Since navigation and imagery information is independent of each other, these two sources of information will be analyzed separately.

C. Calculation of the Matrices H_3, H_2 and H_1

The matrices H_3, H_2, and H_1, are $N \times 15$ and are defined as

$$H_3 \doteq \nabla_{\zeta(t_3)}\mathbf{h}, \qquad H_2 \doteq \nabla_{\zeta(t_2)}\mathbf{h}, \qquad H_1 \doteq \nabla_{\zeta(t_1)}\mathbf{h} \tag{71}$$

where $\boldsymbol{\zeta}$ is defined in (28).

From (68) it is clear that these matrices are of the following form:

$$H_i = [H^{\mathbf{Pos}(t_i)} \quad 0 \quad H^{\boldsymbol{\Psi}(t_i)} \quad 0 \quad 0] \tag{72}$$

with $i = 1, 2, 3$. Since $\mathbf{T}_{23} = \mathbf{Pos}(t_3) - \mathbf{Pos}(t_2)$ and $\mathbf{T}_{12} = \mathbf{Pos}(t_2) - \mathbf{Pos}(t_1)$,

$$H^{\mathbf{Pos}(t_3)} = \mathcal{A} \tag{73}$$

$$H^{\mathbf{Pos}(t_2)} = -(\mathcal{A} + \mathcal{B}) \tag{74}$$

$$H^{\mathbf{Pos}(t_1)} = \mathcal{B}. \tag{75}$$

Note that the influence of position errors on the LOS vectors that appear in the matrices $\mathcal{A}$ and $\mathcal{B}$ is

neglected: the position errors affect only the rotation matrices transforming the LOS vectors to the LLLN system at t_2. These errors are divided by the Earth radius, and therefore their contribution is insignificant.

Calculation of $H^{\Psi(t_3)}$, $H^{\Psi(t_2)}$ and $H^{\Psi(t_1)}$: Recall the definition of the matrices F, G, U, and W

$$U = [\mathbf{u}_1 \cdots \mathbf{u}_{N_{123}}]^T \tag{76}$$

$$F = [\mathbf{f}_1 \cdots \mathbf{f}_{N_{23}}]^T \tag{77}$$

$$W = [\mathbf{w}_1 \cdots \mathbf{w}_{N_{123}}]^T \tag{78}$$

$$G = [\mathbf{g}_1 \cdots \mathbf{g}_{N_{12}}]^T \tag{79}$$

with

$$\mathbf{u}_i = \mathbf{u}_i(\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}) = -[\mathbf{q}_{3_i}]_\times [\mathbf{q}_{1_i}]_\times \mathbf{q}_{2_i} \tag{80}$$

$$\mathbf{w}_i = \mathbf{w}_i(\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}) = -[\mathbf{q}_{1_i}]_\times [\mathbf{q}_{2_i}]_\times \mathbf{q}_{3_i} \tag{81}$$

$$\mathbf{f}_i = \mathbf{f}_i(\mathbf{q}_{2_i}, \mathbf{q}_{3_i}) = [\mathbf{q}_{2_i}]_\times \mathbf{q}_{3_i} \tag{82}$$

$$\mathbf{g}_i = \mathbf{g}_i(\mathbf{q}_{1_i}, \mathbf{q}_{2_i}) = [\mathbf{q}_{1_i}]_\times \mathbf{q}_{2_i}. \tag{83}$$

Since the development of expressions for the matrices $H^{\Psi(t_3)}$, $H^{\Psi(t_2)}$, and $H^{\Psi(t_1)}$ is similar, we elaborate only on the development process of $H^{\Psi(t_3)}$. This matrix is given by

$$H^{\Psi(t_3)} = \nabla_{\Psi(t_3)} \mathbf{h} = \nabla_{\Psi(t_3)} [\mathcal{A}\mathbf{T}_{23}] - \nabla_{\Psi(t_3)} [\mathcal{B}\mathbf{T}_{12}]. \tag{84}$$

