UAV Sensor Payload Interface for Operator-in-the-Loop Target Geolocation and Live Map Mosaic Overlays

Ahmed Ashry[1], Christopher Titus[2], Mudit Singal[1], Joshua Schmucki[2], Zachary Bortoff[1], Joshua Gaus[2], and Derek A. Paley[3]

***Abstract*— Time-critical response missions require rapid geolocated target reports and scene context to support fast decisions. Unmanned aerial vehicles (UAVs) are well suited for wide-area scanning and target search, but aerial video can be difficult to interpret reliably under clutter, occlusions, and time pressure. To address this, we present USPI, a modular ROS 2 UAV Sensor Payload Interface that supports operator-in-the-loop use of live RGB/thermal streams. USPI enables click-to-center gimbal pointing and a pause-and-annotate workflow to localize one or more targets from image clicks, and it maintains a map-aligned mosaic overlay that updates scene context in real time. We evaluate USPI in controlled 15 m AGL field tests and in a timed mock mass-casualty scenario at 30 m AGL, showing sub-meter localization accuracy in the controlled tests and a median error of 1.3 m (RMSE 1.7 m) in the mock scenario.**

I. INTRODUCTION

Time-critical incidents place strong requirements on how quickly responders can form an accurate operational picture. In search and rescue (SAR), public safety response, and disaster management, responders need geolocated target reports and sufficient scene context to plan entry, allocate resources, and prioritize actions. Unmanned aerial vehicles (UAVs) can accelerate wide area search and provide overhead viewpoints that are difficult to obtain from the ground [1]–[3]. Mass-casualty incidents are a representative case where these requirements are pronounced, and the DARPA Triage Challenge (DTC) is one recent example that explicitly targets the generation of actionable, geolocated casualty information using heterogeneous robotic teams in mass-casualty scenarios [4].

A practical challenge in these settings is that purely automated casualty detection can be hard when operating from aerial viewpoints. Targets of interest often occupy a small number of pixels, may be partially occluded, and can appear under large scale and pose variation, which is known to degrade detector reliability in UAV imagery benchmarks and datasets [5], [6]. This is also consistent with SAR literature showing that person detection from aerial imagery remains challenging in realistic conditions [7]. In aerial SAR imagery search, human operators can achieve high precision but are slow, and their attention can degrade as the number of images increases, leading to missed detections. By contrast, automated detectors can process imagery in under a second and often achieve higher recall at the expense of increased false alarms. Human–AI teaming, where an automated detector proposes candidates and a human rapidly verifies them, can improve both recall and precision in large scale aerial imagery review (i.e., hundreds of images) [8]. This motivates an operator-in-the-loop workflow where an operator can rapidly confirm targets and trigger georeferenced reporting while the system performs the geometric computations consistently and in real time. Recent open-source ROS 2 frameworks for UAV-driven SAR similarly integrate human supervision with onboard sensing for mission planning and control [9].

This paper presents UAV Sensor Payload Interface (USPI),

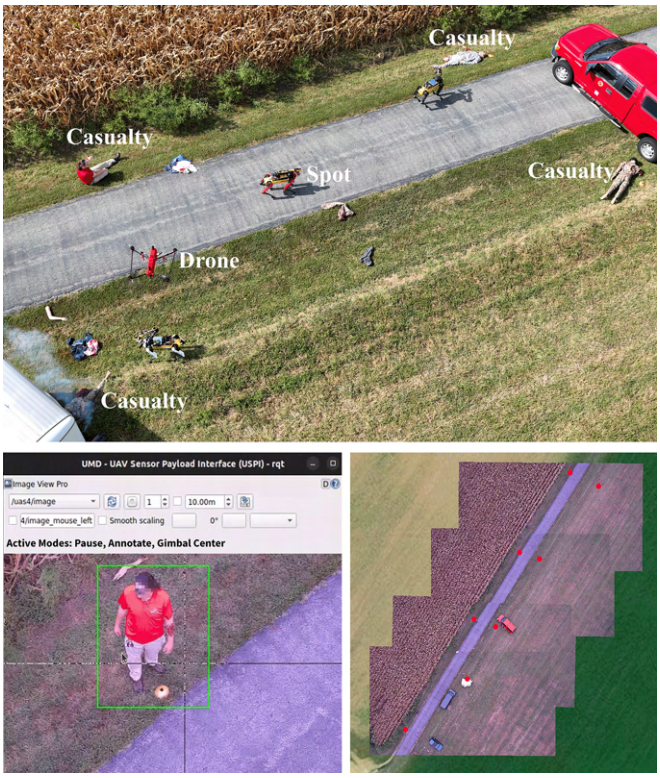

Fig. 1. USPI field operation overview: heterogeneous team in a simulated mass-casualty exercise (top), operator UI (bottom-left), and live mosaic overlay with geolocated markers (bottom-right).

[1]Ahmed Ashry, Mudit Singal, and Zachary Bortoff are with the Department of Aerospace Engineering, University of Maryland, College Park, MD, USA. aashry@umd.edu, msingal@umd.edu, zbortoff@umd.edu

[2]Christopher Titus, Joshua Schmucki, and Joshua Gaus are with the UAS Research and Operation Center, University of Maryland, College Park, MD, USA. ctitus@umd.edu, jschmu@umd.edu, jgaus@umd.edu

[3]Derek A. Paley is with the Department of Aerospace Engineering and the Institute for Systems Research, University of Maryland, College Park, MD, USA. dpaley@umd.edu

* This work was supported by Defense Advanced Research Projects Agency (DARPA) contract HR00112420304. Approved for public release; distribution is unlimited.

a practical UAV toolkit that focuses on rapid, geolocated target reporting and live situational-awareness products from gimbaled multi-sensor payloads. The workflow supports pausing an RGB/thermal video stream, centering the gimbal on a clicked pixel, annotating one or more targets using clicks or bounding boxes, and generating a structured report with synchronized platform and camera metadata and target geolocation estimates. The same information is used to project paused frames onto a ground plane to produce live mosaic overlays with target markers on a map. USPI is designed as a task-agnostic interface for rapid operator use to detect and localize targets of interest from onboard imagery. It is implemented as a modular ROS 2 pipeline and interfaces with PX4/MAVLink-based UAV workflows, which enables integration with a wide range of compatible platforms.

