# Architecture for Asymmetric Collaborative Navigation

Zhen Zhu[1], Stergios Roumeliotis[2], Joel Hesch[2], Han Park[1], Don Venable[3]

[1]Northrop Grumman Corporation
Woodland Hills, CA

[2]University of Minnesota
Minneapolis, MN

[3]Air Force Research Laboratory
WPAFB, OH

*Abstract*— Under the Air Force Research Laboratory (AFRL) Collaborative Robust Integrated Sensor Positioning (CRISP) program, Northrop Gruman Corporation (NGC) is designing and building a collaborative navigation system for multiple airborne platforms. The collaborative navigation architecture has been designed to take advantage of AFRL's Layered Sensing construct which enables platforms to share information. In particular, the ability to share GPS, relative range, imagery, geo-registered maps, and other measurements opens up many opportunities to improve the navigational accuracy and the robustness to GPS-denied conditions.

In the CRISP program, the collaborative navigation system is being designed to be more robust and accurate by leveraging the asymmetry in the sensing, computation, and communication capabilities of disparate platforms. For example, the system takes advantage of higher performing sensors on the high-flyer (HF) platform, which are less susceptible to jamming, and cameras that generate larger sensor footprint and higher resolution images of the terrain. The low-flyers (LFs) have poorer navigation sensors, are more likely to be jammed, and have a more limited view of the terrain. Under this scenario, the HF may assist one or more LFs such that they, too, can have similar accuracy as the HF in a GPS-denied environment.

*Keywords- layered sensing, vision-aided, asymmetric, collaborative navigation*

## I. INTRODUCTION

In the layered sensing paradigm introduced by AFRL, every platform will be networked such that they can share information with every other platform. Layered sensing offers the promise of new revolutionary capabilities in cooperative operations. In particular, navigation may benefit since aircraft with lesser capability or in disadvantageous conditions (e.g. GPS-denied) may be able to take advantage of assets on another aircraft. The first phase of AFRL's CRISP program seeks to explore concepts for improving the navigation performance if aircraft can share information such as GPS (for relative positioning), relative ranging, imagery, and inertial measurement unit (IMU) data. The concept is being proven using a government furnished (GF) dataset provided by AFRL. The second phase of the program is to develop a real-time system which will be tested in-flight. This paper will describe the system architecture and algorithms which currently being developed by NGC and teammate, University of Minnesota (UMN).

The CRISP concept deviates from the usual peer-to-peer cooperative navigation. In peer-to-peer systems, such as the ones reported in [1] and [2], the vehicles are often equipped with similar sensors. In CRISP, however, there is a high-flyer (HF) and a low-flyer (LF) each with different capabilities. The HF, a persistent surveillance platform, is assumed to have higher quality sensors, including IMU and camera, and less susceptible to GPS jamming. The LF(s) is expected to have lower quality navigation sensors and cameras, and is more likely to be GPS-denied. The NGC team has been designing an architecture and algorithms that fully exploit these asymmetries in the processing, sensing, and communication capabilities between the HF and LF(s).

## II. SYSTEM ARCHITECTURE

The NGC CRISP physical architecture is shown in Figure 1. Instead of a point-to-point communication architecture, it also includes a ground base station. For collaborative navigation, geo registered imagery is shared between the HF and the LFs. In a point-to-point architecture, it may require datalink bandwidth greater than a few MHz between a HF and each LF. Availability of a dedicated high-speed datalink will be subject to a number of limitations. For instance, it may require directional gain pattern for the LF antenna, which becomes challenging for small UAVs. Alternatively, the proposed architecture exploits the current asymmetry in datalinks – the downlink bandwidth to the ground is much greater than the uplink bandwidth. Hence, in this architecture, the imagery from both LF and HF are transmitted to the ground, which is part of the data product already being collected today. Computationally intensive operations, including inter-platform feature matching are now accomplished on the ground. Thus, it is more aligned with the current military datalinks and concept of operations. It is also suited to support multiple LFs. However, there is increased complexity and reduced reliability because of the increase in the number of datalinks in the architecture.

