2026 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
***AUTONOMY AND ROBOTICS (A&R)* TECHNICAL SESSION**
AUGUST 11-13, 2026 - NOVI, MICHIGAN

DEVELOPMENT OF A UAV PLATFORM FOR REMOTE INSPECTION AND VISUALIZATION

Yeen K. Lee[1], Sean Bainard[1], Michael Shaughnessy[1], Matt Bolger[1], R. Tucker Koepp[1], Roya Salehzadeh[1], Stephen Mallory[1], James A. Mynderse[1], Pedro Guillen[2], Margarita Hernandez[2]

[1]Lawrence Technological University, Southfield, MI
[2]Centrepolis Accelerator, Southfield, MI

ABSTRACT

This paper details the development of an intelligence and inspection platform consisting of an attritable sub-250g UAV, a ground control station, and a visualization interface for users. The UAV architecture combines onboard obstacle detection and avoidance along with simultaneous localization and mapping to have full autonomous navigation inside of complicated GPS-denied environments. The ROS 2-to-Unreal Engine data pipeline allows for sensor fusion, data cleansing, and initial analysis as well as creation of a high-fidelity real-time 3D digital twin. The visualization interface allows users to easily identify critical features and turn data into intelligence to support decision making by soldiers and first responders.

Citation: Y.K. Lee, S. Bainard, M. Shaughnessy, M. Bolger, R.T. Koepp, R. Salehzadeh, S. Mallory, J.A. Mynderse, P. Guillen, M. Hernandez, "Development of a UAV Platform for Remote Inspection and Visualization," In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 11-13, 2026.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have evolved into a powerful tool for providing situational awareness and performing inspection tasks in hazardous or hard-to-reach locations. Soldiers and first responders can benefit from either a bird's-eye view or a first-person view of dynamic and dangerous environments (e.g., an active fire, a potential structural collapse, or a chemical leak) without endangering personnel. Combining UAVs with a mission-specific data dashboard can transform raw sensor readings into actionable intelligence.

Specific to emergency response, UAVs have demonstrated utility in emergency response over the last decade and have been used for flammable gas detection [1], autonomous mapping during fire incidents [2], and emergency response training [3]. Recent advancements in sensing, autonomy, and cost [4, 5], have enabled effective mapping and perception on smaller, lowercost platforms, thereby reducing barriers to adoption for municipalities.

Common ways of using drones in indoor information gathering are often limited by both the operator's cognitive load and the challenge of maneuvering in GPS-denied, indoor environments. For drones to be a valuable decision-support tool, they must function with a high level of autonomy, enabling first responders to concentrate on mission rather than flight control. This change towards autonomous reconnaissance demands the integration of both onboard sensing and high-level visualization.

Mutually Assured Development, LLC (MAD) is developing an intelligence and inspection platform consisting of an attritable sub-250g UAV, a ground control station (GCS), and a visualization interface (VI) for users. The ROS 2-to-Unreal Engine data pipeline allows for sensor fusion, data cleansing, and initial analysis at the GCS level as well as creation of a high-fidelity real-time 3D digital twin at the VI level. The VI allows users to easily identify critical features and turn data into intelligence. The system architecture is shown in Figure 1. Each component in the system will be explored in following sections.

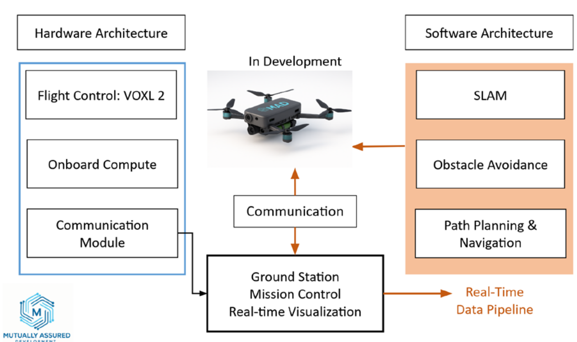

Figure 1: System Architecture of the MAD drone

2. ATTRITABLE UAV DESIGN

The first component of the system is the attritable sub-250g UAV. A summary of the design is provided below with more details available in [6]. The MAD team conducted customer interviews to better understand firefighter and inspector needs in order to define the UAV specifications. The targets included low cost, long flight time, and an interchangeable mission-specific sensor suite. Between the FCC's restrictions on foreign-made UAV components and the requirement that many federal and municipal governments purchase US-manufactured goods, the MAD team elected to target all Blue UAS certified components.

2.1. *Mission-Specific Sensor Suite*

Different types of missions require different UAV sensor packages. Customer discovery with first responders identified six types of sensors that were desirable: IMU for UAV stabilization, RGB camera for visual navigation and mapping, time of flight (ToF) camera for depth estimation, thermal camera for identifying fires or people, gas sensors for evaluating chemical leaks, and microphones for communication.

