2012 IEEE International Conference on Robotics and Automation
RiverCentre, Saint Paul, Minnesota, USA
May 14-18, 2012

Cooperative Vision-aided Inertial Navigation Using Overlapping Views

Igor V. Melnyk, Joel A. Hesch, and Stergios I. Roumeliotis

***Abstract*— In this paper, we study the problem of Cooperative Localization (CL) for two robots, each equipped with an Inertial Measurement Unit (IMU) and a camera. We present an algorithm that enables the robots to exploit common features, observed over a sliding-window time horizon, in order to improve the localization accuracy of both vehicles. In contrast to existing CL methods, which require robot-to-robot distance and/or bearing measurements to resolve the robots' relative position and orientation (pose), our approach recovers the relative pose through indirect information from the commonly observed features. Moreover, we analyze the system observability properties to determine how many degrees of freedom (d.o.f.) of the relative transformation can be computed under different measurement scenarios. Lastly, we present simulation results to evaluate the performance of the proposed method.**

I. Introduction

Teams of coordinating autonomous robots have potential uses in many applications such as aerial surveillance [3], search and rescue missions [22], and environmental mapping [27]. Accurate localization, i.e., estimating the position and orientation (pose) of each robot in the team, is a key prerequisite for successfully accomplishing these tasks. For instance, during a natural disaster, such as a flood or an earthquake, it is important to quickly locate survivors within the affected area. A team of Unmanned Air Vehicles (UAVs), equipped with high-resolution cameras, can be deployed to visually surveil the area. Knowing the positions of the vehicles at the times the images are recorded is critical for guiding the rescue personnel to reach the injured people.

Existing navigation systems typically rely on GPS signals for localization, however, many environments preclude the use of GPS (e.g., in the urban canyon or under the tree canopy). An alternative for localizing a UAV in GPS-denied environments is to utilize onboard sensors that measure the vehicle's motion with respect to the surrounding environment to track its pose. Since each UAV can be equipped with its own sensors, one potential strategy is to have each team member localize independently. However, if the robots cooperate, not only will they be more effective in accomplishing their required tasks, but their localization accuracy will also be improved [14]. In heterogeneous teams, this is particularly effective since the vehicle with the least accurate sensors can gain localization accuracy comparable to the vehicle with the highest quality sensors.

This work was supported by the University of Minnesota (DTC), the National Science Foundation (IIS-0811946), and AFOSR (FA9550-10-1-0567). J. A. Hesch was supported by the UMN Doctoral Dissertation Fellowship.

The authors are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, Emails: {melnyk|joel|stergios}@cs.umn.edu

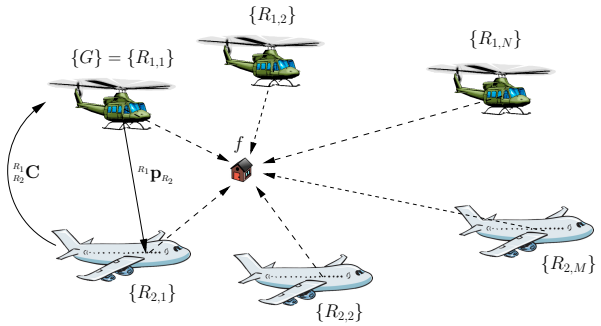

Fig. 1: Geometry of the trajectories of two robots navigating in 3D and acquiring visual observations of a common landmark f. At time step k, the pose of robot R_i, $i = 1, 2$, with respect to the global frame of reference $\{G\}$ is denoted as $\{R_{i,k}\}$. $({}^{R_1}\mathbf{p}_{R_2}, {}^{R_1}_{R_2}\mathbf{C})$ is the relative transformation between the robots' initial frames $\{R_{1,1}\}$ and $\{R_{2,1}\}$. The dashed lines represent the camera observations.

Many existing Cooperative Localization (CL) approaches assume that the robots can *directly* measure the distance and/or bearing to each other [26]. This is a limitation, since in many cases a direct line-of-sight requirement is hard to satisfy, or the distance between the robots might be too large, causing the measurement data to be inaccurate. Alternatively, the robots can perform CL using *indirect* measurements, i.e., they can infer their relative pose by observing the same scene features (see Fig. 1). Since in most cases a map of the environment is not known *a priori*, the robots would need to perform Cooperative Simultaneous Localization and Mapping (C-SLAM) [6]. However, this requires them to estimate and store a map of the environment, which is impractical for robots with limited resources.

To address these issues, we propose a method to perform CL in a team of vehicles, each equipped with an Inertial Measurement Unit (IMU) and a camera, which avoids building and maintaining a map of the environment. Each robot localizes by fusing its inertial information with indirect vision-based observations of its team members. This work extends our Multi-State Constraint Kalman Filter (MSC-KF) [15], to the case of two or more robots localizing cooperatively[1]. The MSC-KF estimates a robot's 3D pose by combining visual and inertial measurements without building a map of the environment, and has computational complexity that is only *linear* in the number of features.

To this end, we introduce the Cooperative Localization MSC-KF (CL-MSC-KF) algorithm and investigate the in-

[1]For the purpose of clarity, we focus on the two-robot case; however, the results of this work can readily be extended to the case of multiple robots.

formation that is available to the two robots when they observe different numbers of common features in one or more images. The summary of this analysis is as follows: (i) Given five or more common features at one time step, at most five degrees of freedom (d.o.f.) of the robots' relative pose are observable. (ii) If at least three features can be matched in two consecutive images, all six d.o.f. of relative pose can be recovered. (iii) All six d.o.f. can also be determined when at least two features are tracked in two images and each robot measures the gravity-vector direction with its IMU. The practical implication of this analysis is when the relative transformation is observable, CL-MSC-KF can be effectively utilized to provide high accuracy pose estimates for the entire team.

The remainder of this paper is organized as follows: In the next section, we discuss the related work on cooperative localization and vision-aided inertial navigation. Section III presents the CL-MSC-KF algorithm. In Section IV, we present the observability analysis of CL based on indirect visual measurements. Simulation results are shown in Section V, which demonstrate the validity of the proposed algorithm. Finally, in Section VI we provide our concluding remarks and discuss future research directions.

II. Related Work

A. Cooperative Localization and Mapping

Several different techniques have been developed to localize a team of cooperating robots. Kurazume et al. [9] presented one of the earliest CL methods which relied on coordinated motion, where some of the robots remain stationary while the others move and use the first group as static landmarks to improve their localization accuracy. Similar approaches, based on specific motion strategies, have also been presented in [19], [24]. The drawback of these techniques is that restricting the robots' motions may prevent them from being used in time-critical tasks.

Howard et al. [8] proposed an algorithm to localize a team of robots by treating the individual team members as mobile landmarks without any motion restrictions. Using robot-to-robot relative pose observations and odometry measurements from each robot they derived a Maximum Likelihood estimator that jointly computes the poses of all the robots. A Kalman filter-based approach was developed in [20] that avoids the excessive computational complexity of the previous method by only estimating the robots' poses at the current time step, and marginalizing past poses. A Maximum a Posteriori estimator, which distributes the data processing amongst the robot team has also been proposed for CL [17]. A common characteristic of these methods is that they rely on robot-to-robot distance and/or bearing measurements. This is a limitation, since in many practical situations, inter-robot observations may not be available (e.g., due to large distances or visibility constraints).

Alternatively, exteroceptive measurements of common environmental features can be used to improve the localization accuracy of the team. For example, C-SLAM algorithms can be utilized to create a map of the environment, which all robots can use to perform cooperative localization (e.g., [6], [7], [23]). However, the processing and storage requirements of C-SLAM depend on the map size, which may be prohibitive for resource-constrained vehicles exploring large areas. Furthermore, most of these methods address the localization problem in 2D settings, which limits their applicability in real-world scenarios that require the team to move in 3D.

