Flight Test Design and Implementation for Airspace Independent Surveillance through a Distributed Ground Based Sensor Network

Vahram Stepanyan *
NASA Ames Research Center, KBR Wyle Services, Moffett Field, CA 94035

Corey Ippolito †
NASA Ames Research Center, Moffett Field, CA 94035

Keerthana Kannan‡
NASA Ames Research Center, KBR Wyle Services, Moffett Field, CA 94035

Evan T. Kawamura§
NASA Ames Research Center, Moffett Field, CA 94035

Thomas Lombaerts¶
NASA Ames Research Center, KBR Wyle Services, Moffett Field, CA 94035

Wendy L. Holforty‖
NASA Ames Research Center, Moffett Field, CA 94035

Chester V. Dolph**
NASA Langley Research Center, Hampton, Virginia 23681

The paper presents a system architecture for distributed sensing, networking and computing, its hardware implementation, and execution of initial flight experiments to validate theoretical findings. It induces development of distributed sensing requirements, framework, and architecture, development of distributed ground node hardware prototypes, integration of all nodes and testing of baseline functionalities, integration of in-house developed perception, migration and tracking software packages, establishing flight scenario and flyable path for a selected UAS, flying the air vehicle along the path, recording sensors measurements, preprocessing them and transferring the resulting data to an optimal computing center. It also addresses the challenges related to pre-flight hardware calibration, clock synchronization, sensor registration and establishing a communication network. Sensors data processing results demonstrate the functionality of the presented distributed architecture and satisfactory performance of the applied technologies.

I. Introduction

Distributed sensing and cooperative reasoning seek to address challenges for advanced airborne autonomy through utilization of observations from sources distributed across the airspace and are decisive for safe autonomous operations for passengers and cargo delivery in dense and highly dynamic metropolitan areas. These technologies are capable of supporting functions such as detect-and-avoid, monitoring and surveillance of the airspace along urban corridors and in terminal areas around/above vertiports, precision approach and landing, precision navigation, autonomous separation

*Technical Advisor, KBR Wyle Services, Intelligent Systems Division, Mail Stop 269-1, email: vahram.stepanyan@nasa.gov
†Aerospace Research Engineer, Intelligent Systems Division, Mail Stop 269-1, email: corey.a.ippolito@nasa.gov
‡Embedded Software Developer, KBR Wyle Services, Intelligent Systems Division, Mail Stop 269-1, email: keerthana.kannan@nasa.gov
§Aerospace Research Engineer, Intelligent Systems Division, email: evan.t.kawamura@nasa.gov
¶Aerospace Research Engineer, KBR Wyle Services, Intelligent Systems Division, Mail Stop 269-1, email: thomas.lombaerts@nasa.gov
‖Aerospace Research Engineer, Intelligent Systems Division, email: wendy.l.holforty@nasa.gov
**Aerospace Engineer, Aeronautics Systems Engineering Branch, email: chester.v.dolph@nasa.gov

and spacing, and autonomous fleet management. It can utilize both ground based and airborne observations trough network communications and dynamic computing to complete objectives and tasks relating to Advanced Air Mobility concept operations. Ref. [6] outlines the challenges and opportunities for distributed sensing and smart space concepts to meet the emerging needs of advanced urban operations in the national airspace.

Initial concepts to overcome these challenges are presented in Ref. [10] as a framework to enable a dynamic, topologically-adaptive, and distributed estimation system with mathematical formulation for abstraction of the problem, identification of requirements and constraints for operations, and algorithmic constructs to demonstrate the concepts of operation in precision navigation and independent surveillance supporting conformance monitoring of aircraft in airspace corridors. Solution examples for vision-based precision navigation in approach and landing by localizing known feature points in the scenery and by deep learning based object detection are presented respectively in Refs. [8, 9], and for independent surveillance of an airspace through a distributed ground sensing network are presented in Refs. [3, 5, 11, 12]. Ref. [12] compares several Extended Kalman Filter (EKF) based fusion trackers with radar and vision measurements, Ref. [5] focuses on the object tracking using an EKF based algorithm to fuse distributed ground-based sensors with specific properties and limitations, Ref. [3] presents an approach to a non-cooperating target tracking using an adaptive information fusion from a ground based distributed sensors with communication delays, and an air vehicles classification problem is addressed in Ref. [11] using velocity-based metrics.

As aircraft move through the environment, various sensors distributed across the system will have temporary opportunistic observations of the target. One challenge for such collective reasoning involves uncertainty in the topological structure of the estimation problem, associated with uncertainty and variability both temporally and spatially in the topology of data transmission network, and the sensor modalities being utilized. Therefore, to make a temporally and spatially changing information based in-time decision for safe and reliable airspace operations, it is necessary to automatically determine where to execute the reasoning algorithm to meet system's latency, cost and power consumption requirements given the communication network and computing hardware limitations. Ref. [1] presents an approach to data and reasoning optimal migration utilizing underlying edge computing infrastructure.

As a path toward flight test evaluations of theoretical derivations reported in Refs. [3, 5, 8, 11, 12] for a single non-cooperative airborne object detection and tracking using fusion of asynchronous variable-time measurements in a ground based distributed sensing, communication and computing infrastructure, Ref. [7] presents a simulation framework and software architecture for flight operations in a densely populated urban centers at a moderately-high scale (tens to hundreds of flights) that incorporates the NASA AAM concept vehicle dynamics integrated with a custom flight management system, a detailed simulated urban environment, and sensors infrastructure. This simulation framework enables evaluation of distributed sensing concepts and technologies using synthetic data, but the real-world flight data collected over environments relevant to the applications such as surveillance of the airspace or precision approach and landing could provide more benefits. The team conducted initial data collection flight tests for approach and landing of a VTOL aircraft at the NASA Armstrong Flight Research Center, which include high resolution camera images with overhead and orthographic views of the landing area, the ground truth for trajectory provided by the aircraft's INS-GPS blended navigation solution, and the camera pan, tilt, and roll angles for pose evaluation.

