2022 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
AUTONOMY, ARTIFICIAL INTELLIGENCE & ROBOTICS TECHNICAL SESSION
AUGUST 16-18, 2022 - NOVI, MICHIGAN

APPLICATIONS OF A SHARED DATA WAREHOUSE FOR GROUND VEHICLE AUTONOMY

Michael Boulet[1,3], Eric Cristofalo, PhD[1,3], Paul DeBitetto, PhD[1,3], Daniel Griffith[1,3], Andrew Heier[1,3], Sam Kassoumeh, PhD[2], Aaron Plotnik, PhD[1,3], Alan Wu[1,3]

[1]MIT Lincoln Laboratory, Lexington, MA
[2]U.S. Army DEVCOM Ground Vehicle Systems Center, Warren, MI
[3]Authors contributed equally

ABSTRACT

Many recent advances in autonomy are derived from algorithm optimization and analysis with a large volume of data. The Autonomous Mobility Through Intelligent Collaboration (AMIC) program established a resource to host and access data to accelerate autonomy capability development across the U.S. Army Robotics and Autonomous Systems enterprise. The repository is seeded with high-quality multi-modal Autonomous Ground Vehicle sensor data collected from relevant operating environments. Development of unmanned air-ground teaming capability that extends the perception and planning horizon of an individual ground vehicle exercises and informs the development of the data warehouse. Collected data was also used to train a convolutional neural network to estimate relative vehicle position from camera images for communication-free formation control.

Citation: M. Boulet, E. Cristofalo, P. DeBitetto, D. Griffith, A. Heier, S. Kassoumeh, A. Plotnik, A. Wu, ”Applications of a Shared Data Warehouse for Ground Vehicle Autonomy,” In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 16-18, 2022.

This material is based upon work supported by the Department of the Army under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Army.

1 INTRODUCTION

Datasets are increasingly becoming critical assets required to rapidly and efficiently develop advanced autonomy in future ground vehicles. Data are used throughout the autonomous ground vehicle (AGV) development lifecycle. Offline data, such as raw sensor measurements recorded during field experiments, are often provided as input to software modules to evaluate and improve their performance without requiring time-consuming hardware cycles. Stored data from a diverse set of environments can be used to validate the robustness of an algorithm across anticipated operational conditions. Furthermore, with the emergence of modern machine learning techniques, developers leverage stored data to train deep neural networks to achieve superior levels of performance relative to traditional human-designed algorithms.

Data availability tends to greatly accelerate autonomy innovation and maturation. Despite the evident utility of data at the ready, the national defense autonomous ground vehicle community lacks available datasets of relevant environments. While individual programs typically collect significant volumes of AGV data during tests and demonstrations, often at significant cost, the data are rarely shared outside of the program or platform owner.

The Autonomous Mobility through Intelligent Collaboration (AMIC) program is developing an open data architecture, guidance, and infrastructure enabling the national defense community to effectively collect, share, and leverage autonomous ground vehicle data. The AMIC open data architecture is aligned with the DoD Modular Open Systems Approach (MOSA) and U.S. Army Data Strategy principals to accelerate autonomy innovation, maximize reuse, and enable competition decoupled from vehicle ownership through shared, readily available, interoperable, protected, and trustworthy data.

This paper provides an overview of the AMIC shared data repository in Section 2. Section 3 details the development of a kit used to collect high-quality ground vehicle sensor data. The kit is deployed to relevant environments in order to seed the repository. The utility of the data hub is validated with two efforts building autonomy capability with machine learning. Section 4 highlights the use of data to train a semantic segmentation model that executes on an unmanned aerial vehicle to inform AGV maneuver beyond the ground vehicle's sensor range. In Section 5, data are used in a supervised learning architecture to estimate the relative position of peer vehicles for formation control.