B. Visual Odometry and Vision-aided Inertial Navigation

For camera-equipped vehicles, visual odometry is an alternative approach for tracking a robot's trajectory, which avoids estimating the landmarks' positions. For example, Nister et al. [18] estimate the motion of the camera by imposing constraints over consecutive camera poses. The main drawback of this method is the continuous accumulation of displacement errors for which no measure of uncertainty is provided. For a group of UAVs [12], a homography-based method is presented in which the observations of the common scene enables the robots to estimate their relative poses and localize with respect to a common frame of reference. Unfortunately, the planar scene assumption is unsuitable for many real-world scenarios (e.g., when flying near the ground or indoors). Moreover, since only visual information is used to estimate motion, the above approaches may lead to large estimation errors when no image features are extracted or matched.

Alternatively, vision-aided inertial navigation methods have been proposed which utilize an IMU, in addition to a camera. For example, in [1] the information about the rotation and the direction of translation between two vehicles viewing a common scene is fused with IMU measurements to estimate the relative transformation between two robots. The formulated algorithm, however, does not localize the robots with respect to a global frame of reference, thus, limiting its practical applications. On the other hand, in [2] and [4] constraints between current and past images are combined with IMU measurements to perform single-robot pose estimation in the global frame of reference. However, since the constraints are only defined between *pairs* of images, information is discarded when the same features are visible in more than two images. The MSC-KF algorithm [15], on the other hand, exploits the geometric relationship between features observed from *multiple* camera poses to constrain the robot trajectory. This provides higher estimation accuracy in cases when a feature is observed in more than two views. Since the landmarks' positions are not estimated, the computational complexity is linear in the number of features, enabling real-time performance.

In this paper, we extend the MSC-KF algorithm to the case of two robots performing 3D cooperative localization (termed CL-MSC-KF). In contrast to the CL methods mentioned above, our approach is more flexible since it utilizes indirect relative-pose measurements, i.e., scene features visually observed by both robots, instead of inter-robot measurements. Moreover, as compared to the map-building approaches, the processing and memory requirements of our algorithm are

lower since we do not construct or maintain a map of the environment. In this work, we also perform an observability analysis to examine how many degrees of freedom of the robots' relative pose can be determined, under different measurement scenarios.

III. Problem Formulation and Solution

We begin by formulating the problem of CL for two robots, each equipped with an IMU that measures its rotational velocity and linear acceleration, and a camera that observes point features in the environment, whose global positions are unknown. Common visual features are tracked by both vehicles across multiple frames in order to gain information about the robot-to-robot transformation, and increase their localization accuracy. In this work, we consider a centralized estimation architecture for CL, where each robot sends its measurements to a fusion center that processes the data and estimates the poses of both robots. We assume that the initial poses of the robots are approximately known, e.g., using the method described in Section IV [see (17)], and the data association problem is solved, e.g., using visual feature descriptors [11] in conjunction with RANSAC.

A. State Vector

In what follows, the subscripts i and j ($i,j=1,2$) correspond to robots R_1 or R_2, while the subscript l denotes the camera pose index. The state vector of robot R_i is[2]

$$\mathbf{x}_{R_i}=\begin{bmatrix}{}^{R_i}_{G}\overline{q}^T & \mathbf{b}_{g_i}^T & {}^{G}\mathbf{v}_{R_i}^T & \mathbf{b}_{a_i}^T & {}^{G}\mathbf{p}_{R_i}^T\end{bmatrix}^T \tag{1}$$

where ${}^{R_i}_{G}\overline{q}$ is the unit quaternion that describes the orientation of the global frame $\{G\}$ with respect to the frame $\{R_i\}$ of robot R_i, ${}^{G}\mathbf{p}_{R_i}$ is the position and ${}^{G}\mathbf{v}_{R_i}$ is the velocity of R_i expressed in $\{G\}$, and $\mathbf{b}_{g_i}$ and $\mathbf{b}_{a_i}$ are the gyroscope and accelerometer biases, respectively. Without loss of generality, we assume that $\{G\}$ coincides with the initial frame of robot R_1. The error-state vector corresponding to (1) is

$$\widetilde{\mathbf{x}}_{R_i}=\begin{bmatrix}\delta\theta_{R_i}^T & \widetilde{\mathbf{b}}_{g_i}^T & {}^{G}\widetilde{\mathbf{v}}_{R_i}^T & \widetilde{\mathbf{b}}_{a_i}^T & {}^{G}\widetilde{\mathbf{p}}_{R_i}^T\end{bmatrix}^T \tag{2}$$

where $\delta\theta_{R_i}$ is the angle-error vector, defined by the error quaternion $\delta\overline{q}_{R_i}={}^{R_i}_{G}\overline{q}\otimes{}^{R_i}_{G}\hat{\overline{q}}^{-1}\simeq\begin{bmatrix}\frac{1}{2}\delta\theta_{R_i}^T & 1\end{bmatrix}^T$. Here, ${}^{R_i}_{G}\hat{\overline{q}}$ and ${}^{R_i}_{G}\overline{q}$ are the estimated and true orientation, respectively, and the symbol $\otimes$ denotes quaternion multiplication [25]. For the other terms in the error state an additive error model is employed, i.e., the error in the estimate $\hat{\mathbf{x}}$ of a quantity $\mathbf{x}$ is $\widetilde{\mathbf{x}}=\mathbf{x}-\hat{\mathbf{x}}$.

When either robot records a new image, the state vector is augmented with the corresponding camera pose (see Section III-C). This process, termed stochastic cloning [21], enables us to apply measurement constraints across multiple images recorded at different time instances, while correctly accounting for the correlations in the error-state (see Section III-D). Robot R_i's l-th camera pose and corresponding error vector are

$$\mathbf{x}_{C_{il}}=\begin{bmatrix}{}^{C_{il}}_{G}\overline{q}^T & {}^{G}\mathbf{p}_{C_{il}}^T\end{bmatrix}^T,\quad \widetilde{\mathbf{x}}_{C_{il}}=\begin{bmatrix}\delta\theta_{C_{il}}^T & {}^{G}\widetilde{\mathbf{p}}_{C_{il}}^T\end{bmatrix}^T \tag{3}$$

where ${}^{C_{il}}_{G}\overline{q}$ and ${}^{G}\mathbf{p}_{C_{il}}$ denote its attitude and position, respectively, and the error quantities are defined as above.

The joint state vector comprises the current states of both robots and a history of their past camera poses

$$\mathbf{x}=\begin{bmatrix}\mathbf{x}_{R_1}^T & \mathbf{x}_{R_2}^T & \mathbf{x}_{C_{11}}^T\cdots & \mathbf{x}_{C_{1N}}^T & \mathbf{x}_{C_{21}}^T\cdots & \mathbf{x}_{C_{2M}}^T\end{bmatrix}^T=\begin{bmatrix}\mathbf{x}_{R_1}^T & \mathbf{x}_{R_2}^T & \mathbf{x}_{C}^T\end{bmatrix}^T \tag{4}$$

where $\mathbf{x}_C$ is the vector containing the $N+M$ previous camera poses of robots R_1 and R_2.

B. Propagation

We now proceed with an overview of the CL-MSC-KF algorithm. We first present the continuous-time kinematic model of the robots' motion. By linearizing it, we obtain the model describing the time evolution of the error-state. Finally, we discretize these models to obtain the equations for propagating the state and its associated covariance estimates using the IMU measurements.

