

Field-Testing of a UAV-UGV Team for GNSS-Denied Navigation in Subterranean Environments

Jason Gross, Matteo De Petrillo, Jared Beard, Hayden Nichols, Tommy Swiger, Ryan Watson, Connor Kirk, Cagri Kilic, Jacob Hikes, Emily Upton, Derek Ross, Matt Russell, Yu Gu, and Christopher Griffin *West Virginia University*

ABSTRACT

We present the system design, navigation algorithm formulation, and field-testing results for a cooperative unmanned ground vehicle (UGV) and unmanned aerial vehicle (UAV) team designed for operation in a GNSS-denied subterranean environment. Central to the presented system design, to support longer UAV mission duration, the sensing and computation performed on-board the UAV is assumed to be limited in order to reduce UAV payload mass, and the UGV assumes the responsibility of mapping the unknown environment and tracking the state of the UAV. To achieve this, the UGV is instrumented with an omnidirectional fish-eye camera, a 128 channel 3D scanning LIDAR, and an ultra wideband (UWB) ranging radio that is linked with a radio on the UAV. In addition, the UAV shares inertial measurement unit and laser altimeter data to the UGV over a local WiFi network. This paper outlines the UAV localization algorithm for the UAV/UGV team and presents results from multiple flight tests that were conducted in a large indoor flight testing tunnel environment that is instrumented with a VICON motion capture system for providing reference solutions.

1 INTRODUCTION

The use of robotic systems in subterranean environments has the potential to enable many important applications including: search and rescue [1], disaster response [2], maintenance monitoring [3], border security [4], and waste cleanup [5], among others. These environments pose many operational challenges for autonomous robots including: poor lighting conditions, harsh traversability, lack of access to Global Navigation Satellite Systems (GNSS) signals, unknown/unstructured obstacles, and a high likelihood of encountering unfavorable conditions such as water, smoke, and dust. These challenges and the potential benefits for the use of robotic systems in these environments is perhaps best exemplified by the fact that in 2018 a DARPA SubT challenge was established to promote breakthrough technology development for these environments [6].

A key enabling technology for the use of autonomous robots in subterranean environments is the ability of the robot to localize itself within a map. Recently, visual-inertial localization technologies for use in GNSS-denied environments on UAVs has been demonstrated impressive performance levels. For example, using a factor-graph framework Forster et al. [7] developed the pre-integrated IMU factor, which enables real-time UAV visual-inertial localization. This was demonstrated to achieve decimeter-level localization errors in an indoor experimental test. More recently, Nisar et al. [8] have further taken into account UAV dynamic model constraints and actuation signals to improve performance for GNSS-denied localization to the centimeter-scale. Likewise, these technologies have been extended for use in visually degraded environments, such as dust or smoke, by leveraging thermal cameras [9, 10]. While very promising localization technology, challenges remain regarding onboard collision avoidance and mapping.

To address these challenges and limitations, this paper presents a cooperative UAV and UGV system, in which a UGV is available to produce a 3D map of the unknown environment and provide the sensing needed to localize the UAV within the map. This paper substantively builds upon our previous workshop poster [11], which only provided a conceptual overview of our system design, and herein we detail the UAV's navigation sensor system, algorithm formulation, and the results of experimental testing. This paper offers the contributions of: 1) detailing a novel cooperative UAV/UGV strategy for the exploration of subterranean environments, and 2) providing the reader insight regarding the benefits of our presented approach through the analysis of field-testing results.

The rest of this paper is organized in the following manner. Section 2 details the sensor systems of the UGV and UAV. Section 3 details the navigation algorithm. Section 4 shows the performance of the proposed algorithm during a set of experimental tests. Finally, Section 5 highlights some conclusions of the paper and our plans for improving the system.

2 UGV AND UAV SYSTEM DESIGN

To support 3D Simultaneous Localization and Mapping (SLAM) and to track the state of a multi-rotor UAV, the UGV system is outfitted with a host of sensors. As shown in Figure 1, the UGV's main drive chassis is a Clearpath Robotics Husky UGV

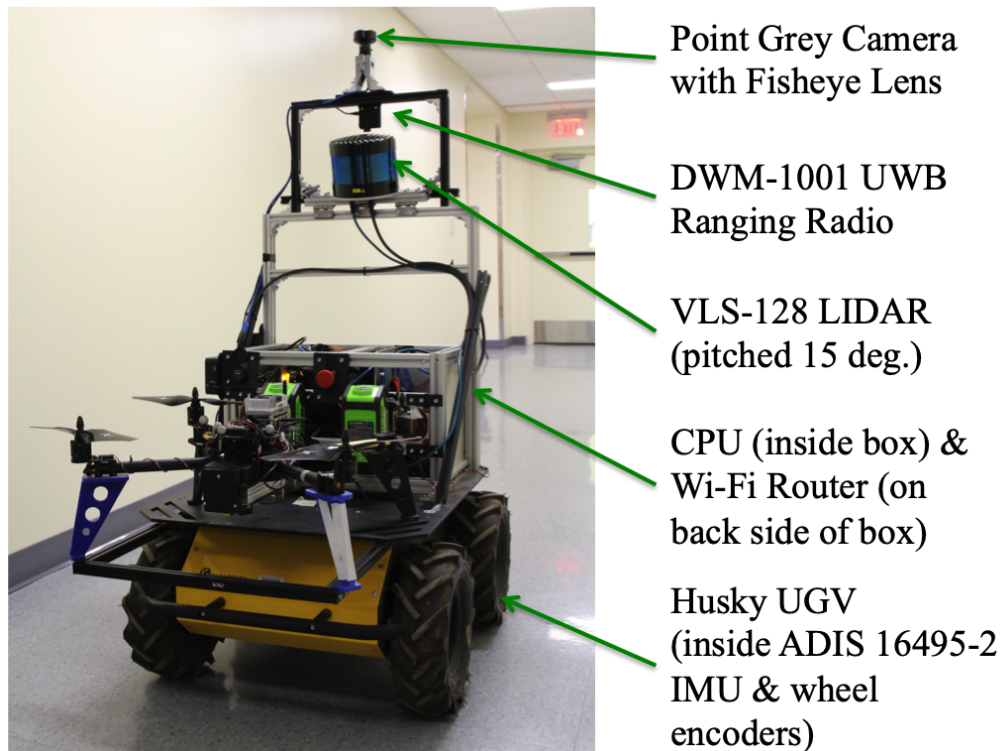


Figure 1: Subterranean UGV (named Badger) sensor suite setup. The Badger robot is primarily responsible for producing a map of the surrounding environment and tracking the state of the UAV within the map.

platform [12] that is outfitted with a custom computer box and custom sensor mounting stack. For tracking the state of the UAV, three primary sensing modalities are included on the UGV. First, a Point Grey camera that is outfitted with a fish-eye lens is mounted on the top of the sensor stack. Second, an ultra wideband (UWB) (DWM-1001) ranging radio is mounted on the UGV to range with a companion UWB radio mounted on the UAV. Finally, a Velodyne VLS 128 channel 3D LIDAR with -25 to 15 degree vertical field of view and 300 meter. In addition, the 3D LIDAR is tilted 15 degrees to provide a 30 degree upward and 10 degree downward field of view on one side of the UGV. To support 3D SLAM, the UGV is also outfitted with a tactical grade Inertial Measurement Unit (IMU) (Analog Devices ADIS 16495-2) and wheel encoders. All software running on the UGV operates under the Robot Operating System (ROS) [13] and the main computer host an i7-7700 processor.