The focus of this paper is the real-word sensor data based tracking of a single non-cooperative airborne object, for which we presents a distributed sensing, networking and computing architecture, and its hardware implementation. It also presents initial flight experiments to validate theoretical findings reported in Refs. [3, 5, 8, 11, 12]. It induces development of distributed sensing requirements, framework and architecture, development of distributed ground node hardware prototypes, integration of all nodes and testing of their baseline functionalities, integration of in-house developed perception, migration and tracking software packages, establishing flight scenario and flyable path for a selected UAS, flying the air vehicle along the path, recording sensors measurements, pre-processing them and transferring the resulting data to an optimal computing center. It also addresses the challenges related to pre-flight hardware calibration, clock synchronization, sensor registration and establishing a communication network. The results of processing recorded in the course of several field experiments data demonstrate the functionality of the presented distributed architecture and satisfactory performance of the applied technologies.

II. Flight Test Scenario and Architecture

We consider the following flight test scenario at NASA Ames Research Center DART site. A set of heterogeneous sensors are operating at ground nodes 1-4 (yellow circles in Fig. 1) , which are stationary and have fixed directions. Each sensor takes measurements of an air vehicle flying in the area along a predefined flight path (magenta line in the figure) autonomously or by a remote pilot. We assume that the air vehicle is non-cooperative, but the flight telemetry data are transmitted to the monitoring observer as a ground truth for comparison, which is located at one of the computing centers associated with a ground node for this experimentation purposes. Sensors measurements accuracy may be a function of actual distance and corrupted with a sensor specific uncertainties. Each ground node is equipped with an independent power source, which also provides power to sensors, computers that host algorithms, and wireless communication antennas with associated modems. We assume that all hardware elements at the ground nodes are wired connected between themselves, but the nodes are communicating wirelessly with each other through included modems subject to communication constraints, which are assumed to be time-varying as a function of the information path and the current state of the system. The target aircraft takes off from the close proximity of Ground Node 1 and lands near Ground Node 3 and vice-versa. Fig. 2 displays schematics of the flight test architecture and hardware layout at each ground node.

Fig. 1 Target's flight path and observation stations distribution.

The main task to be accomplished in this flight experiment is estimation of the operating within the airspace aircraft's state based on the distributed sensors readings, which are transmitted over the deployed wireless network to a computing center that provides minimum end-to-end latency. To this end, several sub-tasks have to be completed before the actual flight addressing the challenges related to clock synchronization, sensor registration, communication calibration and computing timing. The clock synchronization is essential for tugging the sensor data at each stage. Sensors registration makes sure the accuracy of sensor placement and orientation setup, and measure the noise level and maximum useful range for sensing the target vehicle. Also, the tests should be carried out to for the communication functionality in Point-to-Point and Point-to-Multipoint settings, and for latency measurement in both settings. Timing tests for the image registration/segmentation and tracking algorithms execution latency are critical for the optimal computing location selection using the migration algorithm to minimize end-to-end latency. During the target's flight, one needs to make sure to record sensors valid measurements with a current time tag, to process them in local computing units and to transmit processed data to the selected optimal location for the tracking computation and display.

III. Hardware Assets

A. Air vehicle

For this flight experiment we use Freefly Alta X aircraft capable of autonomous way point tracking and telemetry data long range broadcasting [13], which is displayed in Fig. 3 with its specifications.

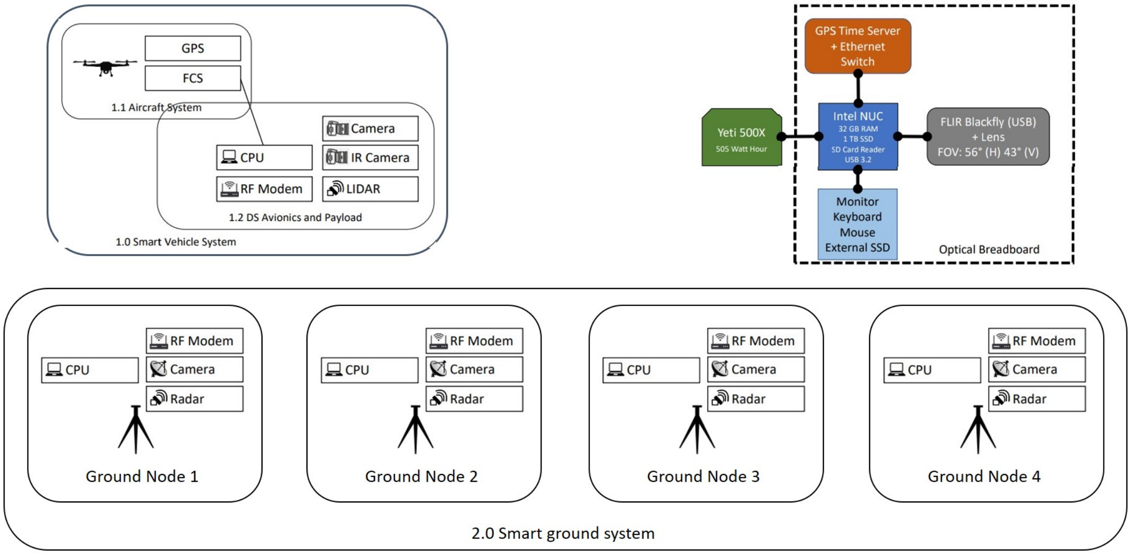

Fig. 2 Flight test architecture.

FREEFLY Alta X

- Number of Motors: 4
- Propeller Orientation: (2) CW and (2) CCW Props
- Propeller Type: Folding - 840 x 230 mm (33 x 9in)
- Maximum RPM 6300
- Peak Battery Voltage: 50.4V
- Nominal Battery Voltage: 44.4V
- Motor Max Continuous Power: 17,760W
- Motor Max Instantaneous Peak Power: 23,088W
- Maximum Payload: 15.9kg
- Typical Empty Weight: 10.4 kg
- Unfolded Diameter (Including Props): 2273 mm
- Autopilot Name: Custom PX4 flight control stack
- Maximum Pitch/Roll Angle 45°
- Maximum Yaw Rate 150° / second
- Transceivers: RFD900, 1-2 Miles LOS

=

Fig. 3 Target air vehicle.