2 DATA HUB

The objectives, requirements, and design of the AMIC Data Hub are detailed in [1]. In summary, the AMIC Data Hub is a repository of data organized into datasets. Users access the Data Hub over the Internet using a web browser or through custom software via an API. Datasets are augmented with user-provided metadata such as a description, site conditions, terrain type, and activity. The system accepts arbitrary types of data. However, the Data Hub additionally processes ROS bag files, which are a common storage format used in the community, to extract additional derived metadata. Derived metadata include the location that the data was collected from GPS position reports, the timestamp of the initial and final recorded measurements, and the types of data stored in the bag. Users may query the AMIC Data Hub metadata to identify and then download specific datasets of interest.

Since the prior publication referenced above, the AMIC Data Hub has matured into an operational prototype. A typical view of the AMIC Data Hub website is shown in Figure 1. The Data Hub employs authentication and other security measures to limit access to authorized individuals.

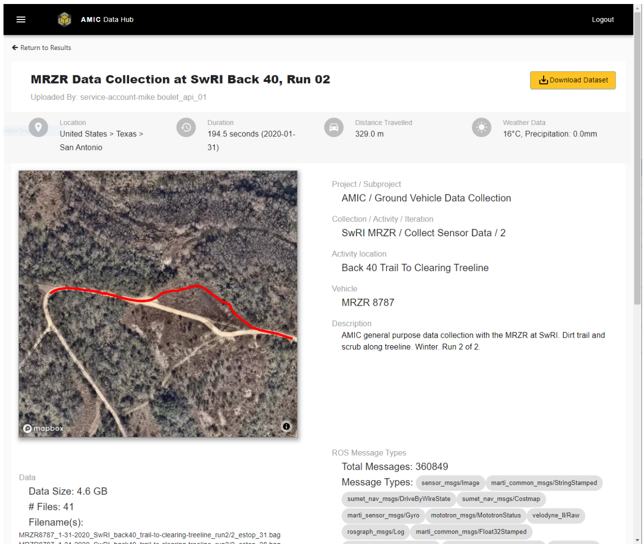

Figure 1: The AMIC Data Hub website's dataset detail page shows the vehicle's trajectory and other metadata without the user needing to download large files.

3 DATA COLLECTION KIT

The AMIC Data Hub is able to accept data recorded from any source including the U.S. Army's existing fleet of autonomous ground vehicles. However, shipping full-sized AGVs to a large number of geographically distant sites to collect data in diverse environments and conditions is logistically challenging, costly, and interferes with ongoing autonomy development efforts. The AMIC Data Collection Kit (DCK) is a modular assembly of sensors similar to the sensing suite used on Army AGVs. Deploying and integrating the DCK on site-resident human-driven vehicles greatly reduces the logistics burden to enable large-scale collection of AGV sensor data across many environments. Once data collection at a site is complete, the DCK is returned, collected data are moved to the AMIC Data Hub, and the DCK is refurbished to support the next deployment.

3.1 Design

The Data Collection Kit consists of three core components, the External Sensor Assembly, the Internal Computer Assembly, and a readily available commercial off-the-shelf (COTS) portable Power System that serves as an Uninterruptible Power Source (UPS) between the DCK and standard commercial (12V) or military (24V) electrical power sources. The sensor assembly consists of a suite of sensors commonly employed on autonomous driving vehicles. This includes multiple (6) High Definition color GigE cameras providing images at 30 fps, three Velodyne LiDARs, and a GPS/INS system. The nominal sensor configuration on the prototype DCK is shown in Figure 2. The attachment plates (both center and side plates) are designed with a grid of threaded attachment holes to accommodate easy reconfiguration of sensor positions and orientations. The DCK is also designed to record vehicle data, e.g., throttle, brake, and steering angle, through a commercial OBD2 port or CAN bus interface over a DB9 connector.

External Sensor Assembly The Sensor Assembly is designed to be modular and can be attached to either commercial or military vehicles using common rack-mounts or straps as shown in Figures 2 and 3 respectively.