$$x_{C_i}^T = [{}_G^{C_i}\bar{q}^T, \; {}^G p_{C_i}^T] \tag{2}$$

where ${}_G^{C_i}\bar{q}$ is a unit quaternion representing the orientation of the global frame (G) with respect to the camera frame ($C_i$), and ${}^G p_{C_i}^T$ is the position of $C_i$ with respect to the global frame. Whenever the HF records an image, we save the corresponding camera pose, $x_{C_i}$ which can be used later to initialize new features into the map. The vehicle state is given by the following vector

$$x_R^T = [{}_G^R\bar{q}^T \; b_g^T \; {}^G v^T \; b_a^T \; {}^G p_R^T] \tag{3}$$

where $b_g$ and $b_a$ are the gyroscope and accelerometer biases, which are time-varying and modeled as random-walk processes, driven by zero-mean white Gaussian noise; ${}^G v$ is the velocity of the vehicle expressed in a global frame and ${}_G^R\bar{q}$ and ${}^G p_R$ represent attitude and position of the HF. Lastly, the map is comprised of N landmark features, where the j-th feature state in the map is denoted using the 3x1 position vector $x_{L_j} = [\,{}^G p_{L_j}]$. One potential issue with SLAM implementation is that the computational complexity grows quadratically with the number of features [3]. This issue is address by applying the Power SLAM algorithm [4]. In PowerSLAM, a two-step novel estimation process is as followed: (i) use dimensionality-reduction techniques to reduce the rank of the covariance correction term during updates, and (ii) determine and apply low-rank covariance corrections only along the most informative state space directions.

With the HF pose estimation optimized by PowerSLAM, geo-registered imagery or map can be created and processed in different ways. For example, images can be ray-traced to a terrain model, such as a digital terrain elevation data (DTED), where the 3D position of each pixel can be estimated. However, the accuracy of such geo-registration is limited by camera pose accuracy, and the resolution and accuracy of available DTED. Consequently, accuracy of LF navigation is also limited accordingly when using this type of maps. Alternatively, a dense 3D map can be created by tracking visual features in continuous frames by using bundle adjustment or similar techniques. As a result, the accuracy of map is no longer directly affected by DTED. However, a dense 3D map based on this approach requires significant amount of computational power, which may not be feasible for real-time implementation. In this architecture, geo-registered HF imagery will be transmitted to the ground station, and used to aid LF navigation, which will be further discussed in the following sections.
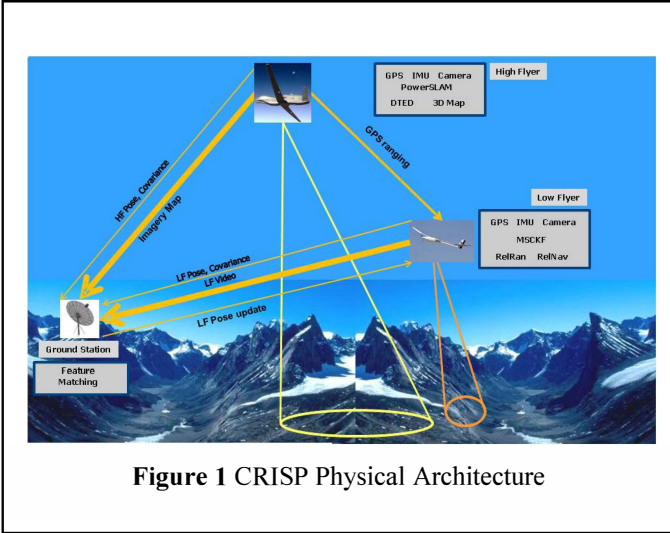


**Figure 1** CRISP Physical Architecture

## III. SUBSYSTEMS

### A. HF Simultaneous Localization and Mapping

Given the HF's large sensor footprint, and processing and storage capabilities, it is well suited to create a high-fidelity geo-registered imagery map by performing visual Simultaneous Localization and Mapping (SLAM). The imagery map will serve two purposes. First, whenever the HF loses GPS, it will be able to maintain an accurate own-ship pose estimate with bounded errors. Second, the HF will share the map with the LFs to improve their localization accuracy. Figure 2 illustrates a block diagram of the HF components. The GPS receiver and IMU in the HF supply raw measurements for the GPS/IMU navigation solution. The raw GPS pseudorange and carrier phase measurements will also be used in a relative navigation solution for LF initialization. These measurements, along with HF pose data, are transmitted to the LF via the HF-LF datalink. A camera is synchronized with GPS time. In reality, the camera may be installed on a gimbaled platform, which has not been accounted for in this block diagram.