Specifically, the system model describing the time evolution of the robot state (1) is given by

$${}^{R_i}_{G}\dot{\overline{q}}(t)=\frac{1}{2}\Omega\big(\boldsymbol{\omega}_{R_i}(t)\big){}^{R_i}_{G}\overline{q}(t),\quad \dot{\mathbf{b}}_{g_i}(t)=\mathbf{n}_{wg_i}(t),\quad \dot{\mathbf{b}}_{a_i}(t)=\mathbf{n}_{wa_i},$$
$${}^{G}\dot{\mathbf{v}}_{R_i}(t)={}^{G}\mathbf{a}_{R_i}(t),\quad {}^{G}\dot{\mathbf{p}}_{i}(t)={}^{G}\mathbf{v}_{R_i}(t) \tag{5}$$

where ${}^{G}\mathbf{a}_{R_i}$ is the acceleration of robot R_i, $\boldsymbol{\omega}_{R_i}=\begin{bmatrix}\omega_{x_{R_i}} & \omega_{y_{R_i}} & \omega_{z_{R_i}}\end{bmatrix}^T$ is its rotational velocity expressed in the local frame of robot R_i, $\mathbf{n}_{wg_i}$ and $\mathbf{n}_{wa_i}$ are the zero-mean white Gaussian random walk processes driving the IMU biases, and

$$\Omega(\boldsymbol{\omega}_{R_i})=\begin{bmatrix}-\lfloor\boldsymbol{\omega}_{R_i}\times\rfloor & \boldsymbol{\omega}_{R_i}\\ -\boldsymbol{\omega}_{R_i}^T & 0\end{bmatrix},\ \lfloor\boldsymbol{\omega}_{R_i}\times\rfloor=\begin{bmatrix}0 & -\omega_{z_{R_i}} & \omega_{y_{R_i}}\\ \omega_{z_{R_i}} & 0 & -\omega_{x_{R_i}}\\ -\omega_{y_{R_i}} & \omega_{x_{R_i}} & 0\end{bmatrix}.$$

The measured rotational velocity and linear acceleration are modeled as $\boldsymbol{\omega}_{m_{R_i}}=\boldsymbol{\omega}_{R_i}+\mathbf{b}_{g_i}+\mathbf{n}_{g_i}$ and $\mathbf{a}_{m_{R_i}}=\mathbf{C}({}^{R_i}_{G}\overline{q})({}^{G}\mathbf{a}_{R_i}-{}^{G}\mathbf{g})+\mathbf{b}_{a_i}+\mathbf{n}_{a_i}$, respectively. Here, $\mathbf{C}({}^{R_i}_{G}\overline{q})$ is the rotation matrix corresponding to the quaternion ${}^{R_i}_{G}\overline{q}$, $\mathbf{n}_{g_i}$ and $\mathbf{n}_{a_i}$ are zero-mean white Gaussian noise processes, and ${}^{G}\mathbf{g}$ is the gravitational acceleration.

The state-estimate propagation model is obtained by linearizing (5) around the current estimates and applying the expectation operator, i.e.,

$${}^{R_i}_{G}\dot{\hat{\overline{q}}}(t)=\frac{1}{2}\Omega\big(\hat{\boldsymbol{\omega}}_{R_i}(t)\big){}^{R_i}_{G}\hat{\overline{q}}(t),\quad \dot{\hat{\mathbf{b}}}_{g_i}(t)=\mathbf{0}_{3\times1},\quad \dot{\hat{\mathbf{b}}}_{a_i}(t)=\mathbf{0}_{3\times1},$$
$${}^{G}\dot{\hat{\mathbf{v}}}_{R_i}(t)=\mathbf{C}\big({}^{R_i}_{G}\hat{\overline{q}}(t)\big)^T\hat{\mathbf{a}}_{R_i}(t)+{}^{G}\mathbf{g},\quad {}^{G}\dot{\hat{\mathbf{p}}}_{R_i}(t)={}^{G}\hat{\mathbf{v}}_{R_i}(t) \tag{6}$$

where $\hat{\mathbf{a}}_{R_i}=\mathbf{a}_{m_{R_i}}-\hat{\mathbf{b}}_{a_i}$ and $\hat{\boldsymbol{\omega}}_{R_i}=\boldsymbol{\omega}_{m_{R_i}}-\hat{\mathbf{b}}_{g_i}$.

The linearized continuous-time model for the error-state (2) is $\dot{\widetilde{\mathbf{x}}}_{R_i}=\mathbf{F}_{R_i}\widetilde{\mathbf{x}}_{R_i}+\mathbf{G}_{R_i}\mathbf{n}_{R_i}$, where $\mathbf{n}_{R_i}=\begin{bmatrix}\mathbf{n}_{g_i}^T & \mathbf{n}_{wg_i}^T & \mathbf{n}_{a_i}^T & \mathbf{n}_{wa_i}^T\end{bmatrix}^T$ is the system noise whose covariance matrix $\mathbf{Q}_{R_i}$ depends on the IMU noise characteristics of robot R_i and is computed off-line [25]. The Jacobian matrices $\mathbf{F}_{R_i}$ and $\mathbf{G}_{R_i}$ are

[2]For the clarity of presentation, we omit the time variable from time-varying quantities defined hereafter. Time appears when describing the continuous-time equations of motion and the discrete-time measurement equations.

$$\mathbf{F}_{R_i}=\begin{bmatrix} -\lfloor \hat{\boldsymbol{\omega}}_{R_i}\times\rfloor & -\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -\mathbf{C}({}^{R_i}_{G}\hat{\overline{q}})^T\lfloor \hat{\mathbf{a}}_{R_i}\times\rfloor & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\mathbf{C}({}^{R_i}_{G}\hat{\overline{q}})^T & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix},$$

$$\mathbf{G}_{R_i}=\begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\mathbf{C}({}^{R_i}_{G}\hat{\overline{q}})^T & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}$$

where $\mathbf{I}_3$ is the 3×3 identity matrix. When robot R_i records an IMU measurement, the corresponding state estimate $\hat{\mathbf{x}}_{R_i}$ is propagated using 4th-order Runge-Kutta numerical integration of (6). Note that the camera poses in (4) are static, and do not change during the propagation step. To derive the covariance propagation equations, we introduce the following partitioning of the covariance matrix at time-step k given IMU and camera measurements up to time-step k

$$\mathbf{P}_{k|k}=\begin{bmatrix} \mathbf{P}_{R_1R_1} & \mathbf{P}_{R_1R_2} & \mathbf{P}_{R_1C} \\ \mathbf{P}_{R_1R_2}^T & \mathbf{P}_{R_2R_2} & \mathbf{P}_{R_2C} \\ \mathbf{P}_{R_1C}^T & \mathbf{P}_{R_2C}^T & \mathbf{P}_{CC} \end{bmatrix} \tag{7}$$

where $\mathbf{P}_{R_iR_j}$ is the 15×15 covariance/correlation matrix for the robot error-states $\widetilde{\mathbf{x}}_{R_i}$ and $\widetilde{\mathbf{x}}_{R_j}$, and $\mathbf{P}_{R_iC}$ is the $15\times6(N+M)$ correlation matrix between $\widetilde{\mathbf{x}}_{R_i}$ and $\widetilde{\mathbf{x}}_C$. Finally, $\mathbf{P}_{CC}$ is the $6(N+M)\times6(N+M)$ covariance matrix of the $N+M$ combined camera error-states $\widetilde{\mathbf{x}}_C$ for robots R_1 and R_2. With this notation, the propagated covariance matrix for both robots is given by

$$\mathbf{P}_{k+1|k}=\begin{bmatrix} \mathbf{P}_{R_1R_1}^{k+1|k} & \Phi_1\mathbf{P}_{R_1R_2}\Phi_2^T & \Phi_1\mathbf{P}_{R_1C} \\ \Phi_2\mathbf{P}_{R_1R_2}^T\Phi_1^T & \mathbf{P}_{R_2R_2}^{k+1|k} & \Phi_2\mathbf{P}_{R_2C} \\ \mathbf{P}_{R_1C}^T\Phi_1^T & \mathbf{P}_{R_2C}^T\Phi_2^T & \mathbf{P}_{CC} \end{bmatrix} \tag{8}$$

where $\mathbf{P}_{R_iR_i}^{k+1|k}$ is the propagated covariance of the state of robot R_i and Φ_i is the state-transition matrix; both quantities are computed by numerical integration.