The contributions of this paper are at the system-integration and real-world deployment level, specifically: (1) a modular ROS 2 pipeline that converts image annotations into time-synchronized, georeferenced target reports, with click-to-center gimbal pointing and a pause-and-annotate workflow; (2) a real-time, map-aligned situational-awareness mosaic combining telemetry-based planar projection with multi-keyframe feature refinement; and (3) field evaluation with two sensors for quantitative localization accuracy characterization, plus a DTC challenge event case study demonstrating end-to-end operation in a time-constrained SAR workflow.

The remainder of the paper is organized as follows. Section II provides the preliminary background used. Section III describes the USPI system architecture and implementation, including the operator UI, gimbal centering, target localization, and mosaic overlay. Section IV reports experimental results from field tests and the DTC case study. Section V concludes and summarizes future work.

II. BACKGROUND

A. Camera Model and Calibration (RGB & Thermal)

We model each camera with a central pinhole projection. A 3D point in the camera frame $\mathbf{X}_c = [X_c, Y_c, Z_c]^\top$ with $Z_c > 0$ projects to the ideal (undistorted) normalized image coordinate

$$\mathbf{x}_n = \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix} = \begin{bmatrix} X_c/Z_c \\ Y_c/Z_c \\ 1 \end{bmatrix}. \tag{1}$$

The corresponding homogeneous pixel coordinate $\mathbf{p} = [u, v, 1]^\top$ is obtained using the intrinsic matrix

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{p} \sim \mathbf{K}\mathbf{x}_n. \tag{2}$$

where f_x, f_y are focal lengths in pixels and (c_x, c_y) is the principal point. This model is standard in camera calibration and multi-view geometry [10].

Pixel coordinate conventions are defined as follows: u increases to the right and v increases downward in the image. For back-projection, the undistorted pixel defines a ray direction (up to scale) in the camera frame as

$$\mathbf{r}_c \propto \mathbf{K}^{-1}\mathbf{p}. \tag{3}$$

which is the ray used for ray–plane intersection geolocalization and for mapping image corners onto the ground plane.

Real lenses introduce distortion that must be compensated before using the pinhole back projection. We use assumed standard radial and tangential distortion model parameterized by distortion coefficients $\mathbf{d}$. The camera calibration process estimates $\mathbf{K}$ and $\mathbf{d}$ using multiple views of a planar checkerboard target, followed by nonlinear refinement [10]. In practice, undistortion can be performed for sparse pixel sets, or for full images using precomputed undistortion maps and image remapping; both correspond to applying the estimated $\mathbf{K}$ and $\mathbf{d}$ to obtain undistorted coordinates consistent with the pinhole model [11].

The same geometric calibration objective applies to both RGB and thermal cameras. For thermal cameras, achieving reliable corner detection on a checkerboard requires sufficient thermal contrast between the pattern and background. Published approaches commonly rely on thermally contrasted calibration targets or procedures that increase checkerboard visibility in the thermal modality [12], [13].

B. Vision-Based Target Geolocation

Target geolocation maps an image observation (pixel coordinates) to a ground location using known camera intrinsics and platform pose. The standard approach back-projects an undistorted pixel into a camera ray, rotates the ray into a local inertial frame using the vehicle and gimbal attitude, and intersects it with a terrain model. A common approximation is a local horizontal plane (flat-earth assumption).

Localization accuracy is mainly limited by navigation and attitude errors, sensor misalignment, and terrain height uncertainty under a planar model, and degrades with higher altitude and more oblique angles; systematic installation errors can also dominate unless calibrated [14]–[16]. For typical low-flying UAVs (30–100 m AGL), ray–plane methods generally attain 1–5 m accuracy: about 3 m for fixed-wing MAV geolocation at 100–200 m altitude [17], and 0.7 m as a best result from 30 m altitude [18]. Accuracy can be improved with digital elevation models or bundle-adjusted pose, but the flat-earth approach is effective for rapid geotagging when terrain is approximately level. For the scope of operation in this work, USPI uses a local flat-plane model with rangefinder-assisted correction (Sec. III-C).

III. SYSTEM ARCHITECTURE AND IMPLEMENTATION

A. Operator Interface and Data Flow

USPI is launched from a single ROS 2 launch file that starts the video bridge, a camera calibration publisher, the operator interface UI, and backend processing nodes under a common UAV namespace for multi-vehicle use. Figure 2 summarizes the end-to-end data flow between the UAV and the ground control station. Onboard, the camera server

streams RGB/thermal video, and PX4 provides navigation, IMU, rangefinder, and gimbal telemetry over MAVLink. At the ground station, MAVLink Router and a ROS 2 bridge provide the operator interface with synchronized video and telemetry; the UI publishes target pixel coordinates to the gimbal centering node and publishes an annotation bundle to the target localization and mosaic nodes.

The operator interface UI is implemented as a modified image viewer derived from the ROS `rqt_image_view` package (a ROS image display plugin). It adds pause-and-annotate, click-to-center, and explicit mode feedback for the USPI workflow. Figure 3 shows the interface during annotation (RGB and thermal) and the corresponding live mosaic overlay with geolocated markers.