Due to the sub-250g limit, it was not possible to incorporate all sensors into a single UAV. However, different UAVs could be prepared with different mission-specific sensor suites. For example, a mission could involve one UAV with RGB, ToF, and thermal camera plus a second UAV with gas sensor and microphone. The initial UAV design includes only RGB, ToF, and thermal cameras.

2.2. *UAV Design for Autonomy*

UAVs require a flight controller. Adding autonomous navigation with simultaneous localization and mapping (SLAM) requires substantial processing power. This may be achieved by combining the flight control with a companion computer. Some products combine the flight controller and companion computer into a single custom board (e.g., ModalAI VOXL 2) while others mount a discrete companion computer (e.g., Raspberry Pi CM4 or NVIDIA Jetson).

The sub-250g mass constraint imposed significant limitations on the computing architecture of the system. Discrete board solutions, such as the Raspberry Pi or standalone microcontrollers, were evaluated but ultimately deemed infeasible due to the cumulative mass penalties associated with interconnecting wiring, electrical connectors, and mechanical mounting hardware. These parasitic mass contributions, while individually minor, collectively exceeded the available mass margin when considered alongside other required subsystems. To address this, the ModalAI VOXL 2 integrated computing platform was selected.

2.3. UAV Chassis Design

The UAV chassis design is a traditional x-type quadcopter. For ease of manufacturing, the design is mostly flat with slots for mounting a sensor package and battery retaining clips. The arms were reinforced with a T-shaped cross section to increase bending stiffness.

Chassis material selection was a key factor within the total mass budget. Five candidate materials were assessed based on four criteria: mass efficiency, fabrication complexity, structural strength, and cost. The considered materials were PLA, ABS, PEEK, woven carbon fiber (T800 3K), and Markforged Onyx.

Standard thermoplastics like PLA and ABS were found inadequate for the platform's structural needs, lacking the stiffness and durability necessary to handle sustained motor torque and operational loads. High-performance options, such as PEEK and woven carbon fiber (T800 3K), provided good strength but introduced significant fabrication challenges and geometric limitations that did not suit the compact, detailed airframe design of the MAD platform.

Markforged Onyx with Continuous Fiber Reinforcement was identified as the best choice, combining a chopped carbon-fiber-infused nylon matrix with embedded continuous carbon fiber strands to deliver mechanical properties nearing aluminum while maintaining the low weight and design flexibility typical of additive manufacturing. However, due to lead times for procurement, Onyx was not used. Two interim chassis solutions were created: one using carbon fiber-reinforced PLA filament through FDM printing, and another machined directly from a carbon fiber composite slab.

3. ROS 2 AND DIGITAL TWIN

The MAD platform was designed to go beyond a conventional flying sensor and serve as an active decision-making support tool. To achieve this, its software architecture is built around a real-time digital twin — a live virtual replica of the physical environment. In hazardous indoor scenarios, raw telemetry and 2D video feeds can overwhelm incident commanders with unstructured information. By grounding the system in the Robot Operating System (ROS 2), the platform achieves modular communication and flexible sensor integration. This ROS 2 layer is directly coupled with an Unreal Engine interface, which renders a real-time 3D reconstruction of the scene, overlays thermal hotspot data, and spatially maps UAV sensor readings within the digital twin.

3.1. ROS 2 Architecture and Node Structure

The software backbone of the MAD platform is organized as a ROS 2-based distributed architecture that supports real-time sensing, state estimation, mapping, and communication with the digital twin interface. ROS 2 was selected because of its modular publish–subscribe framework, which allows each sensing, processing, and visualization function to operate as an independent node while remaining synchronized across the UAV and operator workstation.

At the UAV level, the architecture begins with the VOXL sensor publisher, which exposes onboard sensing and state-estimation streams to the ROS 2 ecosystem. These include raw visual-inertial odometry (VIO) data, front tracking camera imagery, and ToF point cloud measurements. Since the ModalAI platform natively outputs odometry in the NED (North-East-Down) coordinate convention, a dedicated ROS 2 node, qvio_ned_to_enu, converts this data into the standard ENU (East-North-Up) frame used by the rest of the ROS stack. This step ensures compatibility with downstream localization, SLAM, and visualization modules.

A second node, camera_info_pub, publishes the intrinsic calibration parameters of the front tracking camera, enabling perception and mapping packages to correctly interpret incoming image streams. In parallel, the rgb_compression node compresses RGB imagery before transmission, reducing bandwidth demand while preserving the visual information needed for teleoperation, scene understanding, and Unreal Engine rendering.