B. Power Unit

Each observation station is equipped with a Yeti 500X portable power unit to supply the sensors, computers, modems and antennas with an electric power before and during the flight for data recording, processing, transmitting, track computing and display [14]. It is presented in Figure 4 with its specifications.

Charge Time (hrs)	USB-C: 8-11 hrs.; wall: 9 hrs.
External Charge	Car / USB / Wall
Battery Included	Yes
Battery Type	Lithium Ion
Battery Capacity (Wh)	505 watt hours
Power Output to Device	USB-A: 5V up to 2.4A (12W max) / USB-C PD: 1 at 18W max, 1 at 60W max / 6mm Port:12V, up to 10A (120W max) / 12V Car Port (output), 12V up to 10A (120 max) regulated / 120V AC Inverter: 120V AC 60Hz,
Material(s)	ABS plastic/aluminum
Dimensions	11.2 x 7.5 x 6 inches
Weight	12 lbs. 14.4 oz.

Yeti 500X portable

Fig. 4 Power unit.

C. Communication Hardware

For the wireless communication we use Microhard pMDDL900 Digital Data Link [15]. It include a motherboard with RS232/485 Serial and RJ45 Ethernet interfaces, a modem and dual antennas capable of wireless video and telemetry communications. It is displayed in Fig. 5 with its specifications.

Modem
- Microhard pMDDL900
- **2x2 MIMO OEM 900 MHz Digital Data Link**

Antenas
- MHS031011 - 2dBi 900MHz RUBBER DUCKY ANTENNA - RPSMA
- We are using these antennas (two antennas per unit):

Features
- Robust 2X2 MIMO 900 MHz Operation
- Maximal Ratio Combining (MRC), Maximal Likelihood (ML) decoding
- Low-Density Parity Check (LDPC)
- Up to 21 Mbps Iperf Throughput @ 8 MHz channel (-78 dBm)
- Up to 2 Mbps IPerf Throughput @ 4 MHz channel (-102 dBm)
- Extremely small foot print and very lightweight
- Serial Communication Port - RS232 TTL level (300bps to 921kbps)
- Dual 10/100 Ethernet Ports (LAN/WAN)
- Operating modes - Point-to-Point, Point-to-Multipoint, Relay
- Adjustable total transmit power (up to 1W)
- Input Voltage -Digital Voltage = 3.3V / RF Voltage = 4.2V
- Interface through local console, telnet, and web browser
- Weight - Approx. 230 grams
- Dimensions - Approx. 3.20” x 3.50” x 1.15” (81mm x 88mm x 29mm)

Fig. 5 Modem.

D. Computing Unit

For the sensor data processing and algorithms execution we use Intel® NUC Mini PC unit with Windows 11 [17], which is displayed in Fig. 6 with its specifications.

Product Name	Processor	Operating System	Included Memory	Included Storage	Graphics Output
Intel® NUC 11 Performance Mini PC - NUC11PAQi70QA	Intel® Core™ i7-1165G7 Processor (12M Cache, up to 4.70 GHz)	Windows 10, 64-bit	2x 8GB DDR4-3200	2TB PCIe X4 Gen4 NVMe M.2 SSD	HDMI 2.0a; USB-C (DP1.4); MiniDP 1.4

Fig. 6 Local computer.

E. Visual Sensor

The Blackfly® S leverages the industry’s most advanced sensors in an ice-cube form factor. It is packed with powerful features enabling you to easily produce the exact images you need and accelerate your application development. This includes both automatic and precise manual control over image capture and on-camera pre-processing. The Blackfly S is available in GigE, USB3, cased, and board-level versions.

Resolution	720x540
Max Frame Rate	522 FPS
Megapixels	0.4 MP
Sensor	Sony IMX287, CMOS, 1/2.9"
Pixel Size	6.9 µm
ADC	8-bit / 10-bit / 12-bit
Minimum Frame Rate	1 FPS
Gain Range	0.0 to 47.9943 dB
Exposure Range	4.0 µs to 30.0 S
Acquisition Mode	Continuous, Single Frame, Multi Frame
FOV	56° (H) 43° (V)
Image Buffer	240 MB
Flash Memory	6 MB non-volatile memory
Interface	USB 3.1
Power Requirements	8 - 24 V via GPIO or 5 V via USB 3.1
Power Consumption	3 W maximum
Dimensions/Mass	29 mm x 29 mm x 30 mm / 36 g

FLIR Blackfly USB3 Camera

Fig. 7 Visual sensor.

IV. Flight Test Description

Before the air vehicle starts flying, we setup four ground nodes with integrated hardware units in the test site. The cameras were mounted on the leveled identical tripod platforms, which allowed camera's elevation angle adjustment. The platforms' orientations were set to maximally cover the anticipated flight path of the aircraft, which in this case was an Alta X quadcopter drone. In order to accurately determine the cameras orientation with respect to ENU (East-North-UP) word frame, we placed a reference point in the each camera's field of view. We set the $2D$ GPS coordinates of ENU origin from the Reflection simulation data. Since the test area was fairly flat, we took the pavement level az a zero altitude for the ENU frame and measured the camera and reference points altitudes (in meters) from the pavement. $2D$ GPS coordinates of all cameras and reference points were measured using cell phone apps. All latitude-longitude-altitude coordinates are given in the Table 1. For the data time synchronization we used atomic

Table 1 Origin, cameras and reference points coordinates.