C. State and Covariance Augmentation

Every time a new image is recorded, the state vector is expanded to include the pose estimate of the camera that recorded the image. Note, that if the state already contains the maximum number of past camera poses, the oldest one is marginalized before including a new one. Denoting the current camera pose as l, its estimate is calculated as

$$\begin{aligned} {}^{C_{il}}_{G}\hat{\overline{q}} &= {}^{C}_{R_i}\overline{q}\otimes{}^{R_i}_{G}\hat{\overline{q}} \\ {}^{G}\hat{\mathbf{p}}_{C_{il}} &= {}^{G}\hat{\mathbf{p}}_{R_i}+\mathbf{C}({}^{R_i}_{G}\hat{\overline{q}})^T\,{}^{R_i}\mathbf{p}_C \end{aligned} \tag{9}$$

where the IMU-camera transformation $\{{}^{C}_{R_i}\overline{q},{}^{R_i}\mathbf{p}_C\}$ is computed off-line [13]. The camera poses for robot R_2 are appended at the end of the state vector (4), whereas for robot R_1 they are appended to the end of the list of the existing R_1 camera poses.

The covariance matrix is augmented as

$$\mathbf{P}_{k|k}:=\begin{bmatrix} \mathbf{P}_{k|k} & \mathbf{P}_{k|k}\mathbf{J}_{R_i}^T \\ \mathbf{J}_{R_i}\mathbf{P}_{k|k} & \mathbf{J}_{R_i}\mathbf{P}_{k|k}\mathbf{J}_{R_i}^T \end{bmatrix} \tag{10}$$

where $\mathbf{J}_{R_i}$ is the Jacobian of (9) with respect to the state vector (4). For example, for robot R_1 it takes the following form

$$\mathbf{J}_{R_1}=\begin{bmatrix} \mathbf{C}({}^{C}_{R_1}\overline{q}) & \mathbf{0}_{3\times9} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times[6(N+M)+15]} \\ -\mathbf{C}({}^{R_1}_{G}\hat{\overline{q}})^T\lfloor{}^{R_1}\mathbf{p}_C\times\rfloor & \mathbf{0}_{3\times9} & \mathbf{I}_3 & \mathbf{0}_{3\times[6(N+M)+15]} \end{bmatrix}.$$

Note that for robot R_1, after applying (10), the columns and rows of the resulting matrix need to be appropriately interchanged to obtain the correct covariance matrix.

D. Measurement Update

We now present the measurement model describing the observation of an unknown feature f by robot R_i. Using the perspective projection camera model with unit focal length, the observation of feature f in the l-th camera image is

$$\mathbf{z}_l^i=\frac{1}{{}^{C_{il}}z}\begin{bmatrix}{}^{C_{il}}x\\ {}^{C_{il}}y\end{bmatrix}+\mathbf{n}_l^i \tag{11}$$

where $\begin{bmatrix}{}^{C_{il}}x\\ {}^{C_{il}}y\\ {}^{C_{il}}z\end{bmatrix}={}^{C_{il}}\mathbf{p}_f=\mathbf{C}({}^{C_{il}}_{G}\overline{q})({}^{G}\mathbf{p}_f-{}^{G}\mathbf{p}_{C_{il}})$ is the position of the feature with respect to the camera, and $\mathbf{n}_l^i$ is the zero-mean Gaussian pixel noise with covariance matrix $\sigma^2\mathbf{I}_2$. Linearizing (11), we obtain the measurement residual

$$\begin{aligned} \widetilde{\mathbf{z}}_l^i &\simeq \mathbf{H}_{\delta\theta_l}^i\delta\theta_{C_{il}}+\mathbf{H}_{\mathbf{p}_l}^i\,{}^{G}\widetilde{\mathbf{p}}_{C_{il}}+\mathbf{H}_{f_l}^i\,{}^{G}\widetilde{\mathbf{p}}_f+\mathbf{n}_l^i \\ &= \mathbf{H}_{\mathbf{x}_{C_l}}^i\widetilde{\mathbf{x}}_{C_l}+\mathbf{H}_{f_l}^i\,{}^{G}\widetilde{\mathbf{p}}_f+\mathbf{n}_l^i, \end{aligned} \tag{12}$$

where

$$\mathbf{H}_{\delta\theta_l}^i=\frac{1}{{}^{C_{il}}\hat{z}}\begin{bmatrix}\mathbf{I}_2 & -\hat{\mathbf{z}}_l^i\end{bmatrix}\lfloor\mathbf{C}({}^{C_{il}}_{G}\hat{\overline{q}})({}^{G}\hat{\mathbf{p}}_f-{}^{G}\hat{\mathbf{p}}_{C_{il}})\times\rfloor$$

$$\mathbf{H}_{\mathbf{p}_l}^i=-\frac{1}{{}^{C_{il}}\hat{z}}\begin{bmatrix}\mathbf{I}_2 & -\hat{\mathbf{z}}_l^i\end{bmatrix}\mathbf{C}({}^{C_{il}}_{G}\hat{\overline{q}}),\quad \mathbf{H}_{f_l}^i=-\mathbf{H}_{\mathbf{p}_l}^i$$

where $\hat{\mathbf{z}}_l^i$ is the estimated feature measurement. Since ${}^{G}\mathbf{p}_f$ is unknown, we evaluate the Jacobians at ${}^{G}\hat{\mathbf{p}}_f$, which is obtained by triangulating the feature position from two or more views. By stacking together all the measurement residuals for both robots, we have

$$\begin{bmatrix}\widetilde{\mathbf{z}}_1^1\\ \vdots\\ \widetilde{\mathbf{z}}_N^1\\ \widetilde{\mathbf{z}}_1^2\\ \vdots\\ \widetilde{\mathbf{z}}_M^2\end{bmatrix}=\begin{bmatrix}\mathbf{0}_{2N\times30} & \mathbf{H}^1 & \mathbf{0}_{2N\times6M}\\ \mathbf{0}_{2M\times30} & \mathbf{0}_{2M\times6N} & \mathbf{H}^2\end{bmatrix}\begin{bmatrix}\widetilde{\mathbf{x}}_{R_1}\\ \widetilde{\mathbf{x}}_{R_2}\\ \widetilde{\mathbf{x}}_{C_{11}}\\ \vdots\\ \widetilde{\mathbf{x}}_{C_{1N}}\\ \widetilde{\mathbf{x}}_{C_{21}}\\ \vdots\\ \widetilde{\mathbf{x}}_{C_{2M}}\end{bmatrix}+\begin{bmatrix}\mathbf{H}_{f_1}^1\\ \vdots\\ \mathbf{H}_{f_N}^1\\ \mathbf{H}_{f_1}^2\\ \vdots\\ \mathbf{H}_{f_M}^2\end{bmatrix}{}^{G}\widetilde{\mathbf{p}}_f+\begin{bmatrix}\mathbf{n}_1^1\\ \vdots\\ \mathbf{n}_N^1\\ \mathbf{n}_1^2\\ \vdots\\ \mathbf{n}_M^2\end{bmatrix} \tag{13}$$

where $\mathbf{H}^1=\text{diag}\left[\mathbf{H}_{\mathbf{x}_{C_1}}^1\ldots\mathbf{H}_{\mathbf{x}_{C_N}}^1\right]$ is the block-diagonal matrix of size $2N\times6N$ corresponding to the N camera poses of robot R_1 and, similarly for robot R_2, $\mathbf{H}^2=$