The converted odometry is passed into an Extended Kalman Filter (EKF) fusion layer, which produces a refined vehicle state estimate on /odometry/filtered. This filtered odometry serves as a motion prior for the mapping subsystem. On top of this, RTAB-Map performs real-time SLAM by combining filtered odometry, camera data, and ToF point cloud input to incrementally reconstruct the environment. Its outputs include both a 2D occupancy map and a 3D point cloud map, which form the geometric basis of the digital twin.

The overall frame hierarchy follows the standard ROS 2 transform tree:

map → odom → base_link → sensor frames

where map is generated by RTAB-Map, odom represents the local odometric frame, and base_link corresponds to the UAV body frame. Sensor frames, such as camera_link, tof_link, and imu_link, are rigidly attached beneath the base frame through the robot description model. This node-based structure enables clear separation of responsibilities across the system:

- **sensor interfacing** through VOXL ROS publishers,
- **coordinate conversion** through qvio_ned_to_enu,
- **camera calibration publishing** through camera_info_pub,
- **data stream optimization** through rgb_compression,
- **state estimation** through the EKF, and
- **mapping/localization** through RTAB-Map.

Because ROS 2 supports interoperable communication between onboard computation and external visualization tools, the resulting architecture can stream processed spatial data directly into the Unreal Engine digital twin. This makes it possible for commanders to view not only live video, but also a continuously updated 3D scene reconstruction with spatially registered UAV observations, improving situational awareness in hazardous indoor operations.

3.2. ROS 2-to-Unreal Engine Bridge

The translation of ROS 2 data to Unreal-usable data is done using the Unreal Engine bridging plugin rclUE, made by Rapyuta Robotics [7]. This middleware layer, compatible with Unreal Engine for Linux, enables bidirectional communication between the runtime of Unreal Engine and the ecosystem of ROS 2. The plugin utilizes the ROS 2 client library, also known as "rcl", and adapts it to allow for the integration of ROS 2 nodes directly within Unreal Engine. As a result, Unreal applications are able to publish or subscribe to ROS 2 topics, and call different functions and services, amongst other functionality. This architecture preserves the ROS 2 data-centric approach while exposing it through the object-oriented framework of Unreal Engine. This allows simulation entities in Unreal Engine, such as actors or components, to behave as fully functional ROS 2 nodes.

The rclUE bridge operates by mapping different message types from ROS 2 to Unreal-compatible data structures. Sensor outputs and commands are transmitted via the ROS 2 topic system, then picked up via the destination computer's Unreal Engine via rclUE subscribers. Conversely, the same system allows for Unreal components to publish their own topics and nodes to be fed back to the ROS 2 network. This greatly assists with the use of ROS 2 and Unreal in tandem for a variety of tasks, especially visualization.

MAD has developed middleware between rclUE and Unreal Engine's LiDAR Point Cloud (LPC) functionality library. This allows for the real-time wireless visualization of LiDAR point cloud data generated by drone-mounted sensors, generally the main LiDAR camera, though others can be used if necessary. The drone publishes point cloud messages, which are typically using the "sensor_msgs/PointCloud2.msg" format, via ROS 2, which are then picked up by an rclUE subscriber. The data is taken to a different script, parsed, and converted into an Unreal LPC. Unreal itself, including its rendering pipeline and GUI, is then used to display the LiDAR point cloud data in 3D space, allowing for adjustments to the LPC itself, navigation via a user-controlled camera system, and transformations applied to the LPC field. This all allows for simple navigation, exploration, and adjustment of LPC data in real time.

As shown in Figure 2, the rclUE bridge works as expected with live ROS 2 data from the MAD drone received and presented in Unreal Engine. The Unreal viewport presents an active point cloud reconstruction with ROS 2 topic panel so communication between onboard VOXL 2 and the visualization surface is open.

Figure 3 shows the output of the data pipeline and visualization starting from a high-resolution RGB and ToF camera. The resulting 3D model is easily recognizable as part of an indoor lab environment. Together with Figure 2, this confirms the functionality of the data pipeline from the GCS to ROS 2 and then to real-time 3D scene reconstruction in Unreal Engine.

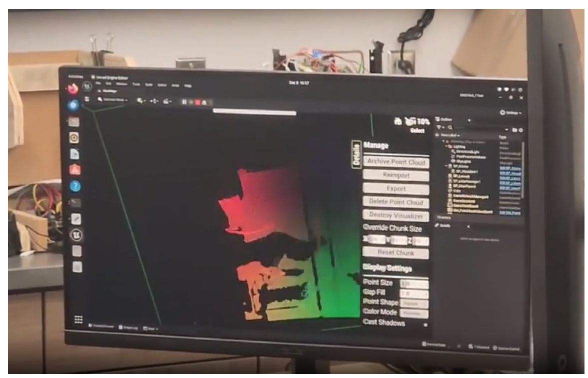

Figure 2: Real-time ROS 2 data stream from the MAD drone rendered within the Unreal Engine environment via the rclUE bridge, demonstrating successful bidirectional communication between the onboard VOXL 2 computing platform and the digital twin visualization interface.