We start by developing the first term in (84). According to the structure of the matrices U and F, the following may be written:

$$\nabla_{\Psi(t_3)} [\mathcal{A}\mathbf{T}_{23}] = \sum_{i=1}^{N_{123}} \frac{\partial \mathcal{A}\mathbf{T}_{23}}{\partial \mathbf{u}_i} \nabla_{\Psi(t_3)} \mathbf{u}_i + \sum_{i=1}^{N_{23}} \frac{\partial \mathcal{A}\mathbf{T}_{23}}{\partial \mathbf{f}_i} \nabla_{\Psi(t_3)} \mathbf{f}_i. \tag{85}$$

Since $\mathbf{u}_i$ and $\mathbf{f}_i$ are independent of each other, and $\partial \mathbf{x}^T \mathbf{T}_{23} / \partial \mathbf{x}_i = \mathbf{T}_{23}^T$ for any vector $\mathbf{x}$, we have

$$\frac{\partial \mathcal{A}\mathbf{T}_{23}}{\partial \mathbf{u}_i} = \mathbf{e}_i \mathbf{T}_{23}^T \tag{86}$$

$$\frac{\partial \mathcal{A}\mathbf{T}_{23}}{\partial \mathbf{f}_i} = \mathbf{e}_{N_{123}+i} \mathbf{T}_{23}^T \tag{87}$$

where $\mathbf{e}_j$ is an $N \times 1$ vector that is comprises zero entries except for the jth element which is equal to one. Note also that the size of the matrices $\partial \mathcal{A}\mathbf{T}_{23} / \partial \mathbf{u}_i$, $\partial \mathcal{A}\mathbf{T}_{23} / \partial \mathbf{f}_i$ is $N \times 3$. The remaining quantities in (85), $\nabla_{\Psi(t_3)} \mathbf{u}_i$ and $\nabla_{\Psi(t_3)} \mathbf{f}_i$, can be calculated as

$$\nabla_{\Psi(t_3)} \mathbf{u}_i = \frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{3_i}} \nabla_{\Psi(t_3)} \mathbf{q}_{3_i} \tag{88}$$

$$\nabla_{\Psi(t_3)} \mathbf{f}_i = \frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{3_i}} \nabla_{\Psi(t_3)} \mathbf{q}_{3_i} \tag{89}$$

here $\mathbf{q}_{3_i}$ refers to the LOS vector of the ith feature in the third view.

Analytical expressions for $\partial \mathbf{f} / \partial \mathbf{q}_j$, $\partial \mathbf{g} / \partial \mathbf{q}_j$, $\partial \mathbf{u} / \partial \mathbf{q}_j$, $\partial \mathbf{w} / \partial \mathbf{q}_j$, for $j = 1, 2, 3$, are easily obtained based on (80)–(83) as

$$\begin{aligned}
&\frac{\partial \mathbf{u}}{\partial \mathbf{q}_1} = [\mathbf{q}_3]_\times [\mathbf{q}_2]_\times, \qquad \frac{\partial \mathbf{w}}{\partial \mathbf{q}_1} = [[\mathbf{q}_2]_\times \mathbf{q}_3]_\times \\
&\frac{\partial \mathbf{f}}{\partial \mathbf{q}_1} = 0_{3\times 3}, \qquad \frac{\partial \mathbf{g}}{\partial \mathbf{q}_1} = -[\mathbf{q}_2]_\times \\
&\frac{\partial \mathbf{u}}{\partial \mathbf{q}_2} = -[\mathbf{q}_3]_\times [\mathbf{q}_1]_\times, \qquad \frac{\partial \mathbf{w}}{\partial \mathbf{q}_2} = [\mathbf{q}_1]_\times [\mathbf{q}_3]_\times \\
&\frac{\partial \mathbf{f}}{\partial \mathbf{q}_2} = -[\mathbf{q}_3]_\times, \qquad \frac{\partial \mathbf{g}}{\partial \mathbf{q}_2} = [\mathbf{q}_1]_\times \\
&\frac{\partial \mathbf{u}}{\partial \mathbf{q}_3} = [[\mathbf{q}_1]_\times \mathbf{q}_2]_\times, \qquad \frac{\partial \mathbf{w}}{\partial \mathbf{q}_3} = -[\mathbf{q}_1]_\times [\mathbf{q}_2]_\times \\
&\frac{\partial \mathbf{f}}{\partial \mathbf{q}_3} = [\mathbf{q}_2]_\times, \qquad \frac{\partial \mathbf{g}}{\partial \mathbf{q}_3} = 0_{3\times 3}.
\end{aligned} \tag{90}$$