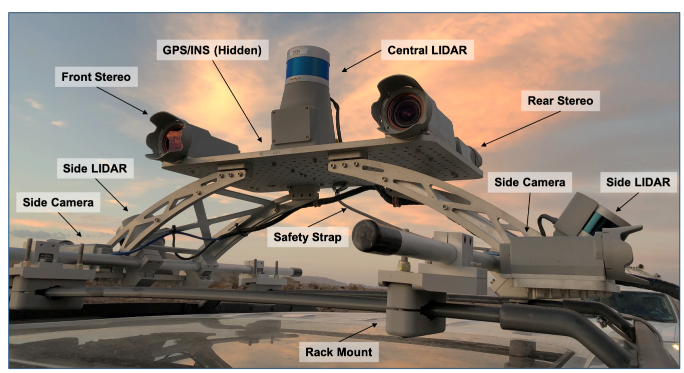

Figure 2: Data Collection Kit System External Sensor Assembly

Figure 3: DCK Mounted on Military HMMWV

Internal Computer Assembly The Internal Computer Assembly contains a ruggedized 9th-gen i7 CPU with 10x1GigE and 2x10GigE network interfaces and 4x4TB SATA III hot-swapable SSD drives configured as a Redundant Array of Independent Disks (RAID-5) for both data integrity and improved read/write performance. The entire embedded computer system consumes approximately 65W of power.

Portable Power System A COTS Portable Power System (PPS) serves as an uninterrupted power source to the kit in the event that vehicle power is disrupted. The DCK was designed to operate at less than 120W of total power which is supportable by commercial vehicle's accessory power outlets and easily supplied by military vehicle auxiliary power over standard NATO output ports. Nominally the PPS remains charged via either 120V AC shore power or through 12/24V vehicle DC output power. The PPS is capable of providing approximately 136W (13.6V @ 10A) of continuous power for approximately 2.0 hours (288Whr) or approximately 2.5 hours for the actual DCK.

3.2 Field Tests and Results

DCK field tests have been conducted at Fort Devens, Massachusetts and at the National Training Center (NTC) at Ft. Irwin, California in late August of 2021. The goal of the NTC collection was to exercise the DCK deployment concept of shipping multiple DCKs to a remote location, installing the kits onto vehicles on-site, performing collections in realistic military terrain environments, offloading the collected data, uninstalling, and returning the kits. The two test vehicles utilized were Ford Explorers, shown in Figure 4. The two kits were shipped to the site via standard UPS ground shipping using a single crate to house the sensor assemblies plus two Pelican cases for the supporting computer hardware. Once the team arrived, the kits were installed onto the the test vehicles within a time-frame of approximately one hour for each vehicle.

The test plan for data collections at NTC included a matrix of test conditions covering a variety of terrain and road classes (Figure 5), lighting conditions (Figure 6), speeds, distances, and vehicle maneuvers such as formation follow.

In total, the team recorded approximately 40TB of data, covering 500 miles of terrain, with 32 hours of active collection, over 5 days of continuous testing in conditions where ambient temperatures reached $\sim$105°F each day with no system failures.

Figure 4: Ford Explorer Test Vehicles Used at NTC

Figure 5: Terrain Types Explored at NTC

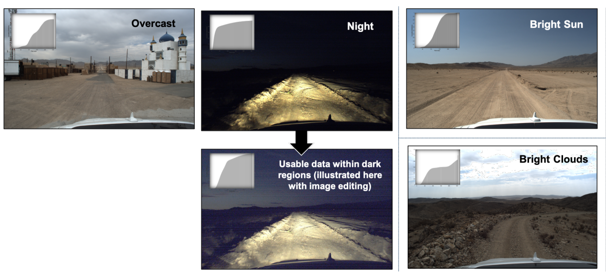

Figure 6: Lighting Conditions Collected at NTC

3.3 Future Work

The DCK concept has been successfully prototyped and field tested in relevant military terrain at Devens and NTC. Future collections are being considered at additional Army training locations for increased diversity in terrain environments and seasonal conditions. Future kits may also carry additional sensing modalities such as radar sensors and infrared/thermal night vision cameras. There is also considerable interest in using an increased number of kits attached to a greater variety of Army vehicles and collecting multi-vehicle data during live training exercises. Recognizing the prototype DCK instance described herein may not be compatible with all Army vehicle types of interest, the DCK concept is being expanded to include variant designs capable of being more easily affixed to a variety of combat vehicles where available onboard space is extremely limited, and there may be no common points for attachment. It is also critically important to ensure the DCK does not interfere with active vehicle components required during live training objectives.