A sensor diagram of the UAV is shown in Figure 2. The onboard computer of the UAV is an Nvidia Jetson, which is coupled with a PixHawk 4 autopilot that has an Analog Devices ADIS 16495-2 IMU external IMU integrated. In addition, the UAV carries the companion UWB for ranging measurement to the UGV. On the bottom of the UAV, a LidarLite laser altimeter and Px4Flow optical flow sensor are integrated. In this study, the optical flow data were not integrated for use in navigation.

The UAV and UGV are Wi-Fi enabled to facilitate communication via a local network. As such, both UAV and UGV are able to access sensor messages under a single instance of Robot Operating System (ROS) running on the UGV's computer. This enables control of the UAV from the UGV as well as sharing the autopilot's IMU data and laser altimeter data to support relative localization.

3 NAVIGATION ALGORITHM FORMULATION

In this robot/drone teaming concept, it is assumed that the UGV first traverses the unknown environment, to the extent that it is able, in order to produce a 3D obstacle map via SLAM. Since we are using SLAM on the UGV as an out of the box capability, we do not detail the SLAM algorithm in this paper and instead detail the relative localization of the UGV with respect to the UAV.

Once the UGV has estimated a 3D map, the UAV is deployed and used to explore the known volume, while offering an 'eye in the sky' perspective that is more favorable for finding for targets of interest (e.g., humans in need of rescue). At this

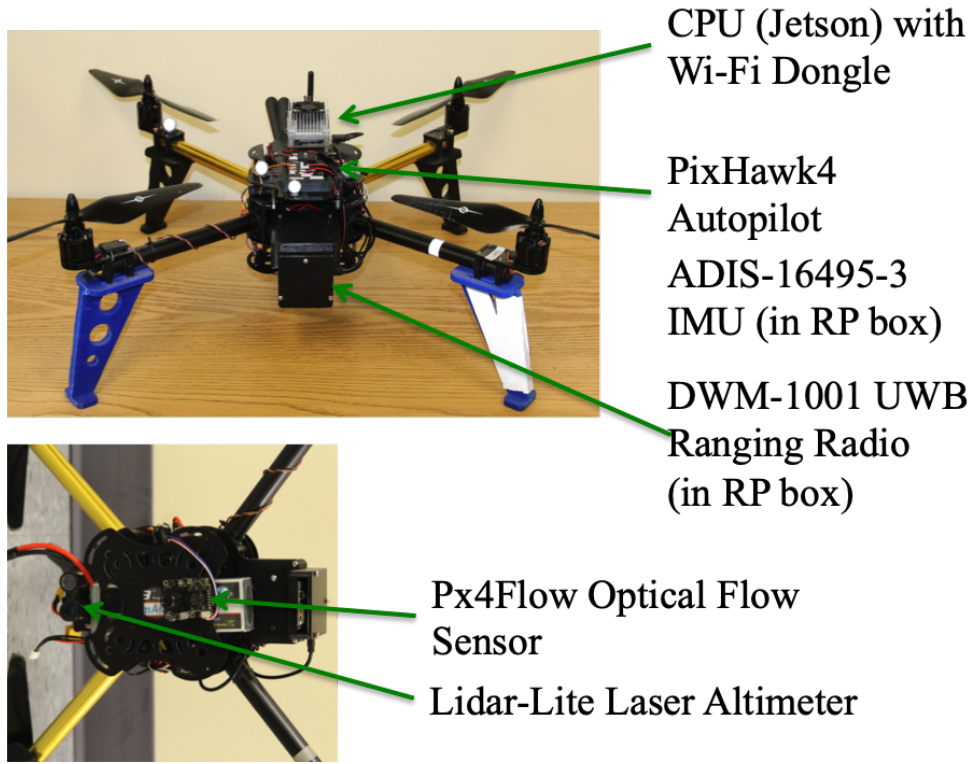


Figure 2: Multi-rotor UAV used for experimental evaluation.

point, it is assumed the UGV is static, such that the UAV's navigation state (position, velocity, and attitude) with respect to a local coordinate system that is centered on the UGV is the state of interest to estimate. This allows the UGV's 3D map to be used by the UAV for mission planning.

The premise of navigation architecture of the UAV relative to the UGV is that neither of the two primary sensing modalities (i.e., fish-eye camera and LIDAR) used for estimating the UAV's position is available for the entire desired flight envelope of the UAV. Therefore, to ensure reliable localization, it is required to have a cooperative motion planner of the UAV and UGV that takes into account sensor uncertainties and sensor fields of view. The development of the cooperative planning under uncertainty will be the basis of our future work with this system. Likewise, this concept will be extended to operations when the UGV is moving or when more than one UAV is assisting in the mission.

Figure 3 shows an overview of the GNSS-denied navigation approach adopted. To fuse the available measurements, an error-state Extended Kalman Filter (EKF) is employed. As shown in Figure 3, the UAV's onboard inertial navigation is integrated to predict the navigation state, and five different asynchronous information sources are available for sensor fusion within an EKF to determine the state of the UAV with respect to the UGV. In this framework, there is no assumption made that the measurement sources are always available. In particular, the fields of view of the LIDAR and fish-eye camera are largely complimentary and tracking the UAV with these sensors is not always successful. As such, the estimator must be able to process any sub-set of measurement updates as they are available.

As listed in Eq. 1, the state vector of interest consists of the attitude, position, and velocity of the UAV in a local North-East-Down, navigation frame, whose origin is centered at the UGV.

$$\hat{x} = [\phi_b^n, \theta_b^n, \psi_b^n, v_N, v_E, v_D, r_N, r_E, r_D]^T \quad (1)$$

However, the error-state EKF, estimates small deviations (δ) of the state vector of interest along with IMU sensor accelerometer biases (b_a) and rate gyroscope biases (b_g), as listed in Eq. 2,

$$\hat{\delta x} = [\gamma(1)_b^n, \gamma(2)_b^n, \gamma(3)_b^n, \delta v_N, \delta v_E, \delta v_D, \delta r_N, \delta r_E, \delta r_D, b_{ax}, b_{ay}, b_{az}, b_{gx}, b_{gy}, b_{gz}]^T \quad (2)$$

where γ_b^n represents the small-angle attitude error.