As for $\nabla_{\Psi(t_3)} \mathbf{q}_3$, recall that the LOS vectors in $\mathbf{f}$, $\mathbf{g}$, $\mathbf{u}$, $\mathbf{w}$ are expressed in the LLLN system at t_2. Thus, for example, for some LOS vector from the first view

$$\begin{aligned}
\mathbf{q}_1 &= C_{L_2}^{C_1} \mathbf{q}_1^{C_1} = C_{L_2}^{L_1} C_{L_1}^{B_1} C_{B_1}^{C_1} \mathbf{q}_1^{C_1} \\
&= C_{L_2}^{L_1} [I + [\Delta\Psi_1]_\times] C_{L_{1,\text{True}}}^{B_1} C_{B_1}^{C_1} \mathbf{q}_1^{C_1} \\
&\approx \bar{\mathbf{q}}_1 - C_{L_2}^{L_1} [\mathbf{q}_1^{L_1}]_\times \Delta\Psi_1
\end{aligned} \tag{91}$$

here $\bar{\mathbf{q}}$ is the true value of $\mathbf{q}$. In a similar manner we get

$$\mathbf{q}_2 \approx \bar{\mathbf{q}}_2 - [\mathbf{q}_2^{L_2}]_\times \Delta\Psi_2 \tag{92}$$

$$\mathbf{q}_3 \approx \bar{\mathbf{q}}_3 - C_{L_2}^{L_3} [\mathbf{q}_3^{L_3}]_\times \Delta\Psi_3. \tag{93}$$

Consequently,

$$\nabla_{\Psi(t_1)} \mathbf{q}_1 = -C_{L_2}^{L_1} [\mathbf{q}_1^{L_1}]_\times \tag{94}$$

$$\nabla_{\Psi(t_2)} \mathbf{q}_2 = -[\mathbf{q}_2^{L_2}]_\times \tag{95}$$

$$\nabla_{\Psi(t_3)} \mathbf{q}_3 = -C_{L_2}^{L_3} [\mathbf{q}_3^{L_3}]_\times. \tag{96}$$

Incorporating all the above expressions, (85) turns into

$$\begin{aligned}
\nabla_{\Psi(t_3)} [\mathcal{A}\mathbf{T}_{23}] = &- \sum_{i=1}^{N_{123}} \mathbf{e}_i \mathbf{T}_{23}^T \frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{3_i}} C_{L_2}^{L_3} [\mathbf{q}_{3_i}^{L_3}]_\times \\
&- \sum_{i=1}^{N_{23}} \mathbf{e}_{N_{123}+i} \mathbf{T}_{23}^T \frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{3_i}} C_{L_2}^{L_3} [\mathbf{q}_{3_i}^{L_3}]_\times.
\end{aligned} \tag{97}$$

Noting that $\mathbf{g}$ is not a function of $\mathbf{q}_3$ and following a similar procedure we get

$$\nabla_{\Psi(t_3)} [\mathcal{B}\mathbf{T}_{12}] = - \sum_{i=1}^{N_{123}} \mathbf{e}_i \mathbf{T}_{12}^T \frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{3_i}} C_{L_2}^{L_3} [\mathbf{q}_{3_i}^{L_3}]_\times. \tag{98}$$

In conclusion, $H^{\Psi(t_3)}$ may be calculated according to

$$\begin{aligned}
H^{\Psi(t_3)}|_{N\times 3} = &\sum_{i=1}^{N_{123}} \mathbf{e}_i \left[\mathbf{T}_{12}^T \frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{3_i}} - \mathbf{T}_{23}^T \frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{3_i}} \right] C_{L_2}^{L_3} [\mathbf{q}_{3_i}^{L_3}]_\times \\
&- \sum_{i=1}^{N_{23}} \mathbf{e}_{N_{123}+i} \mathbf{T}_{23}^T \frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{3_i}} C_{L_2}^{L_3} [\mathbf{q}_{3_i}^{L_3}]_\times.
\end{aligned} \tag{99}$$