$\text{diag}\left[\mathbf{H}_{\mathbf{x}_{C_1}}^2 \ldots \mathbf{H}_{\mathbf{x}_{C_M}}^2\right]$ is the block-diagonal matrix of size $2M \times 6M$. For notational clarity, we have assumed that each feature is observed in all images. In general, however, any subset of them can be used in the update step[3]. In a more compact form, we rewrite (13) as

$$\widetilde{\mathbf{z}} = \mathbf{H}\,\widetilde{\mathbf{x}} + \mathbf{H}_f\,{}^G\widetilde{\mathbf{p}}_f + \mathbf{n} \tag{14}$$

where $\widetilde{\mathbf{x}}$ is the error-state corresponding to (4), and the covariance of the noise $\mathbf{n}$ is $\sigma^2\mathbf{I}_{2(N+M)}$. Note that since ${}^G\hat{\mathbf{p}}_f$ was triangulated using estimated camera poses, the error ${}^G\widetilde{\mathbf{p}}_f$ in the estimated feature position is correlated with the state errors $\widetilde{\mathbf{x}}$, therefore, the residual (14) cannot be directly used in the EKF update step. We could properly account for this correlation, by adding the feature estimate to the state vector, however, this would increase the computational complexity and storage requirements of our algorithm. A more efficient way to overcome this issue is to marginalize ${}^G\mathbf{p}_f$ on the fly. To do so, we eliminate ${}^G\widetilde{\mathbf{p}}_f$ from (14) by projecting $\widetilde{\mathbf{z}}$ onto the left null space of $\mathbf{H}_f$. Let $\mathbf{W}$ be the unitary matrix whose columns span the left null space of $\mathbf{H}_f$. Since $\mathbf{H}_f$ in our problem formulation is of size $2(N+M)\times 3$ and its rank in general is three, the dimension of $\mathbf{W}$ is $2(N+M) \times (2(N+M)-3)$. Multiplying equation (14) from the left with $\mathbf{W}^T$ yields

$$\widetilde{\mathbf{z}}_0 = \mathbf{W}^T\widetilde{\mathbf{z}} = \mathbf{W}^T\mathbf{H}\,\widetilde{\mathbf{x}} + \mathbf{W}^T\mathbf{n} = \mathbf{H}_0\widetilde{\mathbf{x}} + \mathbf{n}_0 \tag{15}$$

where the noise covariance is $\mathbf{E}\left[\mathbf{n}_0\mathbf{n}_0^T\right] = \mathbf{E}\left[\mathbf{W}^T\mathbf{n}\mathbf{n}^T\mathbf{W}\right] = \sigma^2\mathbf{I}_{2(N+M)-3}$. The standard EKF equations can now be applied to perform the update. Note that multiplying $\mathbf{H}$ by $\mathbf{W}^T$ causes the resulting measurement matrix, $\mathbf{H}_0$, to be dense. This couples the robot pose estimates by introducing the cross-correlation terms into the covariance matrix $\mathbf{P}_{k|k}$ during the update step.

IV. Observability Analysis

It is well known that CL methods, which exploit robot-to-robot measurements, result in improved localization accuracy for the entire team [14]. Although in the CL-MSC-KF the robots do not measure each other, their relative pose is observable under some mild conditions. Therefore, by combining the pose estimates from both vehicles, the CL-MSC-KF achieves improved localization accuracy compared to the case in which both vehicles independently localize using the MSC-KF (see Section V). In what follows, we determine these conditions by examining which d.o.f. of the relative pose are observable under different measurement configurations.

Consider two robots navigating in 3D using cameras to observe a number of common scene features in an ideal noise-free environment (see Fig. 1). We will identify the minimum number of common features observed by both robots and the minimum number of images needed to recover the six d.o.f. relative transformation between the robots' initial frames $\{R_{1,1}\}$ and $\{R_{2,1}\}$. We denote the position of robot R_2 with respect to R_1 as $\mathbf{p} := {}^{R_1}\mathbf{p}_{R_2} = \mathbf{C}({}^{R_1}_{G}\overline{q})({}^G\mathbf{p}_{R_2} - {}^G\mathbf{p}_{R_1})$ and the orientation of R_2 with respect to R_1 as $\mathbf{C} := {}^{R_1}_{R_2}\mathbf{C} = \mathbf{C}({}^{R_1}_{G}\overline{q})\mathbf{C}({}^{R_2}_{G}\overline{q})^T$. We assume that R_1 can estimate its poses $\{R_{1,k}\}$, at the time steps $k = 2,\ldots,N$ with respect to its initial frame of reference $\{R_{1,1}\}$ (e.g., by integrating its inertial measurements). Similarly, R_2 can estimate its motion, $\{R_{2,k}\}$, $k = 2,\ldots,M$, with respect to its own initial frame $\{R_{2,1}\}$ (see Fig. 1).

We first consider the case when the two robots observe L common features during a single time step. This is a well studied problem and it is known that $L \geq 5$ features must be observed by both robots in order to obtain a unique ($L > 5$) or a discrete ($L = 5$) set of solutions for the five d.o.f. relative robot-to-robot transformation, i.e., the orientation $\mathbf{C}$ and the position $\mathbf{p}$ up to scale [5]. To compute all six d.o.f. of relative transformation, we need to resolve the scale, which requires the robots to move.

A. Observation of Three or More Features

When three or more scene features are observed from two or more poses, the robots' relative transformation can be determined uniquely. To demonstrate this, let the robots observe $L = 3$ features (denoted as α, β, and γ) at two time steps. We can write the following geometric relationships

$${}^{R_1}\mathbf{p}_f = \mathbf{p} + \mathbf{C}\,{}^{R_2}\mathbf{p}_f, \quad f \in \{\alpha,\beta,\gamma\} \tag{16}$$

which form a system of nine equations in six unknowns. Here, ${}^{R_1}\mathbf{p}_f$ and ${}^{R_2}\mathbf{p}_f$ are the triangulated positions of feature f with respect to the initial frames $\{R_{1,1}\}$ and $\{R_{2,1}\}$, respectively. After defining ${}^{R_i}\mathbf{p}_{\alpha\beta} = {}^{R_i}\mathbf{p}_\alpha - {}^{R_i}\mathbf{p}_\beta$ and ${}^{R_i}\mathbf{p}_{\beta\gamma} = {}^{R_i}\mathbf{p}_\beta - {}^{R_i}\mathbf{p}_\gamma$, $i = 1,2$, $\mathbf{C}$ and $\mathbf{p}$ can be computed from (16) as follows

$$\begin{aligned} &{}^{R_1}\mathbf{p}_{\alpha\beta} = \mathbf{C}\,{}^{R_2}\mathbf{p}_{\alpha\beta}, \quad {}^{R_1}\mathbf{p}_{\beta\gamma} = \mathbf{C}\,{}^{R_2}\mathbf{p}_{\beta\gamma} \quad \Rightarrow \\ &\mathbf{C} = \left[{}^{R_1}\mathbf{p}_{\alpha\beta}\ {}^{R_1}\mathbf{p}_{\beta\gamma}\ {}^{R_1}\mathbf{p}_{\alpha\beta}\times{}^{R_1}\mathbf{p}_{\beta\gamma}\right]\left[{}^{R_2}\mathbf{p}_{\alpha\beta}\ {}^{R_2}\mathbf{p}_{\beta\gamma}\ {}^{R_2}\mathbf{p}_{\alpha\beta}\times{}^{R_2}\mathbf{p}_{\beta\gamma}\right]^{-1} \\ &\mathbf{p} = {}^{R_1}\mathbf{p}_\alpha - \mathbf{C}\,{}^{R_2}\mathbf{p}_\alpha \end{aligned} \tag{17}$$

where we used the fact that $(\mathbf{C}\mathbf{x})\times(\mathbf{C}\mathbf{y}) = \mathbf{C}(\mathbf{x}\times\mathbf{y})$ for any vectors $\mathbf{x}$ and $\mathbf{y}$, and assumed that the points are not collinear. Since we can recover the relative pose when $L = 3$, we can also determine it for the case when $L > 3$.