The interface subscribes to the live image stream and required MAVLink telemetry (GPS, heading, rangefinder range, gimbal attitude, local pose) and camera calibration. When the operator pauses the stream (space-bar), the UI retains the paused frame and a synchronized telemetry snapshot. The operator annotates one or more targets using bounding boxes, optionally revises with undo, and confirms the final selection.

Upon confirmation, the UI publishes a single custom ROS message that bundles the paused image, the telemetry snapshot at pause time, and the list of annotated targets. Each target includes the bounding-box dimensions and center pixel used for downstream geolocation. The message also carries operator-selected flags indicating whether to trigger a mosaic update and whether targets should be forwarded for downstream assessment. In click-to-center mode, the UI publishes a separate command message containing the clicked pixel, image dimensions, and camera field-of-view information derived from the calibration, then this command stream is converted into gimbal attitude commands. All custom ROS messages carry a UAV system identifier from MAVLink to support multi-vehicle operation.

B. Click-to-Center Gimbal Control

USPI provides a click-to-center capability that maps an operator-clicked pixel in the video stream to a gimbal attitude command such that the selected pixel is driven to the image center. The UI publishes the pixel coordinate (u, v) together with camera calibration information. A dedicated gimbal centering node converts this pixel offset into commanded yaw and pitch offsets and sends an attitude setpoint to the gimbal device manager through the MAVLink gimbal manager interface. The clicked pixel is first undistorted. Using the pinhole model, the undistorted pixel defines a camera-frame viewing ray (as in (3)). The corresponding angular offsets relative to the optical axis are obtained from the ray direction

$$\Delta\psi = \tan^{-1}\left(\frac{[\mathbf{r}_c]_x}{[\mathbf{r}_c]_z}\right), \qquad \Delta\theta = \tan^{-1}\left(\frac{[\mathbf{r}_c]_y}{[\mathbf{r}_c]_z}\right), \quad (4)$$

where $\Delta\psi$ and $\Delta\theta$ are the horizontal (yaw) and vertical (pitch) pointing offsets, respectively. The node computes the angular error relative to the image center by evaluating (4) for the clicked pixel and for the principal point (c_x, c_y), then applies their difference as an increment to the current commanded gimbal attitude. The sign convention is selected to match the gimbal pitch definition used by the MAVLink gimbal manager. The commanded angles are bounded to the physical limits of the gimbal, encoded as a quaternion, and published as a gimbal attitude setpoint. Our testing UAV configuration uses a two-axis gimbal in which yaw is mechanically coupled to the vehicle body, so only pitch is actuated in flight and the published setpoint holds gimbal yaw fixed; the same mapping extends directly to a three-axis gimbal where both corrections are applied.

C. Target Geolocation from Operator Annotations

For each confirmed annotation, the localization node receives the bounding-box center pixel (u, v) together with a time-synchronized telemetry snapshot (UAV position/attitude, gimbal attitude, and rangefinder reading). USPI reports (i) ray–plane intersection estimates based on the clicked pixel and the vehicle pose, and (ii) a rangefinder-based estimate when the selected target lies within the rangefinder beam footprint near the image center.

1) Ray–plane intersection localization: Following Section II-A, the annotated pixel (u, v) is first rectified and mapped to normalized image-plane coordinates (x_n, y_n) using undistortion and the calibrated intrinsics. In this notation, the back-projected ray direction can be written in the camera frame (up to scale) as

$$\left[\mathbf{r}_{P/O}\right]_C = \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix}, \quad (5)$$

where C is the camera optical frame, O is the camera origin, and P denotes a point on the ray.

Let I denote the local ENU inertial frame with origin O' fixed at the mission reference. The rotation from the camera frame to the inertial frame is formed through the camera mount, gimbal attitude, and UAV attitude:

$${}^{I}\mathbf{R}^{C} = {}^{I}\mathbf{R}^{B}\ {}^{B}\mathbf{R}^{G}\ {}^{G}\mathbf{R}^{C}, \quad (6)$$

where ${}^{A}\mathbf{R}^{B}$ rotates a vector expressed in frame B into frame A. B is the UAV body frame, and G is the gimbal frame. Here, ${}^{I}\mathbf{R}^{B}$ is obtained from vehicle attitude, ${}^{B}\mathbf{R}^{G}$ from the gimbal attitude, and ${}^{G}\mathbf{R}^{C}$ from the calibrated camera-to-gimbal mounting extrinsics. The ray direction expressed in frame I is then

$$\left[\mathbf{r}_{P/O}\right]_I = {}^{I}\mathbf{R}^{C}\left[\mathbf{r}_{P/O}\right]_C \quad (7)$$

The camera position in frame I coordinates at the paused snapshot is denoted by $\left[\mathbf{r}_{O/O'}\right]_I = [E_c\ N_c\ U_c]^\top$. A point along the ray is parameterized as

$$\left[\mathbf{r}_{P/O'}\right]_I(t) = \left[\mathbf{r}_{O/O'}\right]_I + t\ \left[\mathbf{r}_{P/O}\right]_I, \qquad t \geq 0. \quad (8)$$