Applying the same technique, the matrices $H^{\Psi(t_2)}$ and $H^{\Psi(t_1)}$ were obtained as

$$H^{\Psi(t_2)} = \sum_{i=1}^{N_{123}} \mathbf{e}_i \left[\mathbf{T}_{12}^T \frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{2_i}} - \mathbf{T}_{23}^T \frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{2_i}} \right] [\mathbf{q}_{2_i}^{L_2}]_\times - \sum_{i=1}^{N_{23}} \mathbf{e}_{N_{123}+i} \mathbf{T}_{23}^T \frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{2_i}} [\mathbf{q}_{2_i}^{L_2}]_\times + \sum_{i=1}^{N_{12}} \mathbf{e}_{N_{123}+N_{23}+i} \mathbf{T}_{12}^T \frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{2_i}} [\mathbf{q}_{2_i}^{L_2}]_\times \quad (100)$$

$$H^{\Psi(t_1)} = \sum_{i=1}^{N_{123}} \mathbf{e}_i \left[\mathbf{T}_{12}^T \frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{1_i}} - \mathbf{T}_{23}^T \frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{1_i}} \right] C_{L_2}^{L_1} [\mathbf{q}_{1_i}^{L_1}]_\times + \sum_{i=1}^{N_{12}} \mathbf{e}_{N_{123}+N_{23}+i} \mathbf{T}_{12}^T \frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{1_i}} C_{L_2}^{L_1} [\mathbf{q}_{1_i}^{L_1}]_\times. \quad (101)$$

D. Calculation of the Matrices D and R

The matrices D and R are given by

$$\mathcal{D} \doteq \nabla_{\{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}} \mathbf{h} \quad (102)$$

$$R \doteq \mathrm{cov}(\{\mathbf{q}_{1_i}^{C_1}, \mathbf{q}_{2_i}^{C_2}, \mathbf{q}_{3_i}^{C_3}\}). \quad (103)$$

D reflects the influence of image noise on the measurement $\mathbf{z}$, while R is the image noise covariance for each matching LOS vector in the given images. Assuming that the camera optical axis lies along the z direction, a general LOS vector is contaminated by image noise $\mathbf{v} = (v_x, v_y)^T$, according to

$$\mathbf{q}^C = \bar{\mathbf{q}}^C + (v_x \quad v_y \quad 0)^T \quad (104)$$

where $\bar{\mathbf{q}}^C$ is the true value of the LOS vector, without noise contamination. Note that thus far we have omitted the explicit notation in the LOS vectors, thereby assuming that all the vectors are given in the LLLN system of t_2.

Recall that the sets of matching triplets and matching pairs

$$\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}\}_{i=1}^{N_{12}}, \qquad \{\mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{23}}, \qquad \{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{123}} \quad (105)$$

were assumed to be consistent (see Section III-A). Thus, for example, the matrices U and F, which are part of the matrix $\mathcal{A}$, are comprised of

$$U = [\mathbf{u}_1 \cdots \mathbf{u}_{N_{123}}]^T, \qquad F = [\mathbf{f}_1 \cdots \mathbf{f}_{N_{23}}]^T \quad (106)$$

with $\mathbf{u}_i$ and $\mathbf{f}_i$, constructed using the same LOS vectors, for $i \le N_{123}$. We define ΔN_{12} and ΔN_{23} as the number of additional pairs in $\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}\}_{i=1}^{N_{12}}$ and $\{\mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{23}}$ that are not present in $\{\mathbf{q}_{1_i}, \mathbf{q}_{2_i}, \mathbf{q}_{3_i}\}_{i=1}^{N_{123}}$: $N_{12} = N_{123} + \Delta N_{12}$ and $N_{23} = N_{123} + \Delta N_{23}$. Although the overall number of matches in the above sets (105) is $N = N_{123} + N_{12} + N_{23}$, the actual number of different matches is $N_{123} + \Delta N_{12} + \Delta N_{23}$.