4. UNMANNED AIR-GROUND TEAMING

Most of the research and development today for autonomous ground vehicles focuses on single vehicles. However, many people across the DoD anticipate routine battlefield teaming. That is, they envision that future ground vehicles will perform missions while teaming with humans, other ground vehicles, or even other types of autonomous systems. As both a path-finding use-case for the AMIC effort and to realize an application for the teaming of air and ground vehicles, we have demonstrated a selected teaming concept.

The general idea of an air-ground team is to maximize overall mission effectiveness by combining the advantages of the different vehicles. For example, consider a navigation to cover mission. An air vehicle has the ability to fly above the ground and quickly scan the area to identify routes and detect other entities in the scene. The ground vehicle has the ability to move significant equipment and resources over long distances quickly while automatically routing locally using onboard perception sensors. This is illustrated in Figure 7.

4.1 Approach

Our concept involves an unmanned aircraft that is deployed from a ground vehicle. We consider an air-ground team where the ground vehicle is equipped with GPS and a LiDAR . The air vehicle is equipped with an INS system and downward facing camera and LiDAR. Neither team member is given any *a priori* map or routing information, simply a goal location for the ground vehicle to reach. The approach is designed to require only a low level of communications between the ground and air vehicle, and to tolerate it being intermittent.

In order to keep the distance over which the UAV can explore for routes high, we chose to allocate the map and routing processing to the UAV and therefore not require tethering, high-bandwidth communications or re-docking for the data to be processed on the ground vehicle's computers.