B. Observation of Two Features

When two features are observed across at least two time steps, we can write the following two relationships:

$${}^{R_1}\mathbf{p}_f = \mathbf{p} + \mathbf{C}\,{}^{R_2}\mathbf{p}_f, \quad f \in \{\alpha,\beta\}. \tag{18}$$

Even though this system has six equations in six unknowns, the set of solutions is infinite since either robot can rotate freely about the axis ${}^{R_2}\mathbf{p}_{\alpha\beta}$, and the constraints will not be violated. To see this, we rewrite (18) as ${}^{R_1}\mathbf{p}_{\alpha\beta} = \mathbf{C}\,{}^{R_2}\mathbf{p}_{\alpha\beta}$ and let $\mathbf{C}_{R_2\mathbf{p}_{\alpha\beta}}(\theta)$ denote an arbitrary rotation around axis ${}^{R_2}\mathbf{p}_{\alpha\beta}$ by an angle θ. If $\mathbf{C}$ satisfies the geometric constraints, then so does $\mathbf{C}' = \mathbf{C}\,\mathbf{C}_{R_2\mathbf{p}_{\alpha\beta}}(\theta)$, which is verified as follows

$$\mathbf{C}'\,{}^{R_2}\mathbf{p}_{\alpha\beta} = \mathbf{C}\,\mathbf{C}_{R_2\mathbf{p}_{\alpha\beta}}(\theta)\,{}^{R_2}\mathbf{p}_{\alpha\beta} = \mathbf{C}\,{}^{R_2}\mathbf{p}_{\alpha\beta} = {}^{R_1}\mathbf{p}_{\alpha\beta}$$

[3] For the case when one robot observes features that are not seen by the other robot, (14) can still be used by dropping the corresponding components of $\widetilde{\mathbf{z}}$, $\mathbf{H}$, $\mathbf{H}_f$, and $\mathbf{n}$.

since ${}^{R_2}\mathbf{p}_{\alpha\beta}$ lies along the direction of the eigenvector of $\mathbf{C}_{R_2\mathbf{p}_{\alpha\beta}}(\theta)$ corresponding to the eigenvalue 1 (i.e., it is the axis of rotation). Therefore, we cannot determine all six d.o.f. of the relative transformation.

Recall, however, that in addition to the camera, the robots use IMUs for navigation, which measure the gravity vector $\mathbf{g}$. This provides an additional constraint, ${}^{R_1}\mathbf{g} = \mathbf{C}\ {}^{R_2}\mathbf{g}$. By including this, the six d.o.f. relative transformation can be obtained using an approach similar to (17), as long as $\mathbf{g}$ is not parallel to $\mathbf{p}_{\alpha\beta}$.

C. Observation of a Single Feature

When the robots observe a single feature, $\mathbf{p}_\alpha$, over two or more poses we obtain only one constraint, containing three equations in six unknowns, which is an undetermined system of equations. By including the gravity-vector constraint, we obtain

$$
\begin{aligned}
{}^{R_1}\mathbf{p}_\alpha &= \mathbf{p} + \mathbf{C}\ {}^{R_2}\mathbf{p}_\alpha && (19)\\
{}^{R_1}\mathbf{g} &= \mathbf{C}\ {}^{R_2}\mathbf{g}, && (20)
\end{aligned}
$$

which has six equations in six unknowns, but as we will show, the number of solutions is infinite.

Let us assume that $(\mathbf{p},\mathbf{C})$ is a solution satisfying (19)-(20). Given $(\mathbf{p},\mathbf{C})$, we can show that there are infinitely many $(\mathbf{p}',\mathbf{C}')$, which also satisfy (19)-(20). Specifically, it can be seen from (20) that an arbitrary rotation $\mathbf{C}_{R_2\mathbf{g}}(\theta)$ around the gravity vector ${}^{R_2}\mathbf{g}$ is undetermined. Therefore, $\mathbf{C}' = \mathbf{C}\ \mathbf{C}_{R_2\mathbf{g}}(\theta)$ will also satisfy (20), since vector ${}^{R_2}\mathbf{g}$ is the axis of rotation of $\mathbf{C}_{R_2\mathbf{g}}(\theta)$. Now let $\mathbf{p}'$ be such that together with $\mathbf{C}'$ they satisfy (19), i.e., ${}^{R_1}\mathbf{p}_\alpha = \mathbf{p}' + \mathbf{C}'\ {}^{R_2}\mathbf{p}_\alpha$ holds. Using the latter together with (19), we can express $\mathbf{p}'$ and $\mathbf{C}'$ in the form

$$
\begin{aligned}
\mathbf{p}' &= \mathbf{p} + \mathbf{C}\ {}^{R_2}\mathbf{p}_\alpha - \mathbf{C}\ \mathbf{C}_{R_2\mathbf{g}}(\theta)\ {}^{R_2}\mathbf{p}_\alpha\\
\mathbf{C}' &= \mathbf{C}\ \mathbf{C}_{R_2\mathbf{g}}(\theta).
\end{aligned} \qquad (21)
$$

We can verify that $(\mathbf{p}',\mathbf{C}')$ is a solution by substituting (21) into (19)-(20). Furthermore, since there are infinitely many choices for the rotation angle θ, there are also infinitely many solutions to (19)-(20).

To geometrically interpret the obtained set of solutions, note that $(\mathbf{p},\mathbf{C})$ describes the pose of robot R_2 with respect to robot R_1, while $(\mathbf{p}',\mathbf{C}')$ describes another pose of R_2. Using homogeneous coordinates, we can rewrite (21) as a series of homogeneous transformations

$$
\begin{aligned}
\begin{bmatrix} \mathbf{C}' & \mathbf{p}' \\ 0 & 1 \end{bmatrix} &= \begin{bmatrix} \mathbf{C} & \mathbf{p} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}_{R_2\mathbf{g}}(\theta) & \left(\mathbf{I}_3 - \mathbf{C}_{R_2\mathbf{g}}(\theta)\right)\ {}^{R_2}\mathbf{p}_\alpha \\ 0 & 1 \end{bmatrix}\\
&= \begin{bmatrix} \mathbf{C} & \mathbf{p} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & {}^{R_2}\mathbf{p}_\alpha \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}_{R_2\mathbf{g}}(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & -{}^{R_2}\mathbf{p}_\alpha \\ 0 & 1 \end{bmatrix}.
\end{aligned}
$$

Therefore, given a pose of robot R_2 with respect to R_1 satisfying (19)-(20), any other pose can be obtained by translating R_2 by ${}^{R_2}\mathbf{p}_\alpha$ towards the feature α, then rotating around the gravity vector by an angle θ and, finally, translating back by $-{}^{R_2}\mathbf{p}_\alpha$. The set of all such poses comprises a circular continuum of solutions (see Fig. 2).

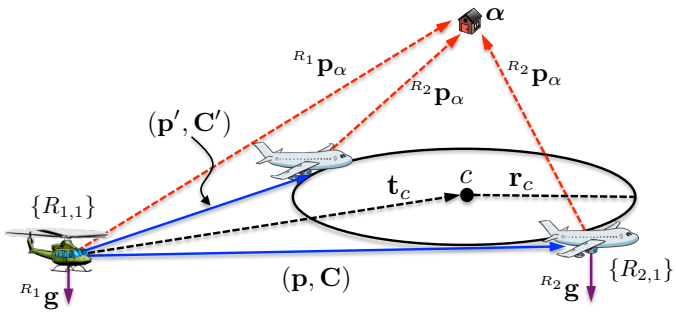

Fig. 2: Geometry of the unobservable motion of robot R_2 frame with respect to R_1. Given a relative transformation $(\mathbf{p},\mathbf{C})$ between the robots, $(\mathbf{p}',\mathbf{C}')$ is any other transformation satisfying the measurement constraints (19)-(20). The circular continuum of solutions is defined by its radius $r_c = ||\Pi\ {}^{R_2}\mathbf{p}_\alpha||_2$ and center $\mathbf{t}_c = \mathbf{p} + \mathbf{C}\ \Pi\ {}^{R_2}\mathbf{p}_\alpha$, where $\Pi = \mathbf{I}_3 - \frac{{}^{R_2}\mathbf{g}\ {}^{R_2}\mathbf{g}^T}{{}^{R_2}\mathbf{g}^T\ {}^{R_2}\mathbf{g}}$ is a projection matrix.