Assuming that the covariance of the image noise is the same for all the LOS vectors in the three images, and recalling the structure of the matrices $\mathcal{A}, \mathcal{B}$ that are used for calculating $\mathbf{h}$, we can write

$$DRD^T = \sum_{i=1}^{N_{123}+\Delta N_{12}} \frac{\partial \mathbf{h}}{\partial \mathbf{q}_{1_i}^{C_1}} R_{\mathbf{v}} \frac{\partial \mathbf{h}^T}{\partial \mathbf{q}_{1_i}^{C_1}} + \sum_{i=1}^{N_{123}+\Delta N_{12}+\Delta N_{23}} \frac{\partial \mathbf{h}}{\partial \mathbf{q}_{2_i}^{C_2}} R_{\mathbf{v}} \frac{\partial \mathbf{h}^T}{\partial \mathbf{q}_{2_i}^{C_2}} + \sum_{i=1}^{N_{123}+\Delta N_{23}} \frac{\partial \mathbf{h}}{\partial \mathbf{q}_{3_i}^{C_3}} R_{\mathbf{v}} \frac{\partial \mathbf{h}^T}{\partial \mathbf{q}_{3_i}^{C_3}}. \quad (107)$$

In the above equation, each summation refers to all the LOS vectors from the relevant image that participate in the calculation of $\mathbf{h}$. For example, the first summation refers to the first image. $R_{\mathbf{v}}$ is a 3×3 covariance matrix of the image noise

$$R_{\mathbf{v}} = \begin{bmatrix} R_x & 0 & 0 \\ 0 & R_y & 0 \\ 0 & 0 & R_f \end{bmatrix} \quad (108)$$

with $R_x = E(v_x v_x^T)$ and $R_y = E(v_y v_y^T)$. R_f represents the uncertainty in the camera focal length. Assuming the focal length is known precisely, it can be chosen as zero.

Next we develop expressions for $\partial \mathbf{h} / \partial \mathbf{q}_k^{C_k}$ for each image (i.e., $k = 1,2,3$). We begin with $\partial \mathbf{h} / \partial \mathbf{q}_1^{C_1}$

$$\left. \frac{\partial \mathbf{h}}{\partial \mathbf{q}_{1_i}^{C_1}} \right|_{N \times 3} = \frac{\partial \mathcal{A} \mathbf{T}_{23}}{\partial \mathbf{q}_{1_i}^{C_1}} - \frac{\partial \mathcal{B} \mathbf{T}_{12}}{\partial \mathbf{q}_{1_i}^{C_1}}. \quad (109)$$

Since the matrices U, W and G contain LOS vectors from the first view while the matrix F does not, the above equals to

$$\frac{\partial \mathbf{h}}{\partial \mathbf{q}_{1_i}^{C_1}} = \sum_{k=1}^{N_{123}} \frac{\partial \mathcal{A} \mathbf{T}_{23}}{\partial \mathbf{u}_k} \frac{\partial \mathbf{u}_k}{\partial \mathbf{q}_{1_i}^{L_2}} \frac{\partial \mathbf{q}_{1_i}^{L_2}}{\partial \mathbf{q}_{1_i}^{C_1}} - \sum_{k=1}^{N_{123}} \frac{\partial \mathcal{B} \mathbf{T}_{12}}{\partial \mathbf{w}_k} \frac{\partial \mathbf{w}_k}{\partial \mathbf{q}_{1_i}^{L_2}} \frac{\partial \mathbf{q}_{1_i}^{L_2}}{\partial \mathbf{q}_{1_i}^{C_1}} - \sum_{k=1}^{N_{12}} \frac{\partial \mathcal{B} \mathbf{T}_{12}}{\partial \mathbf{g}_k} \frac{\partial \mathbf{g}_k}{\partial \mathbf{q}_{1_i}^{L_2}} \frac{\partial \mathbf{q}_{1_i}^{L_2}}{\partial \mathbf{q}_{1_i}^{C_1}}. \quad (110)$$

Noting that $\forall i \neq k$, $\partial \mathbf{u}_k / \partial \mathbf{q}_{1_i}^{L_2} = \partial \mathbf{w}_k / \partial \mathbf{q}_{1_i}^{L_2} = \partial \mathbf{g}_k / \partial \mathbf{q}_{1_i}^{L_2} = 0$, and taking into account that $\partial \mathbf{q}_{1_i}^{L_2} / \partial \mathbf{q}_{1_i}^{C_1} = C_{L_2}^{C_1}$, the above turns into (111), where the derivatives $\partial \mathbf{u}_i / \partial \mathbf{q}_{1_i}^{L_2}$, $\partial \mathbf{w}_i / \partial \mathbf{q}_{1_i}^{L_2}$, and $\partial \mathbf{g}_i / \partial \mathbf{q}_{1_i}^{L_2}$ were already computed (see (90)). Using the same procedure we obtain expressions for the $N \times 3$ matrices $\partial \mathbf{h} / \partial \mathbf{q}_{2_i}^{C_2}$ and $\partial \mathbf{h} / \partial \mathbf{q}_{3_i}^{C_3}$, which are given in (112) and (113).