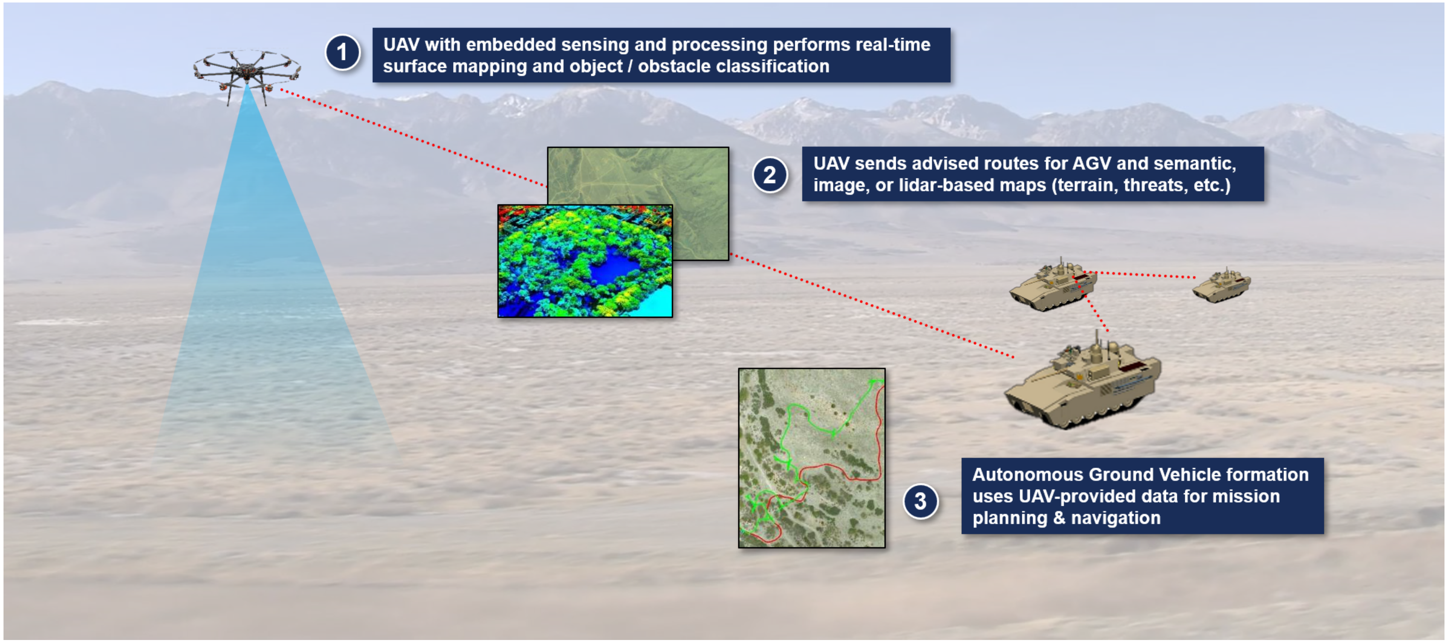

Figure 7: Air-ground teaming navigation concept

Compute capabilities in relatively small packages now make this feasible for UAVs in this class (see section 4.3 for details), though each algorithm must still be carefully chosen to maintain accuracy while keeping computational overhead reasonable. This allocation could be evaluated differently if teaming systems were limited only to smaller UAVs for the task.

At the start of a navigation mission, the ground vehicle launches the unmanned aircraft which is equipped with embedded sensing and computing. The aircraft then explores the area in the direction of the goal and maps the terrain in real time. This map includes 3D information, ground terrain classification, and object/obstacle detection. The aircraft then sends advised routes to the ground vehicle, plus more detailed map information as communication constraints permit. The ground vehicle then uses this advised route to begin navigating to the goal. As the vehicles move through the space, this perception and planning loop happens continuously and updates as appropriate until the ground vehicle reaches the destination. For example, if the aircraft detects an impassable obstacle along the currently advised route, then it covers new ground to explore and recalculate a new, feasible route, which is then communicated to the ground vehicle.

4.2 UAV Autonomy Stack

The autonomy stack is shown in Figure 8 and consists of three main areas: 1) 3D perception, which is fed by 2) Off-line Model Training, and 3) Planning.

3D Perception. Our unmanned aircraft's perception pipeline involves three steps. First, read in downward imagery and estimate semantic labels for each pixel using a semantic segmentation neural network model. Our implementation runs on a GPU and is based on MIT's ADE20K scene parsing dataset [2]. The semantic labels correspond to terrain and object classes relevant to ground vehicle navigation such as grass, gravel, water, and trees. Second, we fuse the semantic labels with LiDAR point clouds by associating pixels to points. The result of this process gives us points in 3D space with