Location	Latitude	Longitude	Altitude
O	37.426861	-122.066015	0
C_1	37.428067	-122.067703	1.5484
P_1	37.428042	-122.067650	1.5758
C_2	37.427098	-122.061213	1.5362
P_2	37.427172	-122.061235	1.5758
C_3	37.424167	-122.060080	1.5667
P_3	37.424161	-122.060139	1.5758
C_4	37.426556	-122.067972	1.5179
P_4	37.426614	-122.067956	1.5484

clock apps on the cell phones, which were briefly place in front of the cameras as the quadcopter took off to record the readings. The communication between observers at the ground nodes was setup through a conference call using Microsoft Teams. Alta X was flown along the preset path, which is depicted in Fig. 1 by a magenta line at about 60 m altitude twice from Ground Node 3 to Ground Node 1 back and forth. The cameras started recording simultaneously as the aircraft took off and stopped recording as it landed.

V. Technologies Used for Data Analysis

A. Image Processing

Recorded by visual sensors, data was processed through an algorithm that utilizes image subtraction and blob detection, which avoids adding additional electromagnetic interference in the environment with sensors such as radar. This ground-based vision tracker (GBVT) outputs the detected objects' azimuth and elevation angles in the image plane, or equivalently its pixel coordinates, which are being sent to the observer for further processing through an Extended Kalman Filter (EKF) based estimation algorithm.

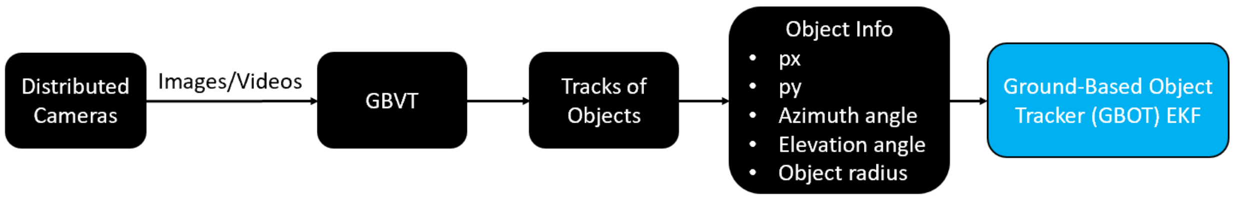

Fig. 8 High-Level Block Diagram

Fig. 8 shows a high-level block diagram of how the GBVT integrates with the EKF. The black boxes show the main steps for GBVT, and the blue box shows how GBVT feeds into the EKF.

This image processing approach crops the original high resolution image to a region of interest (ROI) and masks out irrelevant regions to the tracking at hand objects, after which subtracts the background, which shows only the moving objects. Figure 9 shows a screenshot of the GBVT during its image processing pipeline for a ground-based and fixed-angle camera, where the UAS flies horizontally from left to right across the camera's field of view. The top row shows the original image with masks and ROI, the cropped image via the ROI with the same masks, and subtracted images, respectively. The bottom row shows the current detection (red) and the previous detections (gray), respectively. This image processing pipeline follows the order of operations in Fig. 10.

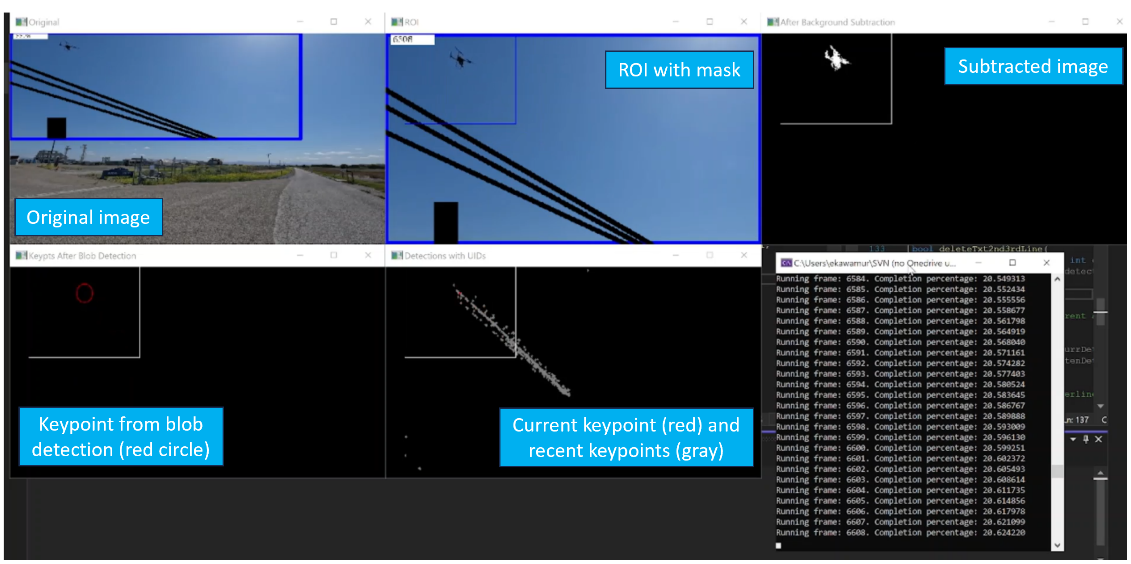

Fig. 9 Image Tracker Screenshot: Image Processing Pipeline for Ground Node 1 (6/26/23 flight test) - the black rectangle and lines remove the street light and power lines to help remove false positives

There are a couple of drawbacks of the image tracker. Sometimes there are labeling issues when tracked objects have the same path due to flying over the same point, i.e., intersection. For instance, Figure 11 shows an example of when the image tracker assigns the same label, 106, to two objects: one is a bird (left), and the other is the UAV (right). Both objects get the same label, 106, because both objects fly over or through a point on the path. The details of the