Under the flat-earth assumption, the target location is obtained by intersecting (8) with a horizontal plane $U = U_p$. Writing the U-component explicitly gives

$$U(t) = U_c + t\ \left[\mathbf{r}_{P/O}\right]_{I,U}, \quad (9)$$

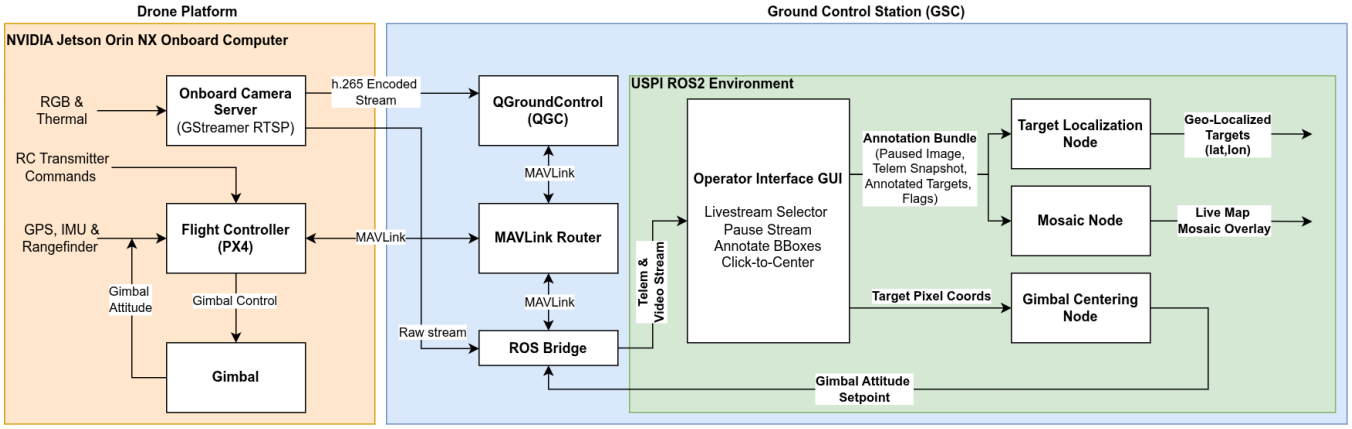

Fig. 2. USPI end-to-end architecture and data flow between the UAV and the ground control station, including the operator interface and backend nodes for gimbal centering, target localization, and mosaic generation.

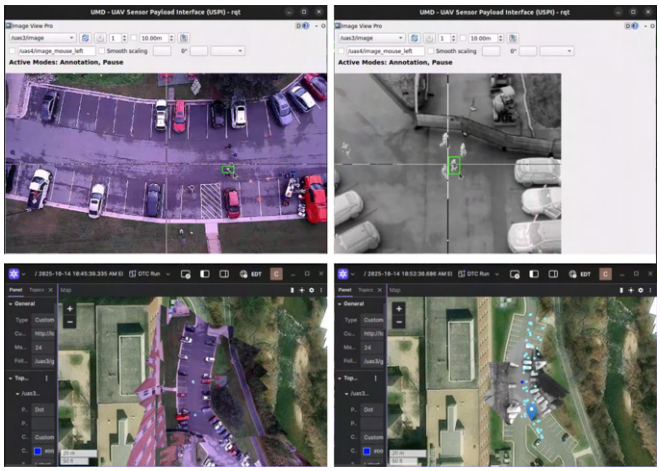

Fig. 3. USPI in operation: RGB/thermal pause-and-annotate UI (top) and Foxglove mosaic overlay with localized target markers (bottom).

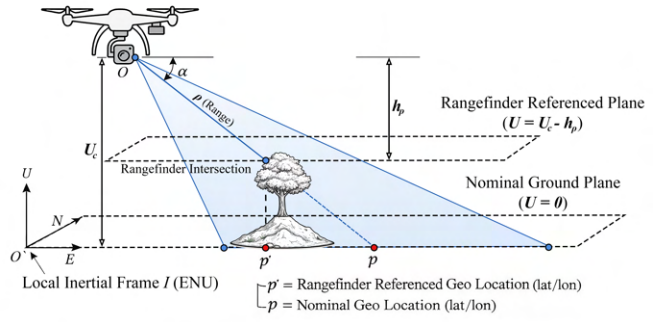

Fig. 4. Target localization geometry in world ENU frame I. The clicked-pixel ray intersects (i) the nominal plane $U = 0$ at p and (ii) a range-referenced plane $U = U_c - h_\rho$ at p', where $h_\rho = \rho \sin \alpha$.

where $\left[\mathbf{r}_{P/O}\right]_{I,U}$ is the vertical (U) component of the ENU ray direction, positive upward. Solving $U(t) = U_p$ yields

$$t_p = \frac{U_p - U_c}{\left[\mathbf{r}_{P/O}\right]_{I,U}}, \qquad \left[\mathbf{r}_{P/O'}\right]_I = \left[\mathbf{r}_{O/O'}\right]_I + t_p \left[\mathbf{r}_{P/O}\right]_I, \tag{10}$$

where $\left[\mathbf{r}_{P/O'}\right]_I$ is now the target position in world frame coordinates.

USPI computes two instances of (10) that use the same ray but different choices of U_p as shown in Figure 4.

Nominal ground-plane intersection: $U_p = 0$, which projects the ray onto the nominal ground reference in the local ENU frame.

Range-referenced plane intersection: when a valid range measurement ρ is available, the plane height is adjusted to match the elevation implied by the slant range and gimbal pointing. Let α denote the downward elevation of the range beam relative to horizontal (computed from the gimbal attitude). The vertical drop to the range hit is $h_\rho = \rho \sin(\alpha)$, and the intersection plane is set to $U_p = U_c - h_\rho$. This is useful when the rangefinder return lies on an elevated surface; using $U_p = 0$ would project the same viewing ray behind that surface.

Each ENU estimate is converted from local Cartesian ENU coordinates to geodetic coordinates (φ, λ, h) using the mission reference $(\varphi_0, \lambda_0, h_0)$ as the local origin and a standard ENU to geodetic transform.