Within the error-state EKF, the states and error-states in Eqs. 1 and 2 are predicted at a high-rate (50 Hz) with inertial navigation, and then updated, as measurements are available. The available measurements include: a laser altimeter onboard

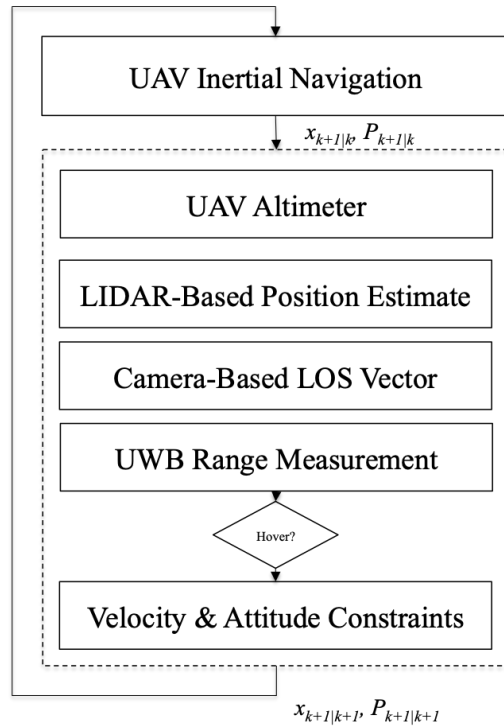


Figure 3: Block diagram of the GNSS-denied UAV-UGV navigation system. All software is running on the UGV.

the UAV, the LIDAR-based tracking of the UAV from the UGV, camera-based tracking of the UAV from the UGV, a UWB ranging measurement between the UAV and UGV, and motion constraints of the UAV whenever a hovering state is detected.

The six subsections below outline the specific formulations adopted for each of the primary navigation information sources used for fusing in the EKF by detailing the nonlinear state updates and linearized error-state transition model Φ adopted for INS, as well as the observation models z , and observation sensitivity matrices, H , for each measurement update source. For details on implementing an error-state EKF the reader is referred to Groves [14].

3.1 UAV Inertial Navigation

Our adopted implementation of inertial navigation is based upon the error-state formulation within the North-East-Down navigation frame that is outlined in Groves [14]. However, due to the desire to estimate position in local coordinate frame without any knowledge of the UAV's absolute position on the Earth, as well as, the relatively low velocities of the UAV assumed for this application, it was elected to simply ignore the contributions of the Earth's rotation and craft-rate terms within the Inertial Navigation Systems (INS) mechanization. These assumptions effectively assume that INS is being carried out in an inertial frame, which although incorrect, was deemed negligible for the proposed application.

For updating attitude, whenever Earth's rotation and craft-rate terms are neglected, the attitude update is given as

$$\hat{C}_{b,k+1|k}^n = \hat{C}_{b,k|k}^n (I_{3 \times 3} + \Omega_i^b \delta t) + w_{3 \times 1, att} \quad (3)$$

where C_n^b is the rotation matrix between the navigation frame and body frame, Ω_i^b is the skew-symmetric matrix with terms defined by the IMU rate gyros (p - roll rate, q - pitch rate, and r - yaw rate) as shown in Eq. 4 [14], $w_{3 \times 1, att}$ is attitude integration process noise, and δt is the IMU sampling rate.

$$\Omega_i^b = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \quad (4)$$

Next, neglecting Coriolis terms, the velocity update transforms the IMU's specific force measurements from the body-frame to

the navigation frame, accounts for the acceleration due to gravity, and integrates over time, as shown in Eq. 5,

$$\begin{bmatrix} \hat{v}_N \\ \hat{v}_E \\ \hat{v}_D \end{bmatrix}_{k+1|k} = \begin{bmatrix} \hat{v}_N \\ \hat{v}_E \\ \hat{v}_D \end{bmatrix}_{k|k} + \left[\hat{C}_{b,k+1|k}^n \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \right] \delta t + w_{3 \times 1, vel} \quad (5)$$

where $w_{3 \times 1, vel}$ is the velocity integration process noise. The position estimates are updated by simply integrating the velocity estimates using trapezoidal integration,

$$\begin{bmatrix} \hat{r}_N \\ \hat{r}_E \\ \hat{r}_D \end{bmatrix}_{k+1|k} = \begin{bmatrix} \hat{r}_N \\ \hat{r}_E \\ \hat{r}_D \end{bmatrix}_{k|k} + \left[\begin{bmatrix} \hat{v}_N \\ \hat{v}_E \\ \hat{v}_D \end{bmatrix}_{k|k} + \begin{bmatrix} \hat{v}_N \\ \hat{v}_E \\ \hat{v}_D \end{bmatrix}_{k+1|k} \right] \frac{\delta t}{2} + w_{3 \times 1, pos} \quad (6)$$

where $w_{3 \times 1, pos}$ is the assumed position integration process noise. Finally, all of the IMU sensor biases, are updated assuming random-walk as in Equation 7.

$$b_{k+1|k} = b_{k|k} + w_{3 \times 1, bias} \quad (7)$$

The linearized error-state dynamics are given by Eq. 8,

$$F_{INS} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \hat{C}_{b,k+1|k}^n \\ 0_{3 \times 3} & \left[\wedge - \hat{C}_b^n \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \right] & 0_{3 \times 3} & \hat{C}_{b,k+1|k}^n & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \quad (8)$$

where \wedge indicates the skew-symmetric matrix of a vector. The error-state transition matrix, expanded to first order, is given in Eq. 9.

$$\Phi_{INS} = I_{15 \times 15} + F_{INS} \delta t \quad (9)$$

It should be noted that while INS was included in this formulation for completeness, currently, our UAV's IMU is experiencing too much vibration to be very useful. That is, in this implementation a process noise error-covariance for INS was initially selected assuming low IMU sensor noise, however, in practice, it was determined that the IMU mounted on the UAV was subjected to very high additional sensor noise due to propeller induced vibrations. Because of this, a tuning parameter was implemented in the EKF to be able to scale the nominal process noise assumed for INS. This value was ultimately tuned to a large number in order to yield satisfactory performance. As such, the IMU data in the error-state EKF results presented in the paper is giving very little weight (i.e., about 100 times less weight than would be typical of INS), which has implications on attitude estimation. Future work will investigate vibration dampening IMU mounts in order to improve the quality of INS.