The core of the SLAM algorithm is an extended Kalman filter (EKF), which estimates a sliding window of past camera poses, the current vehicle pose, and a sparse map of the features in the environment. The SLAM state vector is

$$x = [x_{C_1}^T, \dots, x_{C_N}^T, x_R^T, x_{L_1}^T, \dots, x_{L_M}^T] \tag{1}$$

where $x_{C_i}, i = 1, \dots, N$ is the 7x1 dimensional vector describing the i-th camera pose, $x_R$ is the 16x1 dimensional vector representing the state of the vehicle at the current time step, and $x_{L_i}, i = 1, \dots, M$ is the 3x1 dimensional vector describing the 3D position of the j-th feature in the estimated map. More specifically, the camera pose is defined by the following state vector
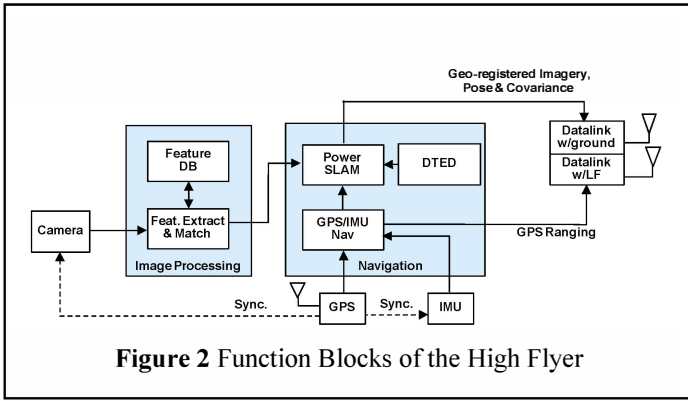
**Figure 2** Function Blocks of the High Flyer

## B. LF Own-Ship Navigation

LF navigation is initialized with standalone GPS-IMU solution or relative navigation (RelNav) from differential GPS solution when available. The LF navigation own-ship algorithm is capable of providing standalone solution when GPS becomes denied, and incorporating updates from HF. Due to limited processing capabilities and smaller sensor footprints, the LFs are less suited to build their own visual terrain maps. Instead, they use common visual features between frames to constrain the motion estimation. Therefore, LF vision-based navigation algorithm is considered a "visual odometry" system instead of a SLAM [5]. LF navigation is based on a Multi-State Constraint Kalman Filter (MSC-KF) [6]. The MSC-KF is referred to as an opportunistic algorithm, in that it uses any matched image features, but does not attempt to build a database from them for long-term mapping. A large number of point features, also known as opportunistic features (OFs) can be observed and tracked between images. The objective of MSC-KF is to impose constraints between consecutive poses and reduce the rate error increase. To efficiently process these OFs without loss of information, the MSC-KF's estimated state vector is augmented to include a sliding window of the camera poses where each image was taken. The OF constraints are enforced in a mathematically rigorous and efficient way, such that hundreds of constraint updates can be processed with linear computational cost, in real time by the MSC-KF. A block diagram of LF navigation system can be found in Figure 3.

Stochastic clones of the past camera pose estimates are created that will account for the maximum time window (or equivalently, number of images) in which a given feature can be tracked. For example, based on the flight profile of the LF in the GF data, we have observed that features remain within the camera's Field of View (FoV) for approximately five to six frames, hence, the MSC-KF state vector is constructed as

$$x_k^T = [\, {}^{C_k}_G \bar{q}_k{}^T \ b_{gk}^T \ {}^{G} v_k^T \ b_{ak}^T \ {}^{G} p_k^T \ {}^{C_{k-1}}_G \bar{q}_{k-1}{}^T \ {}^{G} p_{k-1}^T \ \cdots$$