2) Rangefinder-based localization: When the selected target lies within the rangefinder beam footprint (near the image center), USPI computes an additional estimate using the measured slant range and the vehicle heading. This estimate is typically more accurate when the beam is on the selected target. The measured range is projected into a horizontal distance $d = \rho \cos(\alpha)$, and the target latitude/longitude is obtained by solving the geodesic "direct" problem from the UAV geodetic position (φ_u, λ_u) with bearing ψ (UAV heading) and arc length $\delta = d/R_E$ on a spherical Earth of radius R_E:

$$\begin{aligned} \varphi_t &= \sin^{-1}\Big(\sin\varphi_u \cos\delta + \cos\varphi_u \sin\delta \cos\psi\Big), \\ \lambda_t &= \lambda_u + \tan^{-1}\left(\frac{\sin\psi \, \sin\delta \, \cos\varphi_u}{\cos\delta - \sin\varphi_u \sin\varphi_t}\right). \end{aligned} \tag{11}$$

This rangefinder-based estimate is reported alongside the ray–plane intersection outputs. Targets selected off-center are

localized using the ray–plane intersection method.

In the system implementation, localization is computed only when the required inputs are available (camera calibration, pose/attitude telemetry, and gimbal attitude). Ray–plane intersection is skipped when the ray does not intersect the selected plane in front of the camera (e.g., $t_p \leq 0$) or when the intersection becomes ill-conditioned (e.g., $\left[\mathbf{r}_{P/O}\right]_{I,U}$ is near zero).

D. Georeferenced Mosaic Overlay

USPI maintains a map-aligned mosaic overlay that builds scene context on top of a satellite basemap during flight. Whereas survey-grade frameworks such as OpenREALM [19] treat the orthomosaic along with an associated digital surface model as the primary product and rely on visual SLAM with a downward-looking gimbal and high frame overlap, USPI's mosaic is a secondary situational awareness product that is produced alongside target geolocation from the same operator-confirmed annotation bundle as discussed in Section III-C. Camera pose comes from GPS, IMU, and gimbal telemetry rather than visual SLAM, and the gimbal is allowed to pitch off-nadir. The trade-off is reduced geometric fidelity in oblique or sparse views, since the projection assumes a single horizontal ground plane rather than a reconstructed surface; we accept this in exchange for a mosaic that updates in step with the operator workflow and reuses the same geometric primitives already required for target localization without extra overhead.

Inputs and frame filtering: The mosaic node subscribes to the same annotation bundle as the target localization node, which contains an image frame along with telemetry snapshot. In our deployments, two source modes are used: (i) operator pause-and-annotate events from the UI, and (ii) an automated service publishing the same annotation message format at a fixed rate. To keep the workload bounded when frames are produced at video rate, the node applies two filters. The first employs a user-defined flag in the annotation message that marks frames intended for the mosaic. The second computes the fraction of new ground area added by an incoming frame relative to the existing coverage and skips frames whose new-coverage ratio falls below a threshold (15% in our runs), so that near-duplicate frames from a slowly moving UAV are not reprocessed.

Telemetry-driven planar projection: For each accepted frame, the node localizes the four image corners on the ground using the ray construction and plane intersection of Section III-C, with the same choice of nominal or range-referenced plane. Let $\mathbf{p}_i = [u_i\ v_i\ 1]^\top$ denote homogeneous image pixel coordinates (domain P) and $\mathbf{g}_i = [E_i\ N_i\ 1]^\top$ the corresponding homogeneous planar ground coordinates in local ENU (domain G). The image-pixel-to-ground homography $\mathbf{H}_{GP}$ satisfies

$$s_i\,\mathbf{g}_i = \mathbf{H}_{GP}\,\mathbf{p}_i, \qquad i \in \{1,2,3,4\}, \tag{12}$$

where s_i is a per-point homogeneous scale. $\mathbf{H}_{GP}$ is estimated from the four correspondences using a direct linear transform (DLT) fit by enforcing $\mathbf{g}_i \times (\mathbf{H}_{GP}\mathbf{p}_i) = \mathbf{0}$, which eliminates s_i and solves for $\mathbf{H}_{GP}$ up to an overall scale fixed by normalization.

To warp into the mosaic canvas, ground-plane points are mapped from ENU meters to canvas pixel indices. Let $\mathbf{c} = [u_c\ v_c\ 1]^\top$ denote homogeneous canvas coordinates (domain C). The canvas is defined by an ENU origin (E_0, N_0), resolution r (m/pixel), and a canvas height H_c (pixels), using

$$u_c = \frac{E - E_0}{r}, \qquad v_c = H_c - \frac{N - N_0}{r}, \tag{13}$$

where the negative sign maps increasing ENU N to decreasing canvas row index (v increases downward in image coordinates). In homogeneous form, (13) is an affine transform $\mathbf{c} \sim \mathbf{A}_{CG}\mathbf{g}$, and the resulting pixel-to-canvas warp applied to the paused image is

$$\mathbf{H}_{CP} = \mathbf{A}_{CG}\,\mathbf{H}_{GP}. \tag{14}$$

The frame is resampled onto the canvas grid using $\mathbf{H}_{CP}$ together with a binary footprint mask. The canvas itself can be pre-allocated to a fixed extent or grown automatically from an estimated ground sample distance on the first accepted frame, with a memory cap that coarsens resolution if exceeded so the node fits the available memory budget. Each warp is also clipped to a tight rectangle around the projected footprint to avoid unnecessary computation over the rest of the canvas.