3.2 UAV Laser Altimeter

Measurements from a laser altimeter mounted on the bottom of the UAV are shared with the UGV via a local Wi-Fi network. Assuming that the environment has a flat plane ground, altimeter measurements are assumed to be equal to the UAV's height above the ground whenever the UAV has zero pitch and roll attitude. Assuming relatively small pitch and roll angles, this measurement scales with the UAV's orientation as shown in Eq. 10 whose observation model is given in Eq. 11.

$$\hat{z}_{Alt} = \frac{-\hat{r}_D}{\cos(\theta)\cos(\phi)} + v_{alt} \quad (10)$$

$$H_{alt} = \begin{bmatrix} 0_{1 \times 8} & -\frac{1}{\cos(\theta)\cos(\phi)} & 0_{1 \times 6} \end{bmatrix} \quad (11)$$

The flat ground assumption will be relaxed once we integrate the system with the UGV's onboard map from SLAM. The measurement error-covariance of the altimeter measurement was assumed to be $v_{alt} \sim N(0, 10cm)$.

3.3 LIDAR UAV Position Estimation

To use the 3D LIDAR data to constrain the relative position estimate between the UAV and UGV, the LIDAR produces a pointcloud (at a rate of approximately 5 Hz), $P = \{p_i \in R^3 \text{ for } i = 1, 2, \dots\}$, where each point, $p_i = \{x_i, y_i, z_i\}$, represents the relative location from the sensor reference frame to the surface that the point has been sampled. The collection of these points, P , provides a sparse representation of the environment surrounding the sensor. As such, finding the points within the pointcloud that are reflected off of the UAV gives direct position information of the UAV.

To enable the determination of UAV's location within the pointcloud, the processing algorithm adopted is depicted in Figure 4. First, the pointcloud is segmented into a hollow sphere whose inner and outer radii bound the UWB ranging by two

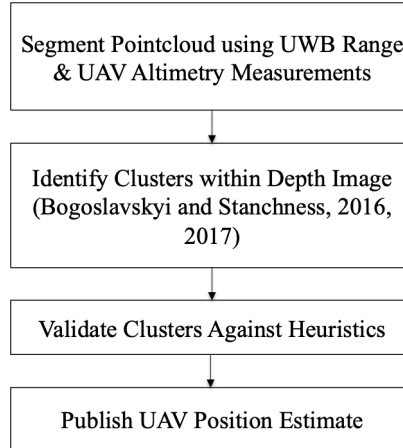


Figure 4: Overview of object detection and tracking with 3D LIDAR data software.

meters. This simply reduces the amount of computation needed to process the pointcloud and eliminates many potential false UAV detections. Next, the segmented pointcloud is passed to a fast depth-image clustering technique that was adopted from Bogoslavskyi and Stanchness [15, 16]. In each pointcloud, this clustering algorithm identifies several potential clusters. Then, the available candidate clusters are evaluated against a set of simple heuristics to determine if a particular cluster represents the UAV.

The specific heuristics used to identify a cluster as a UAV include:

- A minimum volume constraint with respect to the bounding box defined by the extremities of the cluster.
- Shape constraints on the bounding box that defined by the cluster. For example, defining the allowable difference between the length, width, and height of the bounding box.
- Ensuring that the height of the pointcloud is within a manually set threshold of the UAV's on-board laser altimeter.

The thresholds used within these heuristics were simply tuned empirically in order to reduce the number of clusters falsely reported as a UAV location based upon visual inspection. They would need to be updated based upon the size and shape of the UAV.

Once a specific cluster passes the set of heuristics, its centroid is reported as a UAV position estimate for fusing in the error-state EKF (Eq. 12) with the linearized observation sensitivity matrix as given in Eq. 13.

$$\hat{z}_{LIDAR} = \begin{bmatrix} \hat{r}_N \\ \hat{r}_E \\ \hat{r}_D \end{bmatrix} + v_{3 \times 1, LIDAR} \quad (12)$$

$$H_{LIDAR} = [0_{3 \times 6} \quad -I_{3 \times 3} \quad 0_{3 \times 6}] \quad (13)$$

The assumed measurement error-covariance for the LIDAR position updates is $v_{LIDAR} \sim \mathcal{N}(0_{3 \times 3}, I_{3 \times 3} 10cm)$.

3.4 Camera-Based UAV Line of Sight Vector

The method to track the UAV within the fish-eye camera assumes no significant motion of the UGV and employs the mixture of Gaussians 2 background subtraction method from the OpenCV library to distinguish the UAV from the surroundings [17] [18]. First, convolution of an averaging box filter (dimension: 77 x 77 pixels) blurs the image. Next, a binary filter, accepting

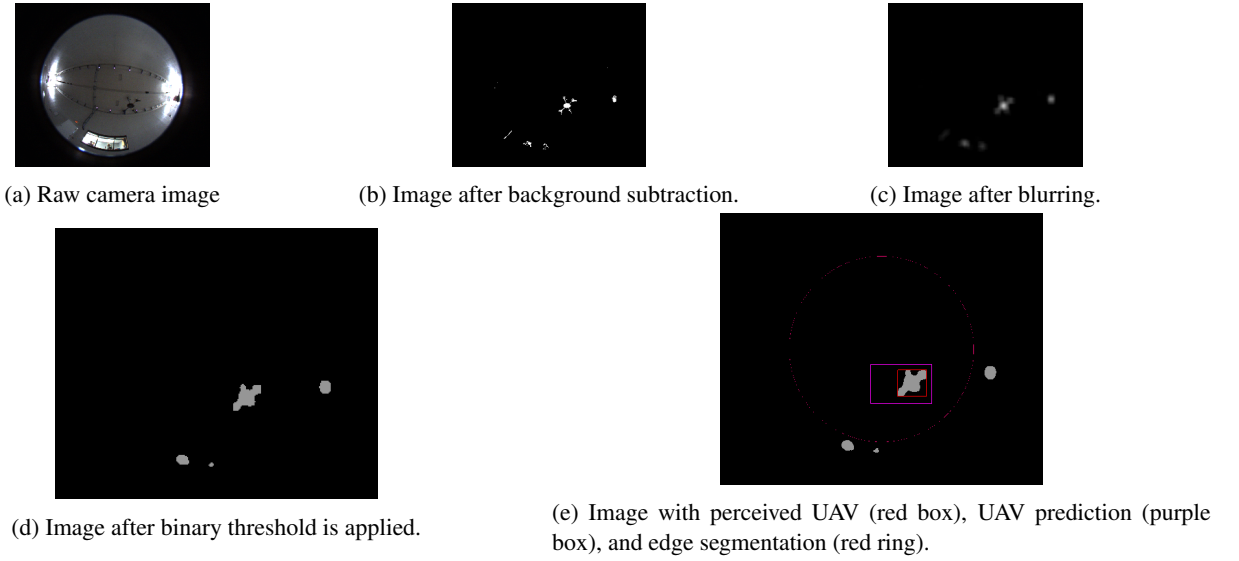


Figure 5: Sample of the UAV tracking process.