$$\left.\frac{\partial \mathbf{h}}{\partial \mathbf{q}_{1_i}^{C_1}}\right|_{N\times 3} = \left[\frac{\partial \mathcal{A}\mathbf{T}_{23}}{\partial \mathbf{u}_i}\frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{1_i}^{L_2}} - \frac{\partial \mathcal{B}\mathbf{T}_{12}}{\partial \mathbf{w}_i}\frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{1_i}^{L_2}} - \frac{\partial \mathcal{B}\mathbf{T}_{12}}{\partial \mathbf{g}_i}\frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{1_i}^{L_2}}\right]\frac{\partial \mathbf{q}_{1_i}^{L_2}}{\partial \mathbf{q}_{1_i}^{C_1}}$$

$$= \begin{cases} \left\{\mathbf{e}_i\left[\mathbf{T}_{23}^T\frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{1_i}^{L_2}} - \mathbf{T}_{12}^T\frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{1_i}^{L_2}}\right] - \mathbf{e}_{N_{123}+N_{23}+i}\mathbf{T}_{12}^T\frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{1_i}^{L_2}}\right\}C_{L_2}^{C_1} & i \le N_{123} \\ -\mathbf{e}_{N_{123}+N_{23}+i}\mathbf{T}_{12}^T\frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{1_i}^{L_2}}C_{L_2}^{C_1} & N_{123} < i \le N_{123}+\Delta N_{12} \end{cases} \tag{111}$$

$$\left.\frac{\partial \mathbf{h}}{\partial \mathbf{q}_{2_i}^{C_2}}\right|_{N\times 3} = \begin{cases} \mathbf{e}_i\left[\mathbf{T}_{23}^T\frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{2_i}^{L_2}} - \mathbf{T}_{12}^T\frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{2_i}^{L_2}}\right]C_{L_2}^{C_2} + \\ + \left[\mathbf{e}_{N_{123}+i}\mathbf{T}_{23}^T\frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{2_i}^{L_2}} - \mathbf{e}_{N_{123}+N_{23}+i}\mathbf{T}_{12}^T\frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{2_i}^{L_2}}\right]C_{L_2}^{C_2} & i \le N_{123} \\ \mathbf{e}_{N_{123}+i}\mathbf{T}_{23}^T\frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{2_i}^{L_2}}C_{L_2}^{C_2} & N_{123} < i \le N_{123}+\Delta N_{23} \\ -\mathbf{e}_{2N_{123}+i}\mathbf{T}_{12}^T\frac{\partial \mathbf{g}_i}{\partial \mathbf{q}_{2_i}^{L_2}}C_{L_2}^{C_2} & N_{123}+\Delta N_{23} < i \le N_{123}+\Delta N_{23}+\Delta N_{12} \end{cases} \tag{112}$$

$$\left.\frac{\partial \mathbf{h}}{\partial \mathbf{q}_{3_i}^{C_3}}\right|_{N\times 3} = \begin{cases} \left\{\mathbf{e}_i\left[\mathbf{T}_{23}^T\frac{\partial \mathbf{u}_i}{\partial \mathbf{q}_{3_i}^{L_2}} - \mathbf{T}_{12}^T\frac{\partial \mathbf{w}_i}{\partial \mathbf{q}_{3_i}^{L_2}}\right] + \mathbf{e}_{N_{123}+i}\mathbf{T}_{23}^T\frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{3_i}^{L_2}}\right\}C_{L_2}^{C_3} & i \le N_{123} \\ \mathbf{e}_{N_{123}+i}\mathbf{T}_{23}^T\frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_{3_i}^{L_2}}C_{L_2}^{C_3} & N_{123} < i \le N_{123}+\Delta N_{23} \end{cases}. \tag{113}$$

REFERENCES

[1] Hartley, R. and Zisserman, A.
Multiple View Geometry.
Cambridge University Press, 2000.