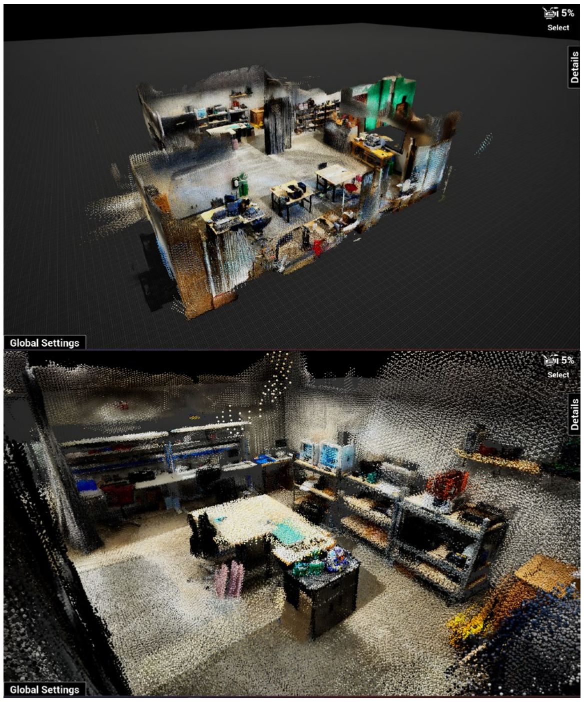

Figure 3: Colorized three-dimensional point cloud reconstruction of an indoor environment generated from ToF depth and RGB sensor data.

4. CONCLUSIONS

In conclusion, an intelligence and inspection platform has been developed. The platform includes an attritable sub-250g UAV, a ground control station, and a visualization interface for users. The UAV is suitable for operating inside complicated GPS-denied environments. The ROS 2-to-Unreal Engine data pipeline has been demonstrated to render visual data from the UAV and produce high-fidelity 3D models from higher resolution ToF depth camera data.

5. REFERENCES

[1] J. Yang, T. Ding, Q. Deng, Z. Li, Y. Wang, J. Wu and M. Shi, "A Multi-UAV Indoor Air Real-Time Detection and Gas Source Localization Method Based on Improved Teaching–Learning-Based Optimization," *Atmospheric Environment,* vol. 318, p. 120200, 2 2024.

[2] H. U. Unlu, D. Chaikalis, A. Tsoukalas and A. Tzes, "UAV Indoor Exploration for Fire-Target Detection and Extinguishing," *Journal of Intelligent & Robotic Systems,* vol. 108, no. 3, p. 54, 7 2023.

[3] A. Khan, S. Gupta and S. K. Gupta, "Emerging UAV technology for disaster detection, mitigation, response, and preparedness," *Journal of Field Robotics,* vol. 39, no. 6, pp. 905-955, 9 2022.

[4] L. Zhuang, X. Zhong, L. Xu, C. Tian and W. Yu, "Visual SLAM for Unmanned Aerial Vehicles: Localization and Perception," *Sensors,* vol. 24, no. 10, p. 2980, 5 2024.

[5] S. Karam, F. Nex, B. Chidura and N. Kerle, "Microdrone-Based Indoor Mapping with Graph SLAM," *Drones,* vol. 6, no. 11, p. 352, 11 2022.

[6] R. Salehzadeh, C. Ortolan, A. Mogulla, A. Mohammed Zia, S. Stepanek, Y. Lee and J. Mynderse, "System Design and Evaluation of a Lightweight Micro-UAV for Emergency Response," *Drones,* vol. 10, no. 6, 2026.

[7] Rapyuta Robotics, "rclUE: A ROS 2 Client Library for Unreal Engine," [Online]. Available: https://github.com/rapyuta-robotics/rclUE.

6. CONTACT INFORMATION

Yeen K. Lee
A. Leon Linton Department of Mechanical, Robotics, and Industrial Engineering
Lawrence Technological University
21000 West Ten Mile Rd.
Southfield, MI
ylee1@ltu.edu

7. ACKNOWLEDGMENT

This work was funded by Mutually Assured Development, LLC. Blue Devil Ventures serves as the managing studio entity supporting Mutually Assured Development, LLC.

8. ACRONYMS

EKF	Extended Kalman Filter
ENU	East North Up
FCC	Federal Communications Commission
FDM	Fused Deposition Modeling
GCS	Ground Control Station
IMU	Inertial Measurement Unit
LPC	LiDAR Point Cloud
NED	North East Down
ROS	Robot Operating System
SLAM	Simultaneous Localization and Mapping
ToF	Time of Flight
UAS	Unmanned Aircraft System
UAV	Unmanned Aerial Vehicle
VI	Visualization Interface
VIO	Visual-Inertial Odometry