$$ {}^{C_{k-K}}_G \bar{q}_{k-K}{}^T \ {}^{G} p_{k-K}^T \,]^T \qquad (4)$$

where ${}^{C_k}_G \bar{q}_k$ , ${}^{G} p_k$ , and ${}^{G} v_k$ denote the quaternion of orientation, position, and velocity of the camera at time step k,

while $b_{gk}$, and $b_{ak}$ are the gyro and accelerometer biases, respectively. Each clone comprises the orientation and position of the camera at the time that a previous image was recorded (i.e., k-1 to k-K). A standard kinematic model of motion is employed for propagating the states.

The OF tracks are obtained by matching Harris corners in consecutive frames. Down selection of feature matches is necessary to detect and reject false matches. Especially in the GF data, the LF flies over both urban and rural areas, where some portion of the imagery data contains only forests. Due to the terrain condition, low image quality and low frame rate, it is likely to have a large number of incorrect matches, which makes this problem much more challenging. The down selection step implemented in this paper utilizes the five point algorithm [7] combined with RANSAC [8]. In addition, when a track is processed, a Mahalanobis distance test is exercised to verify that the measurement residual for that track follows a $\chi^2$ distribution. Even with the above steps, it is still possible for incorrect tracks to pass the test. Additional layers of robustness have to be added to minimize the error caused by outliers.

Own-ship navigation with a visual odometry system is going to drift with time and distance traveled. In order to further limit the drift of LF pose, each LF will exploit the geo-registered imagery map collected by the HF. Ideally, matching features detected in the LF images to corresponding features from HF can make the position and orientation of the LF fully observable. The relative pose between HF and LF is estimated based feature correspondence, such that the drift can be bounded in GPS-denied conditions. This type of update does require that the sensor footprints of the HF and the LF overlap. However, the HF and LFs do not have to fly simultaneously observing the same area.

The RF datalink between LF and HF can be also used for relative ranging estimates. Although it has been discovered that relative ranging integrated with IMU may not be sufficient for stable relative pose computation in the GF data, it can be added as an additional input into MSC-KF.
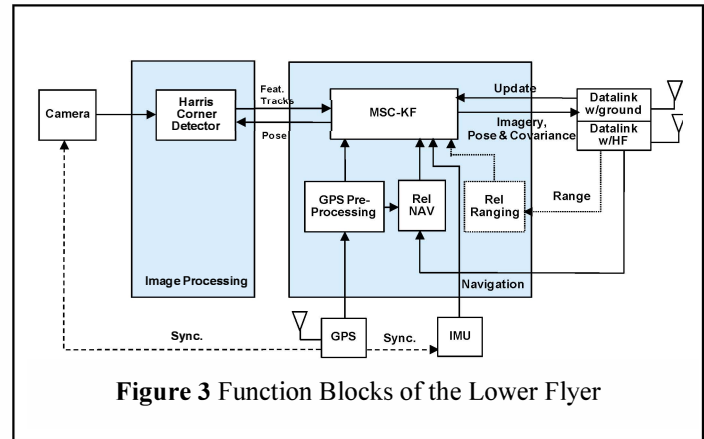


**Figure 3** Function Blocks of the Lower Flyer

## C. Ground Station Block Diagram

The ground station receives imagery from both HF and LF. It is noted that in the GF data, the footprints of HF and LF do not always overlap. When they do, there can be only a limited number of feature correspondences between each pair of LF and HF images. It does not necessarily require a dense 3D map to make use of these matched features. In the meanwhile, we have observed that the features from a single HF image are not sufficient for an accurate update for a single LF pose. In that case, a LF image may be paired with multiple HF images to gain more observables. Without using a dense map, 3D location of the matched features can be jointly estimated together with the relative HF-LF pose.