[2] Indelman, V., et al.
Navigation aiding based on coupled online mosaicking and camera scanning.
Journal of Guidance, Control and Dynamics, **33**, 6 (2011), 1866–1882.

[3] Indelman, V., et al.
Real-time mosaic-aided aircraft navigation: II. Sensor fusion.
Proceedings of the AIAA Guidance, Navigation and Control Conference, Chicago, IL, 2009.

[4] Eustice, R. M., Pizarro, O., and Singth, H.
Visually augmented navigation for autonomous underwater vehicles.
IEEE Journal of Oceanic Engineering, **33**, 2 (2008), 103–122.

[5] Caballero, F., et al.
Improving vision-based planar motion estimation for unmanned aerial vehicles through online mosaicing.
Proceedings of the IEEE International Conference on Robotics and Automation, Orlando, FL, May 2006, pp. 2860–2865.

[6] Gracias, N. and Santos-Victor, J.
Underwater video mosaics as visual navigation maps.
Computer Vision and Image Understanding, **79** (2000), 66–91.

[7] Mourikis, A. and Roumeliotis, I.
A multi-state constraint Kalman filter for vision-aided inertial navigation.
Proceedings of the IEEE International Conference on Robotics and Automation, Apr. 2007, pp. 3565–3572.

[8] Shakernia, O., et al.
Multiple view motion estimation and control for landing an unmanned aerial vehicle.
Proceedings of the IEEE International Conference on Robotics and Automation, Washington, D.C., May 2002, pp. 2793–2798.

[9] Mourikis, A. and Roumeliotis, I.
A dual-layer estimator architecture for long-term localization.
IEEE Computer Vision and Pattern Recognition Workshops, June 2008, pp. 1–8.

[10] Ma, Y., et al.
Rank conditions on the multiple-view matrix.
International Journal of Computer Vision, (May 2004), 115–137.

[11] Kim, J. and Sukkarieh, S.
6DoF SLAM aided GNSS/INS navigation in GNSS denied and unknown environments.
Journal of Global Positioning Systems, **4**, 1–2 (2005), 120–128.

[12] Garcia, R., et al.
Augmented state Kalman filtering for AUV navigation.
Proceedings of the IEEE International Conference Robotics and Automation, vol. 4, Washington, D.C., 2002, pp. 4010–4015.

[13] Davison, A. J., et al.
MonoSLAM: Real-time single camera SLAM.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **29**, 6 (2007), 1052–1067.

[14] Civera, J., Davison, A. J., and Montiel, J. M. M.
Inverse depth parametrization for monocular SLAM.
IEEE Transactions on Robotics, **24**, 5 (2008), 932–945.

[15] Bryson, M. and Sukkarieh, S.
Active airborne localization and exploration in unknown environments using inertial SLAM.
Proceedings of the IEEE Aerospace Conference, Big Sky, MT, 2006.

[16] Bryson, M. and Sukkarieh, S.
Bearing-only SLAM for an airborne vehicle.
Proceedings of the Australasian Conference on Robotics and Automation, Sydney, Australia, 2005.

[17] Eade, E. and Drummond, T.
Scalable monocular SLAM.
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, 2006, pp. 469–476.

[18] Fleischer, S., et al.
Improved real-time video mosaicking of the ocean floor.
Oceans Conference Proceedings, vol. 3, San Diego, CA, Oct. 1995, pp. 1935–1944.

[19] Caballero, F., et al.
Homography based Kalman filter for mosaic building. Applications to UAV position estimation.
Proceedings of the IEEE International Conference on Robotics and Automation, Rome, Apr. 2007, pp. 2004–2009.

[20] Fletcher, J., Veth, M., and Raquet, J.
Real time fusion of image and inertial sensors for navigation.
Proceedings of the 63rd Institute of Navigation Annual Meeting, Apr. 2007.

[21] Yu, Y. K., et al.
Recursive camera-motion estimation with the trifocal tensor.
IEEE Transactions on Systems, Man, And Cybernetics—Part B: Cybernetics, **36**, 5 (Oct. 2006), 1081–1090.

[22] Yu, Y. K., et al.
Robust 3-D motion tracking from stereo images: A model-less method.
IEEE Transactions on Instrumentation and Measurement, **57**, 3 (Mar. 2008), 622–630.