pixels within the range $[40, 150]$ of $[0, 255]$, yields candidate blobs. Together, the filters remove small artifacts and merge larger bodies. Contour detection is then used to determine candidate bodies. In this approach, it is assumed that within the environment, moving objects in the camera field of view are limited to the UAV, humans, or other ground based entities. Since entities on the ground are always visible at the edge of the image, a ring boundary determines which contours belong to bystanders. Thus, the corresponding contours of bystanders can be safely removed from consideration with limited impact to the field of view. For the remaining contour, the centroid of the smallest bounding box is assumed to be the UAV location estimate. Once the UAV is identified, the most three recent frames are utilized to approximate the kinematics of the UAV over a short time span in order predict the motion of the UAV. With this information, the vision system can operate on a smaller field of view window, improving efficiency and rejection of noise sources. The process is shown in Figure 5.

Given the estimate of the UAV's location, this pixel coordinate is then processed with the Scaramuzza fish-eye camera calibration model [19] to determine a line-of-sight (LOS) unit vector for use in navigation. This line-of-sight unit vector is then fused as an observation in the error-state EKF, where the observation is modeled as listed in Eq. 14,

$$\hat{z}_{cam} = \begin{bmatrix} \frac{r_N}{\hat{\rho}_{UGV}^{UAV}} \\ \frac{r_E}{\hat{\rho}_{UGV}^{UAV}} \\ \frac{r_D}{\hat{\rho}_{UGV}^{UAV}} \end{bmatrix} + v_{3 \times 1, cam} \quad (14)$$

where $\hat{\rho}_{UGV}^{UAV} = \sqrt{r_N^2 + r_E^2 + r_D^2}$ is the estimated range of the UAV from the frame origin centered at the UGV.

It is important to note that, in this paper, the proper extrinsic calibration of the lever-arm between the camera and LIDAR, is being neglected, and a simple rotation of the camera axis is applied to put the LOS vector in a NED frame. While the lever arm between the camera and LIDAR is confined to a short distance only on the vertical axis, future work will properly calibrate the extrinsic between the various sensors.

The LOS vector's observation sensitivity matrix, H_{cam} , for is given by Eq. 15.

$$H_{cam} = \begin{bmatrix} 0_{1 \times 6} & \frac{-(r_E^2 + r_D^2)}{\hat{\rho}_{UGV}^{UAV^3}} & \frac{r_N r_E}{\hat{\rho}_{UGV}^{UAV^3}} & \frac{r_N r_D}{\hat{\rho}_{UGV}^{UAV^3}} & 0_{1 \times 6} \\ 0_{1 \times 6} & \frac{(r_N r_E)}{\hat{\rho}_{UGV}^{UAV^3}} & \frac{-(r_N^2 + r_D^2)}{\hat{\rho}_{UGV}^{UAV^3}} & \frac{r_E r_D}{\hat{\rho}_{UGV}^{UAV^3}} & 0_{1 \times 6} \\ 0_{1 \times 6} & \frac{(r_D r_N)}{\hat{\rho}_{UGV}^{UAV^3}} & \frac{r_D r_E}{\hat{\rho}_{UGV}^{UAV^3}} & \frac{-(r_E^2 + r_N^2)}{\hat{\rho}_{UGV}^{UAV^3}} & 0_{1 \times 6} \end{bmatrix} \quad (15)$$

The measurement error-covariance for the LOS unit vector reported by the camera tracking algorithm was $v_{cam} \sim \mathcal{N}(0_{3 \times 3}, I_{3 \times 3} 0.1)$ and is scaled by $\hat{\rho}_{UGV}^{UAV}/5$ based upon emperical tuning.

3.5 UWB Ranging Radio

The pair of ranging radios on the UAV and UGV provide a consistent 10 Hz ranging measurement. As discussed in the LIDAR processing section, this range is used to assist in the LIDAR pointcloud segmentation process. In addition, it is fused as an observation within the error-state EKF. The range measurement is modeled as in Eq. 16,

$$\hat{z}_{UWB} = \hat{\rho}_{UGV}^{UAV} + v_{uwb} \quad (16)$$

and the observation sensitivity matrix is given as in Eq. 17.

$$H_{UWB} = \begin{bmatrix} \mathbf{0}_{1 \times 6} & -\frac{1}{\hat{\rho}_{UGV}^{UAV}} & -\frac{1}{\hat{\rho}_{UGV}^{UAV}} & -\frac{1}{\hat{\rho}_{UGV}^{UAV}} & \mathbf{0}_{1 \times 6} \end{bmatrix} \quad (17)$$

The measurement error-covariance of the UWB measurement was assumed to be $v_{uwb} \sim \mathcal{N}(0, 10cm)$.

3.6 UAV Velocity and Attitude Constraints During Hover

By monitoring the UWB range measurements over time, it is possible to detect when the UAV is likely in a hovering condition. In this paper, this is done by simply determining if two sequential 10 Hz UWB measurements are within 10 cm of each other.

Whenever the UAV is determined to be hovering, it is possible to constrain the level INS drift by processing pseudo-measurement constraints on the velocity and attitude. While hovering is not strictly a zero-velocity or zero pitch and roll condition, it is assumed that the UAV has low velocity and pitch and roll attitude near-zero, such that a pseudo-measurement of zero coupled with measurement noise helps constrain drift.

$$\hat{z}_{hover} = \begin{bmatrix} \hat{\phi}_b^n \\ \hat{\theta}_b^n \\ \hat{v}_N \\ \hat{v}_E \\ \hat{v}_D \end{bmatrix} + v_{5 \times 1, hover} \quad (18)$$

The observation sensitivity for the pseudo-measurements is given as

$$H_{hover} = \begin{bmatrix} -1 & 0 & 0 & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 9} \\ 0 & -1 & 0 & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 9} \\ \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 1} & \mathbf{0}_{3 \times 1} & -\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 9} \end{bmatrix} \quad (19)$$

where the measurement error-covariance for the velocity states is set to $v_{3 \times 1, hover} \sim \mathcal{N}(0, \sigma_{zero-vel} = 0.1 + \delta_{UWB} \mathbf{I}_{3 \times 3})$, and for the pitch and roll pseudo-measurements is set to $v_{2 \times 1, hover} \sim \mathcal{N}(0, \sigma_{zero-vel} = 0.05 + \delta_{UWB} / 10 \mathbf{I}_{2 \times 2})$ Where, δ_{UWB} , is the difference between two sequential UWB range measurements.