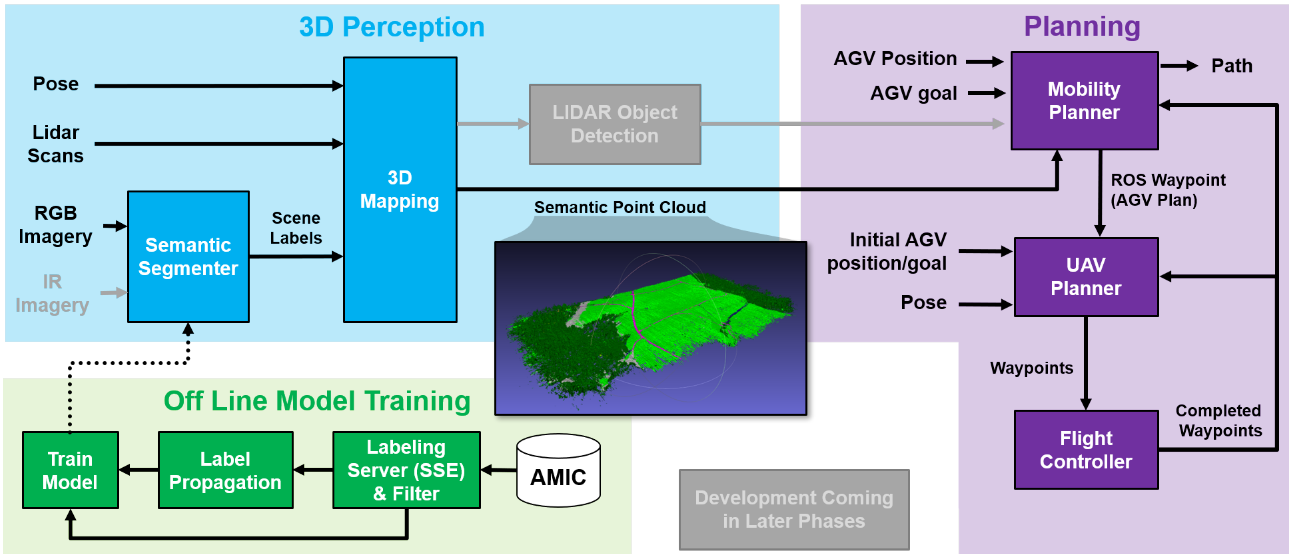

Figure 8: UAV autonomy stack

semantic labels. And finally, we fuse these points into a 3D metric-semantic mesh using Kimera, which is a real-time system that runs on the CPU [3].

Off-line Model Training. In order to train our semantic segmentation network running on the unmanned aircraft, we collected overhead imagery over different sites and throughout various seasons. We hand labeled a small collection of images (less then 100). While this is generally considered too small a training set for robust semantic segmentation, we found that performance was sufficient to support our experiments. See Figure 9 for an example.

Planning. The pipeline then flows the 3D metric-semantic mesh to planning algorithms. The *AGV Mobility Planner* analyzes the mesh for best routes for an AGV, while also identifying the most promising frontier points in the mesh. These ranked frontier points are then used by the *UAV Coverage Planner* to decide how to best cover new areas and expand the mesh autonomously in real-time.

AGV Mobility Planning. The UAV generates a recommended global trajectory for the AGV by first translating the 3D metric-semantic mesh into a cost map that corresponds to estimated vehicle traversability costs. The cost at each vertex is function of the following extracted features [4]:

- Terrain class (i.e. grass, dirt, gravel, etc)
- Ground Slope
- Terrain Roughness

Once the AGV position has been observed by the UAV and is part of the mesh, the AGV planner starts generating candidate trajectories towards the goal. In most cases, the UAV has not built a complete map of the feasible region, so the UAV first generates a set of feasible trajectories to promising frontier points using the Batch Informed Trees planning algorithm [5]. For each trajectory to a frontier point, the full trajectory cost is the sum of the discovered path and the estimated cost-to-go value from the frontier point to the goal. The cost-to-go value is a function of the frontier point cost and Euclidean distance from the frontier point to the goal. The AMIC AGV Planner generates a trajectory to the goal position if possible and then selects the minimum cost trajectory to pass to the AGV. An example planned trajectory is illustrated by the red path in Figure 11.