Multi-keyframe registration refinement: Because the telemetry-driven projection is limited by pose, gimbal, and flat-ground assumptions, the node refines each accepted frame against a ring buffer of previously accepted keyframes (capacity 60 in our runs) before adding it to the mosaic. Candidate keyframes are pre-filtered by ENU footprint overlap with the incoming frame so that only spatially relevant ones are matched. A configurable detector and matcher pipeline extracts and matches descriptors. The default uses ORB [20] with brute-force Hamming matching and Lowe's ratio test [21]; SIFT with FLANN matching and SIFT with the learned LightGlue matcher [22] are available as alternatives. A tiered RANSAC fit [23] then accepts the simplest of similarity, affine, or full-homography models that meets an inlier threshold, and the resulting transform is composed with the keyframe's anchor mapping to produce a refined frame-to-canvas homography. Refinement is rejected and the telemetry-only mapping is retained when the implied average corner shift exceeds a user-specified bound (3 m in our deployments). This bound also serves as a sanity check against degraded GPS or gimbal estimates.

Live blending and shutdown re-blend: For the live overlay during flight, the warped frame is averaged into the running mosaic with weights that taper smoothly to zero at the edges of its footprint, which suppresses visible seams between adjacent frames. A running global mean gain reduces brightness drift between frames captured under different exposure conditions, with optional local exposure compensation in overlap regions. The live exports during flight are a cropped PNG and a JSON bounds file that records the geodetic lat/lon limits of the overlay; a local HTML

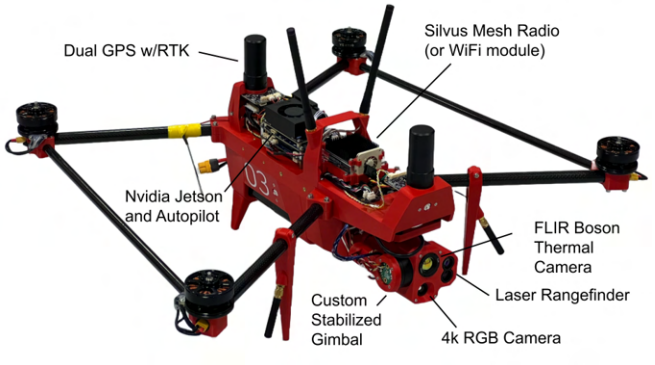

Fig. 5. Chimera UAV configuration.

map is produced via Folium, and a lightweight tile server serves the overlay as a custom map layer in tools such as Foxglove for live viewing (Figure 3). The node retains the warped frames during the run and, on shutdown, performs a batch re-blend over all retained frames using OpenCV's graph-cut seam finder, a global gain compensator that solves a linear system over all pairwise overlaps, and multi-band Laplacian-pyramid blending [24]. In our deployments this completes in around 0.5–1 seconds at the end of the mission and overwrites the live PNG and HTML outputs with the re-blended result. The mosaic shown in Section IV-C was produced this way. As elsewhere in the pipeline, the node requires a mission reference to define the local ENU origin, and updates are silently skipped when intrinsics, telemetry, or a valid plane intersection are unavailable.

IV. EXPERIMENTAL RESULTS AND FIELD EVALUATION

This section reports field test performance of USPI for quantitative target geolocation accuracy and integrated operation in a representative mass-casualty mission workflow.

A. UAV Platform

Field tests used the *Chimera* UAV, an experimental platform developed at the University of Maryland UAS Research and Operations Center (Figure 5). The 3D-printed airframe with carbon-fiber arms has a 2.6 kg flight-ready mass and approximately 45 min endurance on a 25.2 V, 10 Ah Li-ion battery. A custom BaseCam two-axis gimbal carries an IMX477 4k RGB camera, a Boson 640×512 thermal camera, and a laser rangefinder. Two ARK RTK GPS receivers provide position and heading. Flight control runs on PX4 with an onboard NVIDIA Jetson Orin NX (16 GB), and ground connectivity is via WiFi or a Silvus mesh radio.

B. Target Localization Analysis

Localization accuracy was evaluated through repeated outdoor field tests designed to exercise the full USPI pipeline under controlled geometry changes. Each test begins by hovering the UAV approximately 1 m above a ground target and recording its RTK position as a reference ground truth. The UAV then transitions to 15 m AGL and the operator triggers localizations while moving the target image location through a fixed pattern: image center, four corners (top-left, bottom-left, bottom-right, top-right), and back to center. This produces six ray–plane localizations per heading and two rangefinder localizations per heading (center-only). The same pattern is repeated for eight evenly spaced yaw headings. The mission is repeated for the RGB and thermal cameras at two gimbal pitch angles ($-60°$ and $-90°$).

For each localization event, we compute the planar ENU error components (e_E, e_N) relative to the RTK surveyed reference and report the radial error $e_r = \sqrt{e_E^2 + e_N^2}$. Figure 6 shows representative error distributions for the $-60°$ (oblique view) gimbal pitch configuration for each method/sensor pairing. We report circular error probable (CEP) as the percentile of the radial error distribution (CEP50 and CEP95 denote the 50th and 95th percentiles, respectively). Ray-plane localization achieves sub-meter radial performance in these runs (RGB: mean 0.63 m, CEP95 0.94 m; thermal: mean 0.39 m, CEP95 0.79 m). Rangefinder localization is evaluated only when the target pixel overlaps the image center, resulting in fewer samples, and achieves comparable performance when the overlap condition holds (RGB: mean 0.93 m, CEP95 1.19 m; thermal: mean 0.57 m, CEP95 0.83 m).

These localization error distributions also bound the geo-registration accuracy of (i) the target pins displayed on the operator map and (ii) the mosaic overlay footprint, since both products use the same ray construction and ENU projection described in Section III-C. The sub-meter values reported above are consistent with the ranges observed in prior low-altitude UAV ray–plane geolocation work (Section II-B); results near the lower end of the reported 1–5 m range are expected given the RTK-equipped platform and calibrate payload used here. In practice, mosaic visual quality is further influenced by pose/gimbal fidelity and the local flat-ground assumption; therefore, we focus quantitative evaluation on localization accuracy and present the mosaic as a qualitative product in the mission case study in Section IV-C.