In order to find more matching features between the LF and HF it is desirable to have a search method that would find as many potential HF matches as possible for a given LF image, while maintaining robustness to false positives. Furthermore, the method should be efficient in order to support real-time updates. In addition, although the image matching algorithm may make use of the LF's prior pose estimates, it should not depend on them. Thus, even in the absence of any information about the LF's pose, the visual information should still be utilized to generate hypotheses for its location. The matching algorithm is based a vocabulary tree [9], which is a purely appearance-based system for matching images using SIFT descriptors [10] for image comparisons. The vocabulary tree is usually constructed ahead of time with training images. As shown in Figure 4, a set of HF images $S$, is filled into the tree in real time. Given a query LF image $I\_Q$, the vocabulary tree returns a list $L\_VT$ of the top $N$ most visually similar images to $I\_Q$ within $S$. As the approach is purely appearance-based, it will return some image matches that are visually similar but do not actually correspond to the same location, especially for visually ambiguous areas such as forests. In the second stage, a set of a geographic and geometric consistency tests are applied to eliminate false matches, which leverage approximate pose estimates of the HF and LF when available.

When matched images are selected, HF-to-LF relative pose updates become available based on common observations of visual features. However, there are still a few more challenges. First, the LF and HF images of the same area do not necessarily look "similar". For example, the GF data, the HF and LF cameras largely differ in blurriness, resolution and field of view, which result in substantial dissimilarity in image appearance. In addition, the images were recorded at different times of the year, and the seasonal changes in the foliage cause visual disparities. Change of vegetation, on-going construction and direction of shade can be observed in the images. As a result, the appearance-based feature matching approaches such as SIFT keys will limit the number of accurate matches. To address this issue, a patch correlation based method has been used in addition to descriptor matching, which significantly increases matching accuracy. Second, the HF camera has a narrower field of view and takes images at a greater distance compared against the LF. With few matches concentrated in a narrow FoV in HF camera, which is responsible for a possible degenerate geometry condition for LF pose estimation. Finally, accuracy of relative pose update is obviously dependent on the intrinsic and extrinsic calibration of both cameras.

Even with all the challenges, there exist a few viable approaches that can constrain the relative LF pose. For example, common features from each pair of LF-HF images can provide five degrees of freedom (5-DOF) constraints on the relative pose without knowing the location of features. The relative orientation and translation (up to scale) of LF is estimated in a maximum likelihood (ML) sense by solving a nonlinear least-squares problem to minimize the sum of the squared reprojection errors in both the HF and LF images. A 6-DOF method can jointly estimate relative pose and the location of common features, with the n-point pose determination (PNP) on the LF [11]. Robustification techniques, including those introduced in [12] and [13], have been applied to both methods. As afore mentioned, 3D positions of these features can be computed with this method, which is a different approach from extracting feature location from a dense 3D reconstruction.

Although feature matching, computation of relative pose and integration into the LF MSC-KF can be implemented in different approaches, it generally requires substantial amount of data being transmitted from HF and LF. The ground station can be equipped with directional antennas that will enable high speed datalinks from both the HF and LF. Finally, the HF-to-LF updates can be uploaded to LF and integrated into the MSC-KF framework.
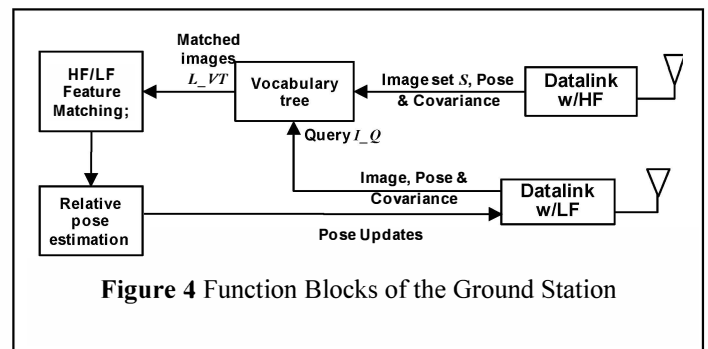


**Figure 4** Function Blocks of the Ground Station

## IV. RESULTS AND ONGOING EFFORT

LF position and attitude are estimated for using GF data by incorporating both MSC-KF and HF-LF updates. As part of the ongoing effort of further improving the accuracy, results from a portion of the data that contains 652 seconds over a distance of approximately 43 km are represented in this section as an example. During the first 30 seconds GPS was available for initialization. The LF navigates with MSC-KF from second 30 to 652. The HF-LF updates were available intermittently, from second 30 to 104, 260 to 334, 432 to 546 and 635 to 652, respectively.