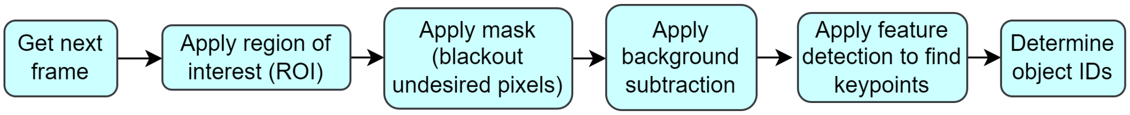

Fig. 10 Image Processing Pipeline

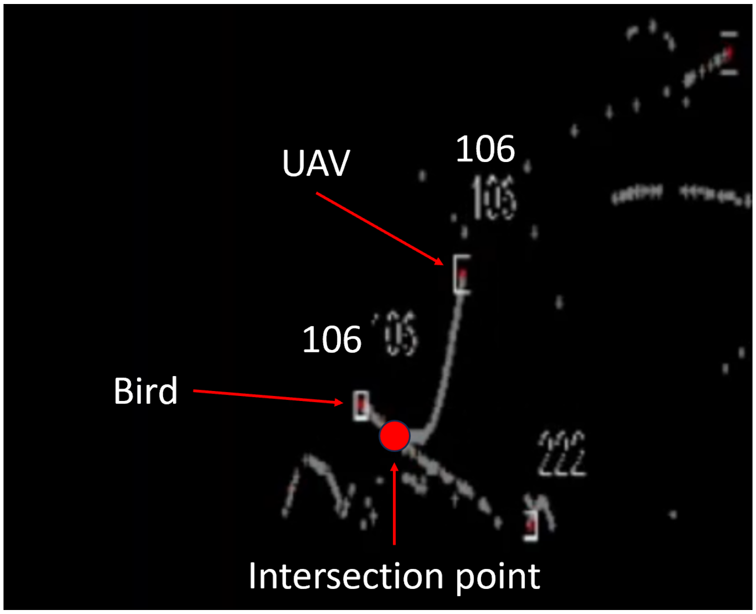

Fig. 11 Image Tracker: Same Label (106) for Two Objects, bird (left) UAV (right)

image processing algorithm and its implementation for this flight test data can be found in the accompanying paper [18] and in references therein..

B. Camera Registration

The GPS coordinates for each camera were translated to ENU frame centered at O using a Matlab function with Earth's WGS84 ellipsoid model. The camera EMU coordinates are presented in Table 2.

Table 2 Camera ENU coordinates with respect to origin O.

Reference point	East	North	Up
C_1	−149.4382	133.9040	1.5452
C_2	424.9986	26.3684	1.5220
C_3	525.3024	−298.9249	1.5380
C_4	−173.2510	−33.7949	1.5155

To be able to conveniently compute camera orientations with respect to ENU, we translate reference points coordinates into East-North-Up frame centered at the corresponding camera's optical center, which are presented in Table 3.

The reference points images were processed using GBVT pipeline. The resulting pixel coordinates are presented in Table 4. For these image points the azimuth and elevation angles (in degrees) are presented in Table 5.

Next we compute the reference points coordinates in the camera frame with an origin at the focal center and oriented

Table 3 Reference points ENU coordinates with respect to respective camera focal centers.

Reference point	East	North	Up
P_1	4.6910	−2.7746	0.0254
P_2	−1.9472	8.2129	0.0381
P_3	−5.2224	−0.6659	0.0127
P_4	1.4162	6.4372	0.0318

Table 4 Reference points pixel coordinates.

Reference point	P_1	P_2	P_3	P_4
u	2739	3522	1164	227
v	1859	2632	1663	2338

Table 5 Reference points image azimuth and elevation angles.

Reference point	P_1	P_2	P_3	P_4
ϕ	27.82	47.94	−30.84	−51.70
λ	12.77	28.44	5.707	20.35

right-down-forward as

$$\begin{aligned} x_c &= r\sin(\phi)\cos(\lambda) \\ y_c &= r\sin(\lambda) \\ z_c &= rcos(\phi)\cos(\lambda)\,, \end{aligned} \tag{1}$$

where $r = \|P\|$ for each point is computed from data in 3. Camera direction matrix is computed assuming a $3-2-1$ rotation from South-Down-East (SDE) to RDF with the camera roll angle (rotation about camera z-axis) set to zero, which results in

$$R_{SDE2RDF} = \begin{bmatrix} \cos(\theta_y) & 0 & -\sin(\theta_y) \\ \sin(\theta_x)\sin(\theta_y) & \cos(\theta_x) & \sin(\theta_x)\cos(\theta_y) \\ \cos(\theta_x)\sin(\theta_y) & -\sin(\theta_x) & \cos(\theta_x)\cos(\theta_y) \end{bmatrix},$$

where

$$\begin{aligned} \theta_x &= \tan^{-1}\left(\frac{y_c}{z_c}\right) - \sin^{-1}\left(\frac{P(2)}{\sqrt{y_c^2 + z_c^2}}\right) \\ \theta_y &= \tan^{-1}\left(\frac{P(1)}{P(3)}\right) - \sin^{-1}\left(\frac{x_c}{\sqrt{P(1)^2 + P(3)^2}}\right). \end{aligned} \tag{2}$$

These computations result in the following rotation matrices from each Camera frame to ENU frame:

$$R_1 = \begin{bmatrix} -0.0615 & 0.2528 & 0.9656 \\ -0.9981 & -0.0156 & -0.0595 \\ 0 & -0.9674 & 0.2533 \end{bmatrix}, R_2 = \begin{bmatrix} 0.5864 & -0.5130 & -0.6268 \\ 0.8100 & 0.3714 & 0.4538 \\ 0 & -0.7739 & 0.6333 \end{bmatrix},$$

$$R_3 = \begin{bmatrix} 0.3972 & -0.1087 & -0.9113 \\ 0.9177 & 0.0470 & 0.3944 \\ 0 & -0.9930 & 0.1184 \end{bmatrix}, R_4 = \begin{bmatrix} 0.5033 & 0.4490 & 0.7383 \\ -0.8641 & 0.2615 & 0.4300 \\ 0 & -0.8544 & 0.5196 \end{bmatrix}.$$

C. Communication Network Setup

The visual sensors at each ground node are wired connected to modems, which are wirelessly communicating each other using a mesh architecture. This architecture is adopted to seamlessly transmit data from sensors to computing centers, which addresses the efficiency of data transmission and ensures the safety and accuracy of the entire system. A mesh network embodies an amalgamation of RF modems operating cohesively as a unified network, offering diverse sources of connectivity within the specified area of interest, in contrast to a singular route. It is noteworthy that not all nodes necessitate direct contact with every other node. As illustrated in the figure, the flight remains within the range of only a subset of ground stations, a condition influenced by factors such as distance. The mesh architecture efficiently facilitates the relay of flight information to all nodes through node-to-node rerouting. Consequently, for expansive areas, strategically positioning a select few modems at specific locations can be sufficient to comprehensively cover the entire designated area. This underscores the efficacy and adaptability of the mesh architecture in optimizing network coverage.