C. DTC Challenge Case Study

USPI was also evaluated as part of an integrated mission workflow in the DTC Challenge Event 2. DTC emphasizes time-constrained casualty detection, geolocation, and injury assessment in complex environments, where location accuracy directly affects responder utility. The evaluated scenario was a simulated mass-casualty incident involving a C-130 aircraft crash with 30 casualties distributed over rough terrain and partial occlusions. The run lasted 20 min over an area of approximately 15,000 ft^2.

The deployed system used five robotic platforms: two Chimera UAVs and three quadruped ground robots. Each platform produced localized observations (location and associated imagery/metadata) that were transmitted to the ground station for downstream association and triage assessment. In the integrated pipeline, observations were associated to candidate casualties using proximity-based gating, and a

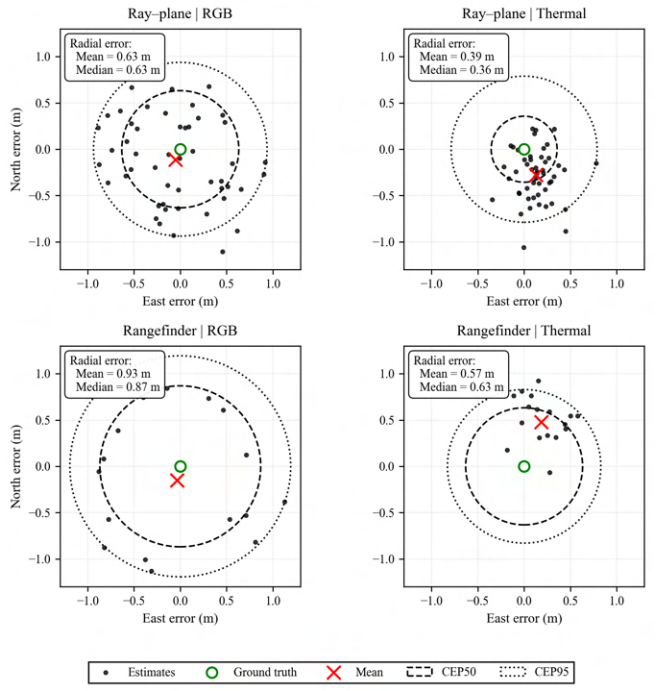

Fig. 6. Planar ENU localization error distributions for ray–plane intersection and rangefinder localization using RGB and thermal sensors at $-60°$ gimbal pitch. Each point is an (e_E, e_N) estimate relative to the ground truth. Dashed and dotted circles indicate CEP50 and CEP95 of the radial error.

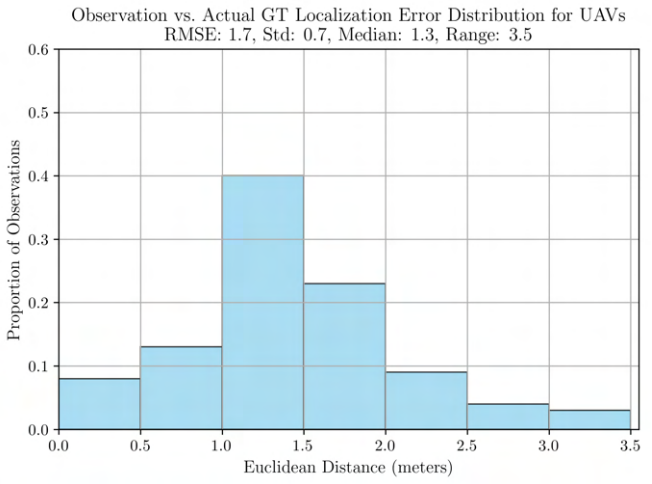

Fig. 7. UAV localization error distribution for observations associated with ground truth in the DTC plane-crash scenario.

Fig. 8. Mission mosaic overlay for the DTC plane crash scenario with geolocated casualty markers overlaid.

Kalman filter maintained each casualty location estimate using the position component of incoming observations.

Figure 7 summarizes UAVs localization error distribution for the plane crash scenario, aggregated over all localization outputs that could be associated to ground truth. In this run, UAV-derived localized observations achieved an RMSE of 1.7 m with a median of 1.3 m (std. 0.7 m) and a maximum error of 3.5 m. DTC scoring requires submitted locations to fall within a fixed acceptance radius (4 m) of the true casualty position. The observed errors fall within this acceptance region. Compared to the controlled 15 m localization tests, this scenario is more challenging with the UAV operating at approximately 30 m altitude, visibility and occlusions varied across the scene, and operator annotations could include occasional mis-clicks under clutter and partial visibility. These effects, along with pose/gimbal estimation errors, contribute to the range of the observed error distribution.

Figure 8 shows the mission mosaic overlay generated during the same scenario, with geolocated casualty markers overlaid. The mosaic provides a real-time situational-awareness product that aggregates image context over time, while its footprint georegistration is bounded by the same ray construction and ENU projection accuracy evaluated in Section IV-B. Small projection inconsistencies are expected when pose/gimbal estimates vary between frames and when the local flat ground assumption is violated or stressed by viewpoint changes.

Operationally, USPI was used to rapidly make detections, localize casualties, and maintain an updated map-level view of covered regions. End-to-end response from operator confirmation to published target reports completed within a fraction of a second, and the live mosaic refreshed within roughly one second of each accepted frame, which kept the operator view current with the search. In DTC Challenge Event 2, our team placed second overall and received the “No One Left Behind” award for the most casualties found, consistent with reliable location reporting under time constraints. Operator feedback highlighted two recurring benefits during runs: (i) click-to-center and confirmation reduced operator workload per localization, and (ii) the mosaic view supported coverage assurance by enabling quick review of previously observed

areas and identification of regions requiring revisits.