**Figure 5** Orientation Errors Using MSC-KF and HF-to-LF Position Updates



**Figure 6** Position Errors Using MSC-KF and HF-to-LF Position Updates

The MSC-KF attains accurate relative orientation between successive images, and the HF-to-LF position updates provide global position information that is utilized to bound the position error growth of the MSC-KF. An added benefit of the HF-to-LF position updates is that they also provide sufficient information to bound the yaw uncertainty, which continually increases when employing the MSC-KF alone. The position and attitude accuracy results of MSC-KF compared against MSC-KF with HF-to-LF position-only constraints are listed in TABLE I. The average roll and pitch uncertainties are 1.12 and 1.27 deg 3σ, respectively, while the average yaw uncertainty is 1.74 deg 3σ. The average position uncertainties along the least accurate axis is 83.73 m 3σ, excluding the camera shutoff period. In comparison to both of the previous methods, the position uncertainty along the z-axis is 105 m 3σ or lower for

the majority of the trajectory, with the exception of the portion between [530, 600] sec when the LF is changing altitudes.

TABLE I SUMMARY OF RESULTS, MSC-KF VS. MSC-KF WITH HF-TO-LF POSITION-ONLY UPDATES

|  | MSC-KF w. HF-to-LF updates |
|---|---|
| Orientation RMSE roll | 0.39 (deg) |
| Orientation RMSE pitch | 0.60 (deg) |
| Orientation RMSE yaw | 0.69 (deg) |
| Orientation Avg. 3σ roll | 1.12 (deg) |
| Orientation Avg. 3σ pitch | 1.27 (deg) |
| Orientation Avg. 3σ yaw | 1.74 (deg) |
| Position RMSE x-axis | 27.61 (m) |
| Position RMSE y-axis | 23.84 (m) |
| Position RMSE z-axis | 37.22 (m) |
| Position Avg. 3σ x-axis | 83.73 (m) |
| Position Avg. 3σ y-axis | 68.82 (m) |
| Position Avg. 3σ z-axis | 80.51 (m) |

The results presented in this section are based on MSC-KF incorporating position correction computed with the 6-DOF HF-LF updates, where a single LF images is matched against multiple HF images. It is considered a loosely coupled approach. Loosely-coupled methods have the advantage that they simplify the filtering equations as well as the outlier-rejection process. However, they may introduce challenges when trying to process measurements across multiple time-steps (e.g., when creating a synthetic aperture using HF images recorded from different viewing directions). In contrast, tightly-coupled approaches seek to utilize the raw measurements (LF and HF camera observations to features matches across multiple images) directly in the filter while at the same time taking advantage of the DTED information. Tightly-coupled methods typically provide the flexibility required for dealing with measurements from different time instants. However, some potential drawbacks exist, which include increased complexity and the necessity to estimate additional parameters in the filter state vector (e.g., the 3D coordinates of each measured point). Tight coupling approach of HF-LF update is currently being explored.

This document is approved for public release; distribution is unlimited, PA Case # RY-12-0229

REFERENCES

[1] Mitch Bryson and Salah Sukkarieh, "Decentralised Trajectory Control for Multi-UAV SLAM," Proceeding of the 4th International Symposium on Mechatronics and its Applications (ISM07), Sharjah, U.A.E. March 26-29, 2007.