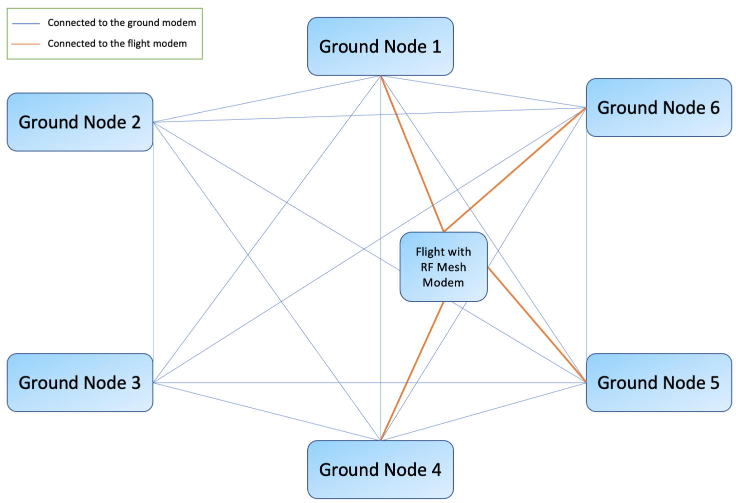

Fig. 12 Mesh Configuration of the modems

Fig. 12 represents a schematic illustration of RF Modems within a Mesh network, which can provide both communication between ground nodes (blue lines) and between an airborne sensor and the subset of ground nodes (orange lines). It has an ability to relay flight based information to all nodes by implementing node-to-node rerouting. The details of the network and modems configurations can be found in the accompanying paper [19] and in references therein.

D. Computation of Transmission Delays

In order to prepare for the next round of flight tests with real time airborne object tracking, we compute the data transmission delays assuming that the acquire camera data would be preprocessed at the observation stations using the available computing power of NUC. This would reduce the data size in each package, which include time tag, pixel coordinates and azimuth and elevation angles on the image centeroid, and package wrapping. This package size is estimated to be around 100 KB. For this flight test data transmission we assume a conservative size of 150 KB. To compute the associated transmission delays we apply optimal migration algorithm from [1]. For this scenario, the computations can be done at one of ground nodes using NUC. The assumption is that inside the ground node everything is wired connected, and the data from other nodes is received via the mesh network of Microhard modems, for which we set the maximum transmit power 1 W, network bandwidth to 8 MHz and maximum reliable coverage radius to 500 m . However, for the selected parameters and for this flight test specific ground nodes locations, the modems will not be able to reliably communicate with each other (see Ref. [19]). Therefore it is necessary to place a fifth ground node with power unit, NUC and modem close to two roads intersection (red dot in Fig. 1). Its position in the ENU frame is set to $[257.9763\ 66.0944\ 1.5220]^\top$.

Application of the optimal migration algorithm, which takes into account channel interference in the mesh network, results in the optimal execution location at Ground Node 1. The corresponding transmission delays in seconds are presented in Table 6.

Table 6 Transmission delays from visual sensors to the Ground Node1.

S_1	S_2	S_3	S_4
0	0.0138	0.0138	0.0012

E. Airborne Object's State Estimation

The object tracker is based on a conventional Extended Kalman Filter, which estimates object's $3D$ position and velocity components in ENU frame and assuming a constant velocity in the prediction stage and makes correction based on the available measurements. The filter gain calculation is modified in order to incorporate valid measurements from different sensors, which makes this framework adaptive and robust against invalid data and loss of signal between the different distributed sensors. This modular architecture is fairly straightforward to extend to any number of distributed multi-modal multi-rate sensors measurements. Fig. 13 illustrates the general overview of how the EKF operates.

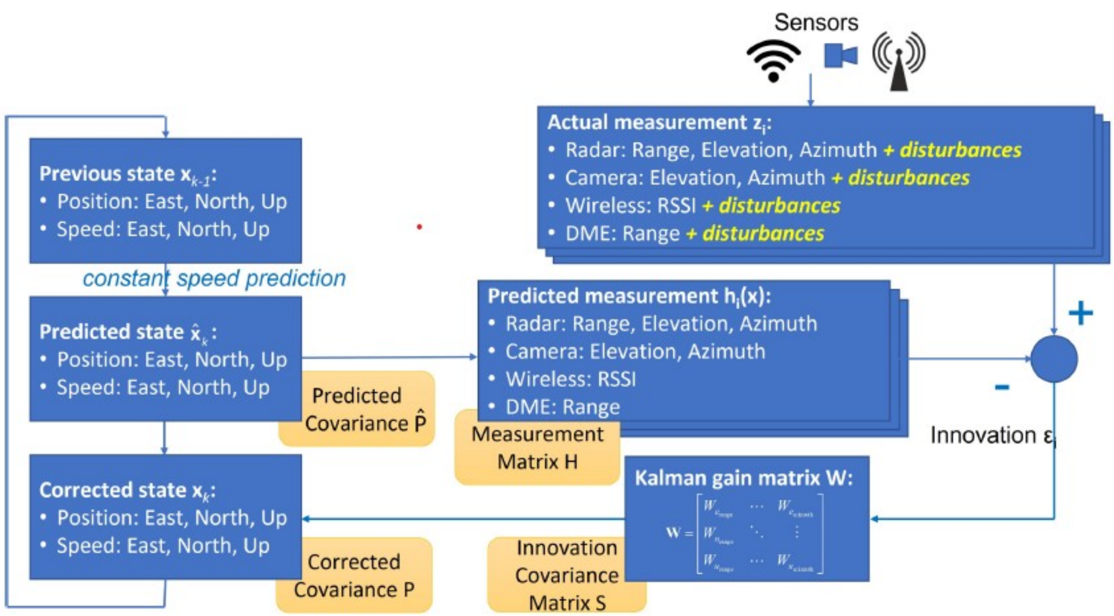

Fig. 13 General overview of the Distributed Sensor EKF setup

The details of the EKF setup and implementation can be found in the accompanying paper [20] and in references therein.