4 EXPERIMENTAL RESULTS

In order to experimentally evaluate the presented UAV-UGV GNSS-denied navigation system, a set of experimental remotely piloted UAV flight tests were conducted in the WVU free-flight wind-tunnel facility, which is a large section open-return wind tunnel approximately 5-m x 5-m square with faceted corners and a length of 36 meters. A total of 40 VICON Vantage V5 cameras are mounted near the ceiling of the test section, equally spaced at approximately 1.9 meters along the full length of the test section. This spacing and the orientation of the cameras ensure marker reflection capture by at least 3 cameras in all regions of the test section. Figure 6, shows the interior of the test section in which flight tests were conducted.

To determine independent position reference solutions, multiple VICON reflective markers were affixed around the UAV's IMU and a marker was affixed to the top of the UGV's LIDAR. This allowed the approximate location of IMU of the UAV be the center of navigation for the UAV within the VICON solution, and for the relative position of the UAV to be determined with respect to the marker on top of the UGV's LIDAR. In addition, a rotation of 6.5 degrees in heading was performed to align the heading of the UGV to the wind-tunnel walls. This was required so that the VICON and LIDAR frames were aligned. Unfortunately, this procedure is prone to error and was performed by rotating the pointcloud so the walls of the wind-tunnel appeared to aligned with the x-axis of the coordinate frame based on visual inspection. In future testing, the origin of the VICON frame will be aligned with the UGV. In addition, for each flight test, the time alignment of the VICON reference solution and the UAV/UGV data was determined by comparing the height-axis of the VICON UAV position solution with the on-board UAV laser altimeter just after take off. Using these two signals, the peak of the initial UAV's ascent after take off could be easily discerned. Once the time stamp between the data recorded in ROS and VICON solutions were aligned, the 100 Hz VICON solutions were spline interpolated to the time stamps of the EKF solution for the purposes of error analysis.

For this analysis, all data were processed by playing back data recorded in ROS bagfiles. However, when playing back data, latency that would be experienced in real-time run must be modeled. In particular, ROS tags sensor messages with a



Figure 6: Test section of WVU Free-Flight Wind-Tunnel that is outfitted with a 40 camera VICON motion capture system.

timestamp at a specific epochs, but in real-time, there is additional latency before these data are available as published topics for use in other processes (e.g., LIDAR tracking, error-state EKF, etc.). Therefore, based on analyzing the difference between the ROS published time and stamped time in the LIDAR sensor messages of the recorded bagfile, the LIDAR position estimates were given a 0.22 second latency before being fused in the EKF, and based on an approximation, the camera updates were given a 0.1 second latency before being fused into the EKF. Future work will require extensive real-time testing to validate the presented performance under true latency.

Figure 7 shows an example of the UAV position estimated by the LIDAR clustering (blue) box alongside the corresponding error-state EKF estimate (red) box. Figure 8 shows an example position estimation performance of the error-state EKF against

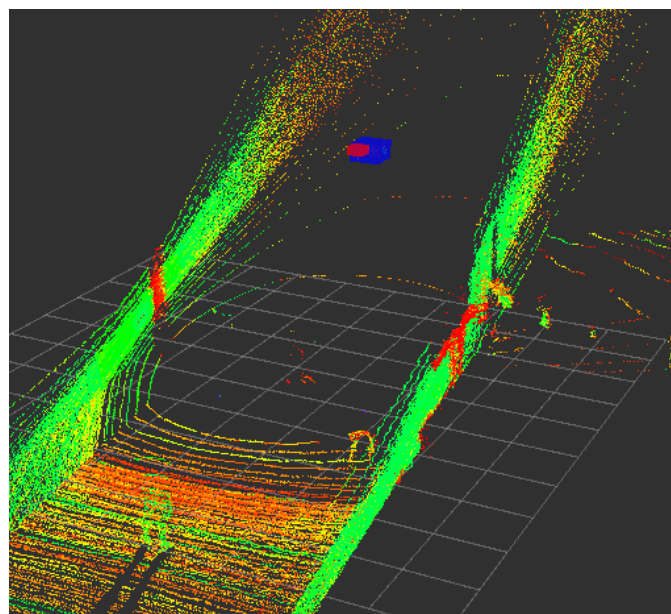


Figure 7: Example UAV position estimate at a single epoch (blue bounding box, LIDAR estimate, red bounding box EKF estimate) shown alongside the LIDAR pointcloud.

the VICON solution for the second of three flight tests.

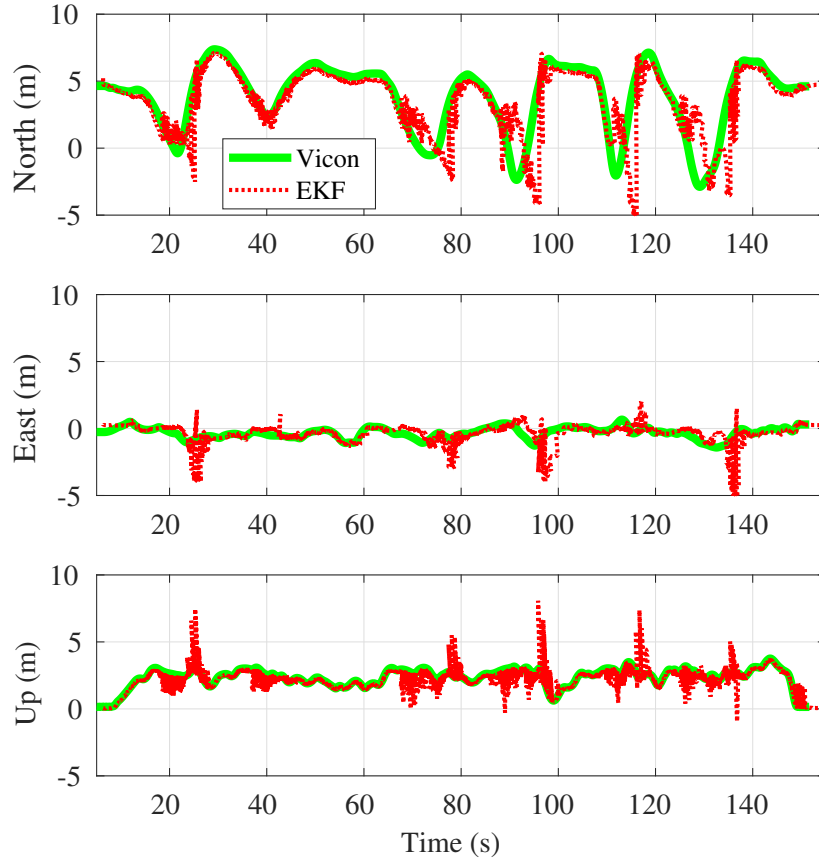


Figure 8: Example UAV position estimation performance against VICON reference solution for Flight 2.