[2] Mohiuddin Ahmed, Md. Raisuddin Khan, Md. Masum Billah and Soheli Farhanam, "A Novel Navigation Algorithm for Collaborative Multi Robots," European Journal of Scientific Research 41(3): 472-481, 2010.
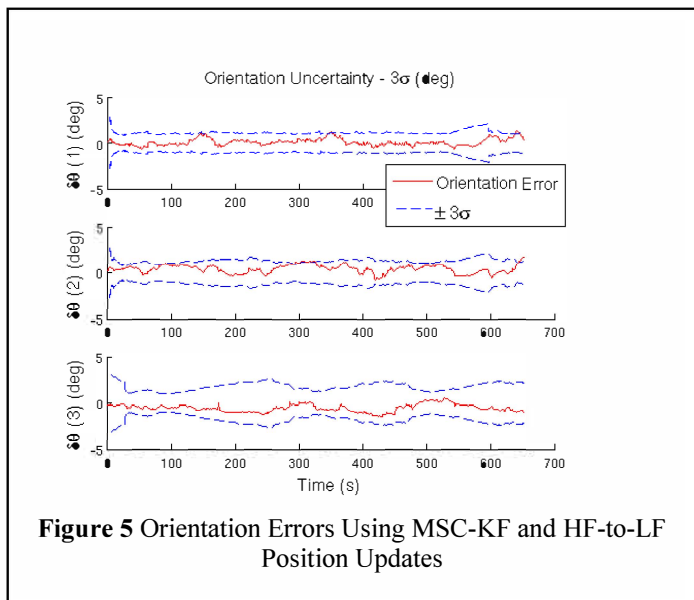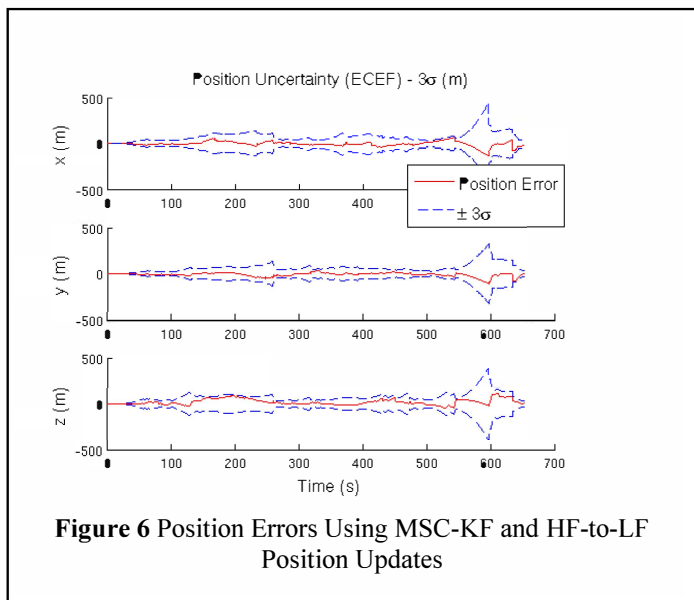
[3]  Andrew J. Davison, "Real-Time Simultaneous Localisation and Mapping with a Single Camera," Proceedings of IEEE International Conference on Computer Vision (2): 1403 - 1410, October 2003

[4]  E. D. Nerurkar and S. I. Roumeliotis, "Power-SLAM: a linear-complexity, anytime algorithm for SLAM". International Journal of Robotics Research, 30(6): 772-789, May 2011.

[5]  Brian Williams and Ian Reid, "On Combining Visual SLAM and Visual Odometry," Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 3494 – 3500, May 2010

[6]  A.I. Mourikis, N. Trawny, S.I. Roumeliotis, and A.E. Johnson, A. Ansar, and L. Matthies. "Vision-aided inertial navigation for spacecraft entry, descent, and landing", IEEE Transactions on Robotics, 25(2): 264-280, 2009

[7]  D. Nistér, "An efficient solution to the five-point relative pose problem," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 26(6):756-770, June 2004.

[8]  Martin A. Fischler and Robert C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," Comm. of the ACM **24** (6): 381–395, June 1981.

[9]  D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pages 2161-2168, June 2006.

[10]  David G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.

[11]  Joel A. Hesch and Stergios I. Roumeliotis, "A Direct Least-Squares (dls) Solution for PnP," *Proc. of the Int. Conf. on Computer Vision*, Barcelona, Spain, November 6-13, 2011.

[12]  Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon "Bundle Adjustment - A Modern Synthesis," Proceedings of the International Workshop on Vision Algorithms: Theory and Practice (ICCV '99), Bill Triggs, Andrew Zisserman, and Richard Szeliski (Eds.). Springer-Verlag, London, UK, 298-372.

[13]  R. I. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision", Cambridge University Press, ISBN: 0521540518, 2004