VI. Flight Test Data Analysis

The detailed flight test data analysis on can find in the accompanying paper [20]. Here we present final results of the quadcopter tracking in a single flight recorded by four visual sensors using the EKF based estimation algorithm. Fig. 14 displays quadcopter's corresponding pixel coordinates obtained by the ground-based vision tracker algorithm. It was computed that the observability matrix has a full rank between approximately 60 s and 470 s, where at least two cameras simultaneously cover the object's flight path, and its condition number is very large outside of this interval. Therefore, it is expected to see large tracking errors there.

Fig. 15 shows the errors between estimated coordinates and the ground truth from the quadcopter's flight computer log in ENU frame for the time interval with sufficient measurement data. An acceptable errors can be observed in all three directions. These errors are much larger in the areas with less than two overlapping sensor's. Therefore, the presence of more ground nodes with optimizes distribution and orientation would significantly increases the estimation accuracy.

Fig. 16 compares the estimated and true flight path in 3D in the East-North-Up frame. It shows that the estimate is mostly accurate in the East-North plane, but not as accurate vertically where the estimate drifts off slowly, which can be

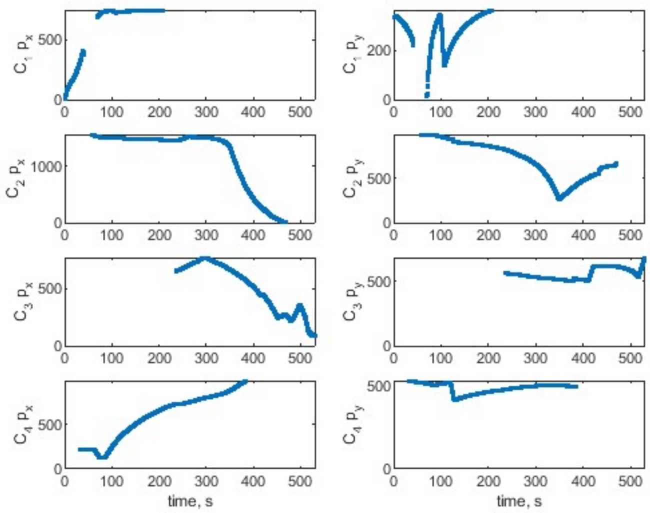

Fig. 14 Sensor readings after processing

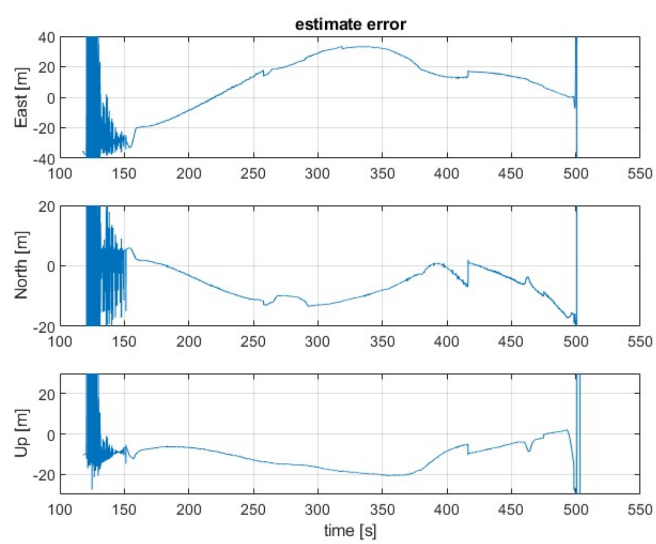

Fig. 15 Object's state estimation errors.

attributed to cameras lens distortion not specifically address in this flight test.

VII. Conclusions

We have presented a distributed sensing, networking and computing architecture and its hardware implementation for a single non-cooperative airborne object tracking using the real-word sensor data. To this end, we developed distributed ground node hardware prototypes, which include a power unit, a computing unit, a communication unit and one or more perception units. All ground nodes were integrated in a mesh network and their baseline functionalities were tested before the actual flight experiments, which were designed to validate in-house developed perception, migration and tracking technologies. We have established a flight scenario and a flyable path for the selected quadcopter UAS. We also have addressed the challenges related to pre-flight hardware calibration, clock synchronization, sensor registration and the communication network setup. In the course of quadcopter flights, the visual sensors recorded videos of them, which were post-processed through a ground-based vision tracker algorithm resulting in azimuth and elevation angles along with associated pixel coordinates of the quadcopter's image centroid. These results were fed to the EKF based estimation algorithm, and the quadcopter's motion was reconstructed, which demonstrates the functionality of the presented distributed architecture and satisfactory performance of the applied technologies. It was also concluded that distributing more ground nodes with optimizes topology and orientation in the area of interest and/or equipping ground nodes with diverse sensors would significantly increases the estimation accuracy.