Due to the shape of the testing facility being a long tunnel, the trajectory of the UAV traversed mostly back and forth along the local North axis. This was true of all three flights. Table 1 shows the EKF estimate positioning error statistics for three flight tests with respect to the VICON reference solutions, where the 3D positioning error of each data epoch was calculated as in Eq. 20.

$$error_{3D} = \sqrt{(\hat{r}_E - r_{E,Vicon})^2 + (\hat{r}_N - r_{N,Vicon})^2 + (\hat{r}_D - r_{D,Vicon})^2} \quad (20)$$

Table 1: EKF 3D positioning error statistics for the 3 flight tests.

	Duration (s)	RMS (m)	μ (m)	σ (m)	Median (m)	Max. (m)	# LIDAR Upd.	# Camera Upd.
Flight 1	158.65	2.60	1.59	2.05	0.91	15.83	250	144
Flight 2	139.86	2.16	1.37	1.67	0.65	10.75	251	201
Flight 3	159.9	2.35	1.20	2.02	0.61	17.77	122	182

As listed in Table 1, each flight was between 2 and 3 minutes in duration. In each of the three flight tests, the median 3D positioning error is below 1 meter and the RMS error is between 2 and 3 meters. However, in each flight, there are several instances where the overall RMS is driven up by periods of 10 meter-level errors. This is likely where LIDAR or camera updates were not available. To further elaborate the value of the LIDAR updates, Table 2 shows the error statistics for only the LIDAR position estimates that were available in each flight, where VICON reference data was pre-processed in the same manner as described above.

Table 2: 3D Positioning error statistics of the LIDAR estimates for the 3 flight tests.

	RMS (m)	μ (m)	σ (m)	Median (m)	Max. (m)
Flight 1	0.58	0.57	0.10	0.58	0.75
Flight 2	0.48	0.48	0.04	0.48	0.59
Flight 3	0.50	0.47	0.17	0.44	1.07

What is apparent from Table 2 is that there is a systematic bias between the LIDAR estimates and the VICON solution. That is, RMS error is dominated by a mean ~ 50 cm and the standard deviation of the error is relatively small ~ 10 cm. It is speculated that these biases could be due to one of the following reasons. First, the LIDAR is tilted 15 degrees. While caution was used to mount this at 15 degrees, by using a angled shim within mounting bracket, a small tilt error could result in a bias of the LIDAR estimates. Second, there could still be a small frame rotation about the heading between the VICON reference frame and the UGV's initial LIDAR. Once a 3D map is used for navigation, this wont be an issue. Significant performance improvements stand to be made when this systematic error can be accounted for properly.

Given the vulnerability of vision systems to and danger from false positives, it is necessary to ensure the system is robust to such occurrences. Whereas false negatives results in relative small error accumulation, false positives may dramatically alter the position estimate. To verify the robustness of the system, a human marked each image as having (or not) a correctly identified drone within the bounding ring. As demonstrated in Table 3, false positives account for 0.58 % of all tracking output. The primary mode of failure for the vision tracking is attributable to false negatives. Future work, will see improvement to this metric, however, the rarity of such occurrences is well within what is required to successfully track the UAV.

Table 3: Performance of fisheye-camera tracking with respect to successfully identifying drone.

	True Positive	True Negative	False Positive	False Negative
Flight 1 (frames)	364	712	8	118
Flight 2 (frames)	411	605	12	82
Flight 3 (frames)	181	1024	1	73
Total (frames)	956	2341	21	273
Total (%)	26.62	65.19	0.58	7.60

Given that the IMU data is largely de-weighted in the algorithm formulation, large errors occur in the position estimates whenever LIDAR or camera updates are not available. To further illustrate this, Figure 9 shows the 3D positioning error for the same flight shown in Figure 8, when plotted alongside the cumulative number of updates from LIDAR and the fish-eye camera fused into the EKF. In Figure 9, when the cumulative update curve is flat, it indicates that no update is available from that sensor. For example, in Figure 9 from approximately, 65 to 80 seconds into the flight, no LIDAR updates are available, and the positioning performance is degraded. This is similar to the periods of 110-120s and 125-140s where there are relatively few camera or LIDAR updates. This is contrary to the periods, which as 45-65s which exhibits relatively low positioning error due to the steady availability LIDAR updates. This clearly motivates the need for coupling this navigation system with a planning algorithm that designs the UAV trajectory to balance exploration with keeping localization errors small.

Finally, as mentioned in the algorithm formulation discussion, while INS was included, due to large noise on the IMU (i.e., most likely from vibration), it is currently de-weighted to the level that provides little valuable information to the EKF. As such, the attitude estimation is very poor. For completeness, the 3D attitude estimation error, with respect to the VICON solutions UAV attitude is reported in Table 4.

Table 4: 3D attitude error statistics of the LIDAR estimates for the 3 flight tests.

	RMS (deg.)	μ (deg.)	σ (deg.)	Median (deg.)	Max. (deg.)
Flight 1	17.27	10.34	13.84	7.49	222.23
Flight 2	13.90	11.36	8.01	9.94	149.83
Flight 3	13.92	11.09	8.40	8.87	102.03

It is suspected that the poor attitude estimation error is an artifact of the IMU noise and mitigation strategies are currently being explored. The hover constraints are able keep the attitude estimates close zero when hovering, and most of the attitude error is accumulated during dynamic parts of each flight.