We conclude, therefore, that in order to determine the six d.o.f. relative transformation between the robots, at least three common features need to be observed at two time steps. If, in addition, the gravity vector is measured by both robots, then only two common features observed at two time steps, are necessary to find the transformation. Finally, when only one common feature is measured at two or more time steps, along with the gravity vector, the relative transformation between the robots remains unobservable.

V. Simulation Results

We hereafter present the results of simulation trials which demonstrate the performance of the proposed algorithm. We performed Monte Carlo simulations which compare the CL-MSC-KF to the case of both vehicles localizing independently using the MSC-KF. In the base case, two robots traversed a sinusoidal trajectory 45 km long, 50 m apart and 300 m above the ground. Each robot was equipped with an IMU, which provided measurements at 100 Hz, and a down-pointing camera that recorded images at 3 Hz. Each camera had 70° field of view and observed 50 features per image with pixel noise $\sigma = 1$ px. The overlap between the cameras' field of views was approximately 80%. The maximum number of camera poses through which a feature could be tracked was set to 15.

In the first simulation, we compared the performance of the proposed CL-MSC-KF algorithm to the single-vehicle MSC-KF, i.e., when each robot localizes independently (see Fig. 3). We conducted 100 Monte Carlo trials in which the estimator was initialized at the ground truth. The performance was evaluated using the Root Mean Squared Error (RMSE) metric by averaging over all Monte Carlo runs at each time step. Since the results for robots R_1 and R_2 are comparable, we show only the results for robot R_1. Note that since the system is not globally observable (i.e., no GPS measurements are available and no observations of known landmarks are used), the RMSE steadily increases for both methods. However, the rate of error increase is

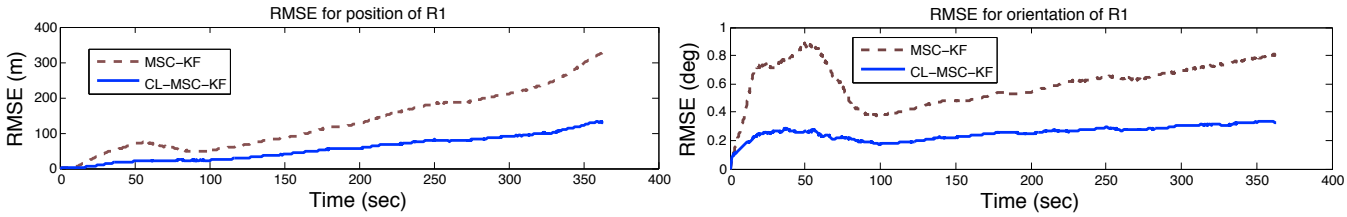

Fig. 3: Robot R_1 pose estimate errors, averaged over 100 Monte Carlo simulations. (Left): RMSE for the position estimate of R_1. (Right): RMSE for the orientation estimate of R_1.

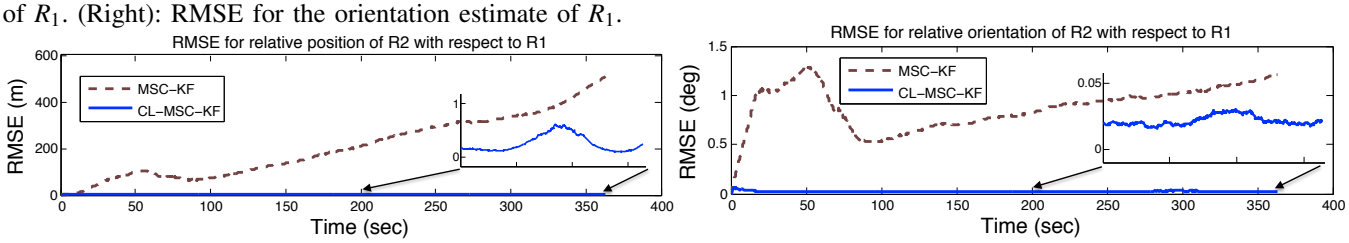

Fig. 4: Accuracy of the relative transformation averaged over 100 Monte Carlo trials. (Left): RMSE for the relative position estimate. (Right): RMSE for the relative orientation estimate. Note that the CL-MSC-KF errors remain bounded, while the MSC-KF errors continuously increase.

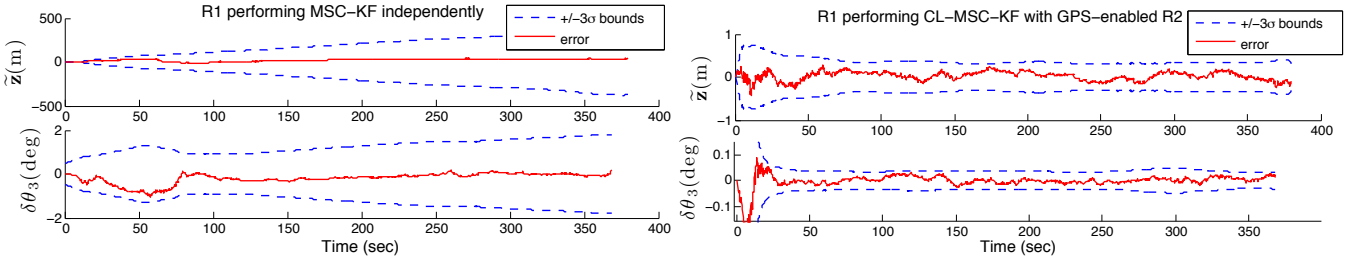

Fig. 5: Robot R_1 pose errors for the worst axis. (Left): R_1 localizes independently. The RMSE, along the trajectory, for the position is 129.7 m and for the orientation is 0.5 deg. (Right): R_1 performs CL-MSC-KF with R_2, when R_2 has access to GPS ($\sigma_{GPS} = 1$m). The RMSE in this case is 0.3 m for position and 0.05 deg for orientation.

lower for the CL-MSC-KF algorithm. At the end of the trajectory, the CL-MSC-KF estimates are 58% more accurate in orientation and 60% more accurate in position, compared to the MSC-KF. In the second simulation, we evaluated the accuracy of the estimated relative transformation $(\mathbf{p}, \mathbf{C})$ between the robots, in order to validate the analysis presented in Section IV. The results in Fig. 4 indicate that in the CL-MSC-KF framework the errors in the relative transformation remain bounded, whereas in the MSC-KF the errors continually increase. This is because in the MSC-KF framework the commonly observed features are treated independently while in the CL-MSC-KF this information is exploited by appropriately processing such measurements as the observations of the common scene. Therefore, even though the global-pose estimates drift, the CL-MSC-KF is able to maintain accurate relative-pose estimates over the whole trajectory. This is clearly a desirable property for CL, since if the group of the robots can maintain an accurate estimate of their relative transformation, then when any one of them measures its global position (e.g., using GPS), all the robots will benefit.

We illustrate this case in the next simulation in which the robots perform CL-MSC-KF, while R_2 has access to periodic GPS measurements with uncertainty $\sigma_{GPS} = 1$ m. Figure 5 shows the performance improvement for the non-GPS enabled robot R_1 compared to how it performed on the same trajectory when localizing independently. Although R_1 is GPS denied, its pose accuracy significantly improves as if it had GPS since it collaborates with R_2 by sharing and processing common visual observations.

Finally, we evaluated the dependence of the accuracy of the pose estimates in the CL-MSC-KF framework on the number of features observed by both robots. For any number of features greater or equal to two the filter performance was not affected significantly. On the other hand, in the case of a single common feature observed over the whole trajectory, the accuracy of the pose estimates of the CL-MSC-KF degraded to the accuracy levels obtained when the vehicles perform MSC-KF independently (see Fig. 6). These results corroborate the analysis in Section IV, in that not all six d.o.f. of relative transformation are observable when only one common feature is viewed by both vehicles. In this case, the relative pose of the robots is unobservable, which prevents the filter from reducing the errors in the estimates of the full six d.o.f. relative transformation.