V. CONCLUSION

This paper presents USPI, a modular ROS 2 sensor payload interface for PX4/MAVLink UAV workflows that produces georeferenced target reports from manual image annotations. USPI supports live RGB/thermal streams, click-to-center gimbal pointing, pause-and-annotate target marking, and a map mosaic overlay that provides scene context for the operator. Results show that USPI can deliver useful localization under operational constraints. In controlled 15 m AGL field tests, localization achieved meter-to-sub-meter accuracy for both sensing modalities, whereas the 30 m AGL DTC plane-crash scenario achieved a median error of 1.3 m and an RMSE of 1.7 m for UAV-derived observations associated with ground truth. Accuracy is mainly limited by navigation and attitude quality, sensor payload alignment, and calibration, with sensitivity to attitude noise increasing under oblique gimbal pitch and at higher altitude; this motivates continued work on error correction and on a dedicated sensitivity analysis as part of future evaluation. Operationally, the pipeline ran continuously without failure and with low latency, which makes it a practical research tool for localization debugging and sensor configuration validation. Although motivated by mass-casualty response, USPI is task-agnostic and applicable to any domain requiring real-time geolocated target reporting, including fire response and campus safety.

In ongoing and future work, a near-term step is to integrate automatic detector proposals into the same workflow, with operator confirmation before publishing reports. We are also integrating multi-platform fusion so that observations from UAVs and other sensing platforms can maintain consistent multi-target tracks in a common map frame.

VI. DISCLAIMER

The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

REFERENCES

[1] C. O. Quero and J. Martinez-Carranza, "Unmanned aerial systems in search and rescue: A global perspective on current challenges and future applications," *International Journal of Disaster Risk Reduction*, vol. 118, p. 105199, 2025.

[2] H. M. Ray, R. Singer, and N. Ahmed, "A review of the operational use of UAS in public safety emergency incidents," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 922–931.

[3] C. Vincent-Lambert, A. Pretorius, and B. V. Tonder, "Use of unmanned aerial vehicles in wilderness search and rescue operations: A scoping review," *Wilderness & Environmental Medicine*, vol. 34, pp. 580–588, 2023.

[4] DARPA, "DARPA triage challenge," 2025, accessed: 2026-01-24. [Online]. Available: https://www.darpa.mil/research/challenges/darpa-triage-challenge

[5] D. Du *et al.*, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[6] Y. Cao *et al.*, "Visdrone-det2021: The vision meets drone object detection challenge results," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2021-October, pp. 2847–2854, 2021.

[7] X. Zhang, Y. Feng, N. Wang, G. Lu, and S. Mei, "Aerial person detection for search and rescue: Survey and benchmarks," *Journal of Remote Sensing*, vol. 5, 2025.

[8] S. Gotovac, D. Zelenika, Željko Marušić, and D. Božić-Štulić, "Visual-based person detection for search-and-rescue with UAS: Humans vs. machine learning algorithm," *Remote Sensing*, vol. 12, no. 20, p. 3295, 2020.

[9] B. Döschl, K. Sommer, and J. J. Kiam, "Auspex: An integrated open-source decision-making framework for UAVs in rescue missions," *Frontiers in Robotics and AI*, vol. 12, p. 1583479, 8 2025.

[10] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[11] OpenCV Contributors, "OpenCV: Camera calibration," accessed 2026-01-22. [Online]. Available: https://docs.opencv.org/4.x/d9/d0c/group__calib3d.html

[12] I. N. Swamidoss, A. Bin Amro, and S. Sayadi, "Systematic approach for thermal imaging camera calibration for machine vision applications," *Optik*, vol. 247, p. 168039, 2021.

[13] N. Sutherland, S. Marsh, F. Remondino, G. Perda, P. Bryan, and J. Mills, "Geometric calibration of thermal infrared cameras: A comparative analysis for photogrammetric data fusion," *Metrology*, vol. 5, no. 3, 2025.

[14] X. Zhang *et al.*, "Precise target geo-location of long-range oblique reconnaissance system for UAVs," *Sensors*, vol. 22, no. 5, 2022.

[15] A. Babinec and J. Apeltauer, "On accuracy of position estimation from aerial imagery captured by low-flying UAVs," *International Journal of Transportation Science and Technology*, vol. 5, pp. 152–166, 2016.

[16] H. Sun, H. Jia, L. Wang, F. Xu, and J. Liu, "Systematic error correction for geo-location of airborne optoelectronic platforms," *Applied Sciences*, vol. 11, no. 22, 2021.

[17] D. B. Barber, J. D. Redding, T. W. McLain, R. W. Beard, and C. N. Taylor, "Vision-based target geo-location using a fixed-wing miniature air vehicle," *Journal of Intelligent and Robotic Systems*, vol. 47, pp. 361–382, 2006.

[18] G. Paulin, S. Sambolek, and M. Ivasic-Kos, "Application of raycast method for person geolocalization and distance determination using UAV images in real-world land search and rescue scenarios," *Expert Systems with Applications*, vol. 237, p. 121495, 2024.

[19] A. Kern, M. Bobbe, Y. Khedar, and U. Bestmann, "Openrealm: Real-time mapping for unmanned aerial vehicles," *International Conference on Unmanned Aircraft Systems*, pp. 902–911, 9 2020.

[20] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *International Conference on Computer Vision*, 2011, pp. 2564–2571.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "LightGlue: Local feature matching at light speed," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 17 627–17 638.

[23] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in *Readings in Computer Vision*, 1987, pp. 726–740.

[24] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.