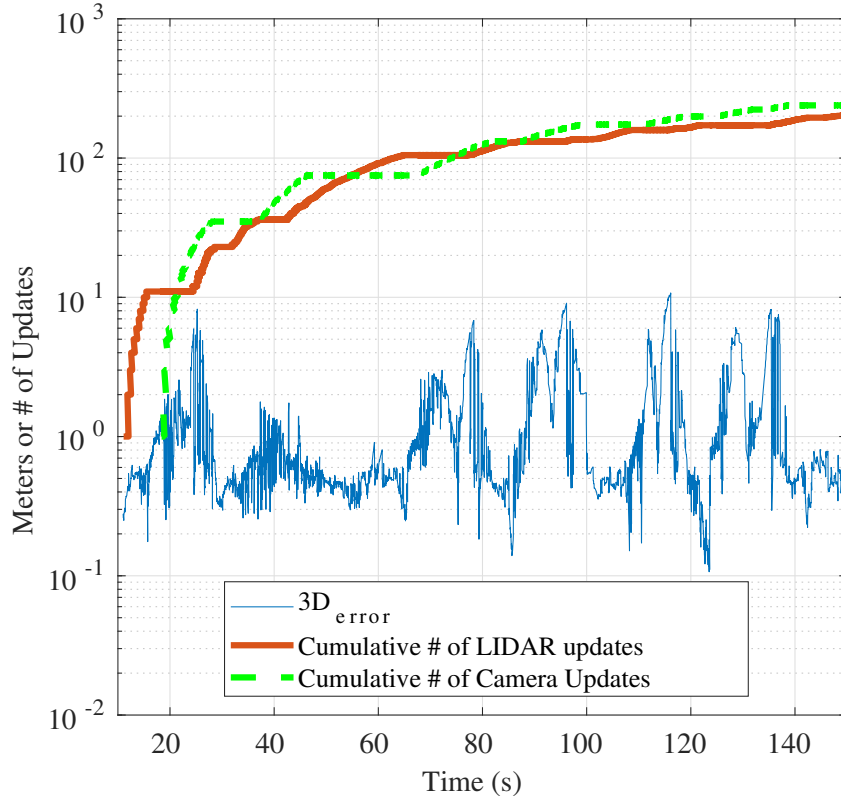


Figure 9: Flight 2’s time history of 3D position error alongside the cumulative number of measurement updates from the LIDAR and fish-eye camera.

5 CONCLUSIONS AND FUTURE WORK

We presented a GNSS-denied navigation system that relies on a cooperative UGV to track the relative state of a UAV. In a median sense, approximately better than meter-level 3D localization errors were experimentally demonstrated with respect to reference solutions provided by a VICON motion capture system. This capability will enable the UAV and UGV to conduct autonomous missions in unknown GNSS-denied environments, where the UGV produces a 3D map and the UAV acts as an ‘eye in the sky’ for search and rescue purposes. The performance of the presented approach is heavily dependent on the availability of the camera and LIDAR updates from the UGV.

To improve upon this baseline capability, our next immediate steps will be to couple this EKF with a planner that takes into account the uncertainties within the navigation system, following an approach similar to the Belief Roadmap algorithm [20]. We expect that this will enable the UAV to autonomously conduct its exploration mission while periodically coming into the field of view of the UGV’s LIDAR and camera to reduce localization error.

There are also several traditional and more incremental ways to improve upon the performance of the presented system. First, it is apparent that there is a systematic bias present between the LIDAR position estimates and VICON reference solutions. This could be due to an unaccounted frame rotation between the reference solution and the sensor system or a LIDAR mounting calibration error. To mitigate the potential that this is due to uncertainty in the mounting of the LIDAR tilt, a calibration procedure must be adopted. Second, to yield better consistency between the camera and LIDAR based updates, the extrinsic calibration between the camera and the LIDAR must be included in the EKF. Third, methods to reduce impact of propeller vibrations on the IMU via mounting hardware could stand to drastically improve the stand-alone INS performance of the UAV. As it stands, the IMU information is de-weighted so much in the EKF that it is not providing much information. For example, attitude estimation, while included in the formulation, is very poor (i.e., unusable). Currently, it is very likely that most of the information of the UAV dynamics is offered by pseudo-measurement constraints used when a hover condition is detected. By also incorporating a UAV dynamic model in the EKF, there is room for further improvements to INS as well as the potential to reduce the need for including hover pseudo-measurement constraints.

Finally, we plan to extend this algorithm to work whenever the UAV and UGV are both moving.

ACKNOWLEDGEMENTS

This research is funded in part by the National Geospatial-Intelligence Agency's Academic Research Program (Award No. HM0476-18-1-2000). "Autonomous Navigation of Small UAV/UGV Teams in Underground Tunnels". Approved for public release, 19-955. Mr. Matteo De Petrillo's is partially supported by a West Virginia University Statler PhD Fellowship.

REFERENCES

- G.-J. M. Kruijff, M. Janíček, S. Keshavdas, B. Larochelle, H. Zender, N. J. Smets, T. Mioch, M. A. Neerinx, J. V. Diggelen, F. Colas, *et al.*, "Experience in system design for human-robot teaming in urban search and rescue," in *Field and Service Robotics*, pp. 111–125, Springer, 2014.
- R. R. Murphy, "A decade of rescue robots," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 5448–5449, IEEE, 2012.
- A. Bharadwaj and M. de Haag, "Keynote: Navigating small-uas in tunnels for maintenance and surveillance operations," in *Proceedings of the ION 2017 Pacific PNT Meeting*, pp. 1065–1076, ION, 2017.
- J. Larson, B. Okorn, T. Pastore, D. Hooper, and J. Edwards, "Counter tunnel exploration, mapping, and localization with an unmanned ground vehicle," in *Unmanned Systems Technology XVI*, vol. 9084, p. 90840Q, International Society for Optics and Photonics, 2014.
- J. W. Wheeler, "A roadmap for robotics and remote systems in nuclear cleanup," tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2018.
- T. Chung, "DARPA SubT Challenge." <https://www.darpa.mil/program/darpa-subterranean-challenge>. Accessed: 2019-01-29.
- C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual-inertial odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2017.
- B. Nisar, P. Foehn, D. Falanga, and D. Scaramuzza, "Vimo: Simultaneous visual inertial model-based odometry and force estimation," *IEEE Robotics and Automation Letters*, 2019.
- C. Papachristos, F. Mascarich, and K. Alexis, "Thermal-inertial localization for autonomous navigation of aerial robots through obscurants," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 394–399, IEEE, 2018.
- S. Khattak, C. Papachristos, and K. Alexis, "Visual-thermal landmarks and inertial fusion for navigation in degraded visual environments," in *2019 IEEE Aerospace Conference*, pp. 1–9, IEEE, 2019.
- R. Watson, N. Ohi, S. Harper, C. Kilic, C. Yang, J. Hikes, M. De Petrillo, J. Strader, G. Hedrick, H. Nichols, *et al.*, "A rover and drone team for subterranean environments: System design overview," "Husky Unmanned Ground Vehicle." <https://www.clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/>. Accessed: 2019-01-29.
- M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, p. 5, Kobe, Japan, 2009.
- P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Artech House, 2013.
- I. Bogoslavskyi and C. Stachniss, "Fast range image-based segmentation of sparse 3d laser scans for online operation," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2016.
- I. Bogoslavskyi and C. Stachniss, "Efficient online segmentation for sparse 3d laser scans," *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, pp. 1–12, 2017.
- Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," vol. 2, pp. 28 – 31 Vol.2, 09 2004.
- Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recogn. Lett.*, vol. 27, pp. 773–780, May 2006.
- D. Scaramuzza, *Omnidirectional Vision: from Calibration to Robot Motion Estimation*. PhD thesis, ETH Zurich, 2008.
- S. Prentice and N. Roy, "The belief roadmap: Efficient planning in belief space by factoring the covariance," *The International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1448–1465, 2009.