VI. Conclusions and Future Work

In this paper, we addressed the problem of cooperative localization (CL) for two robots using vision-aided inertial navigation with overlapping camera observations of a previously unknown scene. Specifically, we presented an

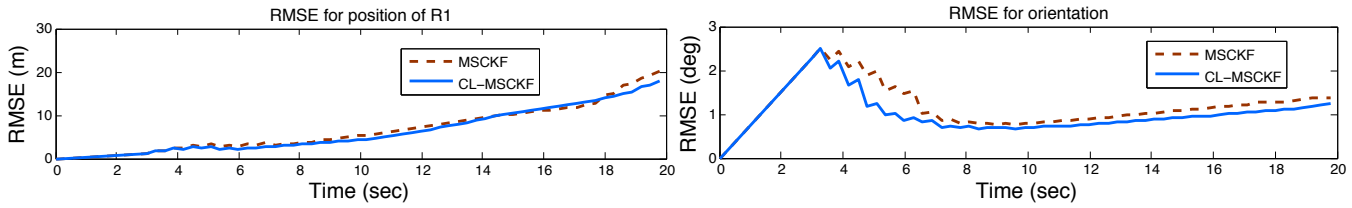

Fig. 6: Robot R_1 pose estimate errors, averaged over 100 Monte Carlo simulations for the case when both vehicles observe a single common feature over the whole path. (Left): RMSE for the position estimate of R_1. (Right): RMSE for the orientation estimate of R_1.

extension to the MSC-KF algorithm [15], termed the CL-MSC-KF, for jointly estimating the poses of both vehicles. Given observations of common scene features, the geometric constraints between the robots' pose estimates over a sliding time window were exploited by the filter to increase the localization accuracy for both of them. Our observability analysis showed that the robots must measure three common features over two or more steps in order to determine their six d.o.f. relative transformation. When the gravity vector is also observed, then only two common features are required. Finally, when only one common feature can be tracked over multiple time steps and the gravity vector is available, the relative transformation between the robots remains unobservable. The performance of the CL-MSC-KF was evaluated in simulations to demonstrate the validity of the proposed method and compare its accuracy with respect to single-vehicle localization.

In our future work, we plan to extend the CL-MSC-KF to consider distributed estimation architectures [10], instead of using the centralized approach adopted in this paper. We also plan to account for communication constraints between the robots and study the impact of quantization schemes on the filter's performance [16].

REFERENCES

[1] M. W. Achtelik, S. Weiss, M. Chli, F. Dellaert, and R. Siegwart. Collaborative stereo. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2242–2248, San Francisco, CA, Sept. 25–30, 2011.

[2] D. S. Bayard and P. B. Brugarolas. An estimation algorithm for vision-based exploration of small bodies in space. In *Proc. of the American Control Conf.*, pages 4589–4595, Protland, OR, June 8–10, 2005.

[3] R. W. Beard, T. W. McLain, D. B. Nelson, D. Kingston, and D. Johanson. Decentralized cooperative aerial surveillance using fixed-wing miniature UAVs. *Proc. of the IEEE*, 94(7):1306–1324, July 2006.

[4] D. D. Diel, P. DeBitetto, and S. Teller. Epipolar constraints for vision-aided inertial navigation. In *IEEE Workshop on Motion and Video Computing*, pages 221–228, Breckenridge, CO, Jan. 5–7, 2005.

[5] O. D. Faugeras and S. Maybank. Motion from point matches: Multiplicity of solutions. *Int. J. Comput. Vis.*, 4(3):225–246, June 1990.

[6] J. W. Fenwick, P. M. Newman, and J. J. Leonard. Cooperative concurrent mapping and localization. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 1810–1817, Washington. D.C., May 11–15, 2002.

[7] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. *Int. J. Robot. Res.*, 25(12):1243–1256, Dec. 2006.

[8] A. Howard, M. J. Mataric, and G. S. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 434–459, Lausanne, Switzerland, Sept. 30–Oct. 4, 2002.

[9] R. Kurazume, S. Nagata, and S. Hirose. Cooperative positioning with multiple robots. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 1250–1257, San Diego, CA, May 8–13, 1994.

[10] K. Y. K. Leung, T. D. Barfoot, and H. T. T. Liu. Decentralized cooperative simultaneous localization and mapping for dynamic and sparse robot networks. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3554–3561, Taipei, Taiwan, Oct.18–22, 2010.

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. of Comput. Vis.*, 60(2):91–110, 2004.

[12] L. Merino, J. Wilkund, F. Caballero, A. Moe, J. Dios, P. Forssen, K. Nordberg, and A. Ollero. Vision-based multi-UAV position estimation. *IEEE Robot. and Autom. Magazine*, 13(3):53–62, 2006.

[13] F. M. Mirzaei and S. I. Roumeliotis. A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation. *IEEE Trans. Robot.*, 24(5):1143–1156, Oct. 2008.

[14] A. I. Mourikis and S. I. Roumeliotis. Performance analysis of multirobot cooperative localization. *IEEE Trans. Robot.*, 22(4):666–681, Aug. 2006.

[15] A. I. Mourikis and S. I. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 3565–3572, Rome, Italy, Apr.10–14, 2007.

[16] E. J. Msechu, S. I. Roumeliotis, A. Ribeiro, and G. B. Giannakis. Decentralized quantized Kalman filtering with scalable communication cost. *IEEE Trans. Signal Process.*, 56(8):3727–3741, Aug. 2008.

[17] E. D. Nerurkar, S. I. Roumeliotis, and A. Martinelli. Distributed maximum a posteriori estimation for multi-robot cooperative localization. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 1402–1409, Kobe, Japan, May 12–17, 2009.

[18] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry for ground vehicle applications. *J. of Field Robot.*, 23:3–20, 2006.

[19] I. Rekleitis, G. Dudek, and E. Milios. Multi-robot collaboration for robust exploration. *Annals of Mathematics and Artificial Intelligence*, 31(1):7–40, Oct. 2001.

[20] S. I. Roumeliotis and G. A. Bekey. Distributed multirobot localization. *IEEE Trans. Robot. and Autom.*, 18(5):781–795, Oct. 2002.

[21] S. I. Roumeliotis and J. W. Burdick. Stochastic cloning: A generalized framework for processing relative state measurements. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 1788–1795, Washington, DC, May, 11-15, 2002.

[22] H. Sugiyama, T. Tsujioka, and M. Murata. Collaborative movement of rescue robots for reliable and effective networking in disaster area. In *Proc. of the Int. Conf. on Collab. Computing: Networking, Applications and Worksharing*, pages 7–15, Los Alamitos, CA, Dec. 19–21, 2005.

[23] S. Thrun and Y. Liu. Multi-robot SLAM with sparse extended information filers. *Int. J. Robot. Res.*, 15:254–266, 2005.

[24] N. Trawny and T. Barfoot. Optimized motion strategies for cooperative localization of mobile robots. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 1027–1032, New Orleans, LA, Apr. 26–May 1, 2004.

[25] N. Trawny and S. I. Roumeliotis. Indirect Kalman filter for 3D attitude estimation. Technical Report 2005-002, University of Minnesota, Dept. of Comp. Sci. & Eng., Mar. 2005.

[26] N. Trawny, X. S. Zhou, K. X. Zhou, and S. I. Roumeliotis. Inter-robot transformations in 3D. *IEEE Trans. Robot.*, 26(2):226–243, April 2010.

[27] S. B. Williams, G. Dissanayake, and H. F. Durrant-Whyte. Towards multi-vehicle simultaneous localization and mapping. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 2743–2748, Washington, DC, Sept. 30–Oct. 4, 2002.