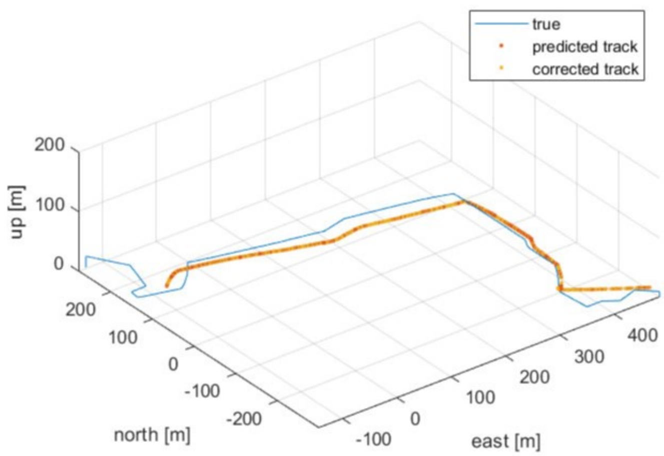

Fig. 16 Comparison between true path and estimate.

Acknowledgment

This work was performed under the NASA Aeronautics Research Mission Directorate (ARMD), Transformational Tools and Technologies (TTT) Project's Autonomous Systems activity. The authors would like to thank Zach Roberts, Jonas Jonsson, Matt Gregory, Edgar Torres and the other colleagues of Smart Mobility at NASA Ames Research Center for the successful drone flights, as well as NASA Armstrong colleague Nelson Brown, for their invaluable contributions to this project.

References

[1] V. Stepanyan, S. Schuet and K. S. Krishnakumar. An Approach to Reasoning Service Migration in Data and Reasoning Fabric (DRF) Implementation. *AIAA SciTech Forum, Jan 2022. https://doi.org/10.2514/6.2022-0787.*

[2] V. Stepanyan, T. Lombaerts, C. Dolph, N. B. Cramer and C. A. Ippolito. Estimation With Range Depended Sensor Model. *AIAA SciTech Forum, Jan 2022. https://doi.org/10.2514/6.2022-0494.*

[3] V. Stepanyan, K. Kannan, E. Kawamura, T., Lombaerts, and C. Ippolito. Target Tracking with Distributed Sensing and Optimal Data Migration. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-2194*.

[4] T. Lombaerts, K. H. Shish, G. Keller, V. Stepanyan, N. B. Cramer and C. A. Ippolito. Adaptive Multi-Sensor Fusion Based Object Tracking for Autonomous Urban Air Mobility Operations. *AIAA SciTech Forum, Jan 2022. https://doi.org/10.2514/6.2022-0362*.

[5] T. Lombaerts, K. Kannan, E. Kawamura, C. Dolph, V. Stepanyan, G. Gorospe and C. A. Ippolito. Distributed Ground Sensor Fusion Based Object Tracking for Autonomous Advanced Air Mobility Operations. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-0896*.

[6] C. A. Ippolito, K. E. Hashemi, E. Kawamura, G. E. Gorospe, W. Holforty, K. Kannan, V. Stepanyan, T. Lombaerts, N. Brown, A. M. Jaffe and C. Dolph. Distributed Sensing and Advanced Perception Technologies to Enable Advanced Air Mobility. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-0894*.

[7] K. Kannan, J. E. Baculi, T. Lombaerts, E. Kawamura, G. E. Gorospe, W. Holforty, C. A. Ippolito, V. Stepanyan, C. Dolph and N. Brown. A Simulation Architecture for Air Traffic Over Urban Environments Supporting Autonomy Research in Advanced Air Mobility. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-0895*.

[8] E. Kawamura, C. Dolph, K. Kannan, T. Lombaerts and C. A. Ippolito. Simulated Vision-based Approach and Landing System for Advanced Air Mobility. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-2195*.

[9] P. Veneruso, E. Miccio, R. Opromolla, G. Fasano, C. Tiana and G. Gentile. Analysis of ground infrastructure and sensing strategies for all-weather approach and landing in Urban Air Mobility. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-2225*.

[10] C. A. Ippolito, E. Kawamura, G. E. Gorospe, W. Holforty, K. Kannan, V. Stepanyan, T. Lombaerts, N. Brown, A. M. Jaffe and C. Dolph. A Structurally-Adaptive Framework for Distributed Airborne Sensing over Real-time Collaborative Information Sharing Networks. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-2680*.

[11] L. Dihel, C. Dolph, H. T. Holbrook and S. Roy. Classifying Aircraft using Velocity Data with Support Vector Machines and Likelihood Ratio Tests. *AIAA SciTech Forum, Jan 2023. https://doi.org/10.2514/6.2023-0898*.

[12] C. Dolph, T. Lombaerts, E.'Kawamura, C. A. Ippolito, V. Stepanyan, K. Iftekharuddin, G. Szatkowski, R. McSwain, C. Morris, M. R. Malekpour and C. Minwalla Ground to Air Testing of a Fused Optical-Radar Aircraft Detection and Tracking System. *AIAA SciTech Forum, Jan 2022. https://doi.org/10.2514/6.2022-0498*.

[13] https://freefly.gitbook.io/freefly-public/products/alta-x/untitled-3/technical-specs#radio-specs.

[14] https://www.rei.com/product/180610/goal-zero-yeti-500x-portable-power-station.

[15] https://www.microhardcorp.com/brochures/pMDDL900.Brochure.Rev.1.2.1.pdf.

[16] https://www.flir.com/products/blackfly-s-usb3/?vertical=machine+vision&segment=iis.

[17] https://www.intel.com/content/www/us/en/products/details/nuc/mini-pcs/products.html.

[18] E. Kawamura, K. Kannan, T.'Lombaerts, V. Stepanyan, C. Dolph, and C. A. Ippolito Ground-Based Vision Tracker for Advanced Air Mobility and Urban Air Mobility, *AIAA SciTech Forum, Jan 2024*

[19] K. Kannan, C. A. Ippolito and W. Holforty Flight Test Configuration of the Sensor Payload and the Ground Nodes in Distributed Sensing Frameworks, *AIAA SciTech Forum, Jan 2024*

[20] T. Lombaerts, E. Kawamura, K. Kannan, C. Dolph, V. Stepanyan, G. Gorospe and C. Ippolito Development and Field Test Results of Distributed Ground-Based Sensor Fusion Object Tracking, *AIAA SciTech Forum, Jan 2024*