[23] Guerrero, J. J., Murillo, A. C., and Sagües, C.
Localization and matching using the planar trifocal tensor with bearing-only data.
IEEE Transactions on Robotics, **24**, 2 (Apr. 2008), 494–501.

[24] Soatto, S., Frezza, R., and Perona, P.
Motion estimation via dynamic vision.
IEEE Transactions on Automatic Control, **41**, 3 (Mar. 1996), 393–413.

[25] Goshen-Meskin, D. and Bar-Itzhack, I. Y.
Observability analysis of piece-wise constant systems—Part I: Theory.
IEEE Transactions on Aerospace and Electronic Systems, **28**, 4 (Oct. 1992), 1056–1067.

[26] Julier, S. J. and Uhlmann, J. K.
A non-divergent estimation algorithm in the presence of unknown correlations.
Proceedings of the American Control Conference, Albuquerque, NM, June 1997, pp. 2369–2373.

[27] Arambel, P. O., Rago, C., and Mehra, R. K.
Covariance intersection algorithm for distributed spacecraft state estimation.
Proceedings of the American Control Conference, Arlington, VA, June 2001, pp. 4398–4403.

[28] Bahr, A., Walter, M. R., and Leonard, J. J.
Consistent cooperative localization.
Proceedings of the IEEE International Conference on Robotics and Automation, Kobe, Japan, May 2009, pp. 3415–3422.

[29] Lowe, D.
Distinctive image features from scale-invariant keypoints.
International Journal of Computer Vision, **60**, 2 (Nov. 2004), 91–110.

[30] Fischler, M. and Bolles, R.
Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography.
Communications of the Association for Computing Machinery, **24** (1981), 381–395.

[31] Gurfil, P. and Rotstein, H.
Partial aircraft state estimation from visual motion using the subspace constraints approach.
Journal of Guidance, Control and Dynamics, **24**, 5 (July 2001), 1016–1028.

[32] Indelman, V., et al.
Distributed vision-aided cooperative localization and navigation based on three-view geometry.
Proceedings of the IEEE Aerospace Conference, Big Sky, MT, Mar. 2011.

[33] Indelman, V., et al.
Graph-based distributed cooperative navigation.
Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, May 2011.

Vadim Indelman received his B.Sc. and B.A. in aerospace engineering and computer science, respectively, from the Technion–Israeli Institute of Technology in 2002.

He has recently finished a direct track Ph.D. with the faculty of aerospace engineering at the Technion. His research interests include vision-aided navigation, cooperative navigation, SLAM, bundle adjustment and mosaicking.

Pini Gurfil received his Ph.D. in aerospace engineering from the Technion in 2000.

He is an Associate Professor of Aerospace Engineering at the Technion–Israel Institute of Technology. From 2000 to 2003, he was with Princeton University's Department of Mechanical and Aerospace Engineering, where he served as research staff member and lecturer. In September 2003, he joined the faculty of aerospace engineering at the Technion, where he is founder and head of the Distributed Space Systems Laboratory. He has been conducting research in astrodynamics, distributed space systems, vision-based navigation, and multiagent systems.

Dr. Gurfil has published over 150 journal and conference articles in these areas of his research interests.

Ehud Rivlin got his Ph.D. in computer science in 1993 from the University of Maryland, College Park.

He is a Professor of Computer Science at the Technion, Israel Institute of Technology. His research interests include visual recognition, event perception, biologically motivated vision and sensory based motion planning.

Hector Rotstein has an electrical engineer degree from the Universidad Nacional del Sur, Argentina, and M.Sc. and Ph.D. degrees in electrical engineering from Caltech, the California Institute of Technology.

He is a research fellow at the Missile Division of Rafael, Advanced Defense Systems Ltd., Israel. In addition, he is a senior visiting lecturer at the Technion, the Israel Institute of Technology, where he teaches courses in advanced control and in navigation systems. In the year 2003–2004 he was a visiting professor at the University of Minnesota, Minneapolis. His current research interests include the use of vision for navigation and control and novel applications of navigation systems.

Dr. Rotstein holds several patents and has published more than 40 papers in technical journals, mostly in the field of robust control and vision navigation.