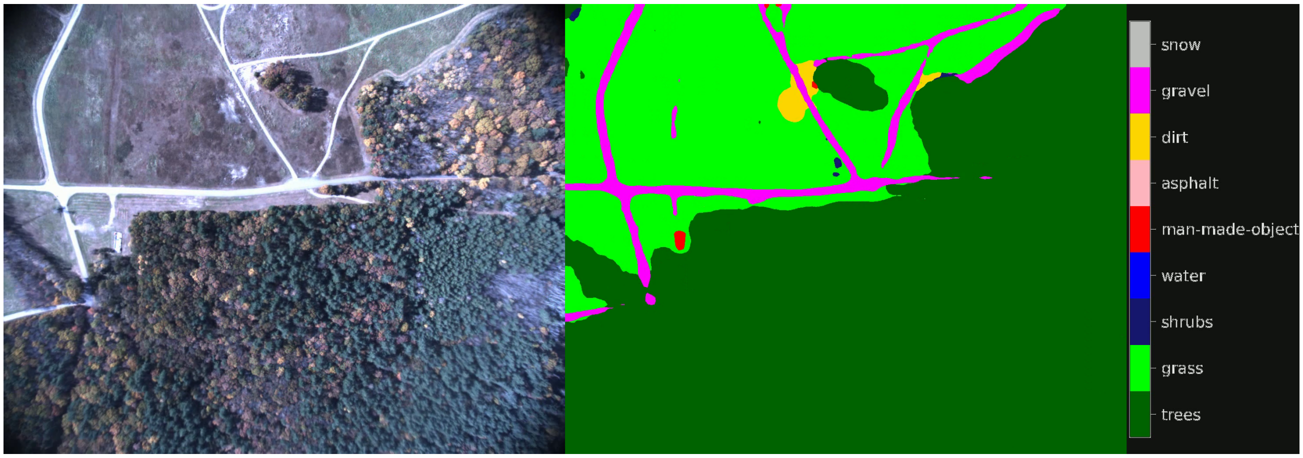

Figure 9: An example overhead image and the semantic segmentation model's predicted labels.

UAV Coverage Planning. The UAV Coverage Planner is responsible for guiding the UAV's sensors towards novel, unexplored terrain so that it can build a complete map of the world below. Our method is a *next-best-view* planner, where the most promising frontier point from the AGV mobility planner is used as the next, new direction to explore. The planner first computes a desired coverage region (2D polygon) that covers new space beyond this next best frontier point. We employ a polygon coverage planner [6] to plan the UAV survey flight pattern within the desired coverage region. This coverage planner respects constraints such as geofences, maximum velocities, and sensor field-of-view footprints that ensure the entire region is covered. We finally send the survey pattern waypoints to the UAV's flight controller via a MAVLink mission command that can be automatically executed by the UAV. An example planned survey pattern is illustrated by the blue path in Figure 11. The UAV planner repeats this process of selecting new frontier directions and flying coverage patterns until the AGV reaches the goal.

4.3 Experiments

The ground vehicle platform is a Polaris MRZR that has been modified for autonomous driving capability. Here it was configured to compute local perception information using its onboard LiDAR and to navigate using route waypoints communicated by its UAV team partner. The control, perception and planning autonomy stack was the RTK Core 2019.

The Harris Aerial Carrier H6 Hybrid serves as the aerial platform. It is controlled by an autopilot and carries an Nvidia Jetson AGX for all onboard perception and planning. The unmanned aircraft is equipped with an SBG Ellipse-D INS system for accurate localization, and a FLIR Chameleon3 EO/RGB camera plus the Velodyne VLP32C LiDAR for terrain sensing. The vehicle and payload assemblies are shown in Figure 10.

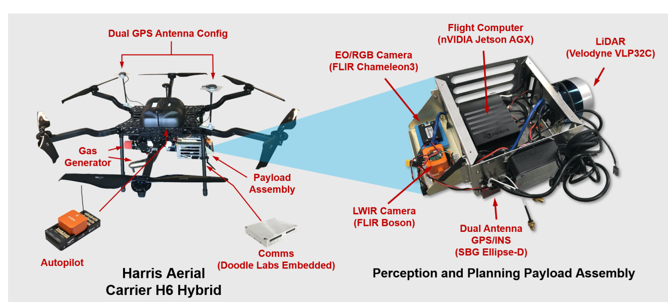

Figure 10: UAV demo hardware configuration

Several experiments with the teamed navigation mission were undertaken in the fall of 2021 at Fort Devens. In Figure 11, the UAV-generated traversability map is shown overlaid on top of overhead satellite map and the resulting recommended route to the goal is shown. The MRZR was able to traverse to the goal using that recommended route, aided by local corrections from its internal local perception and replanning, fully autonomously and without intervention by the safety driver, traversing in about 3 minutes. In comparative experiments with the MRZR given a comparably far point to reach in the same environment, and without any *a priori* map or route information, the vehicle took about twice as long as it felt its way using only local perception and some interventions were required by the safety driver. While upgrades and tuning to the local perception and planning algorithms onboard are ongoing that will improve performance in the solo situation, this nonetheless demonstrates the utility of the freshly generated and more global map provided by the aerial team partner.

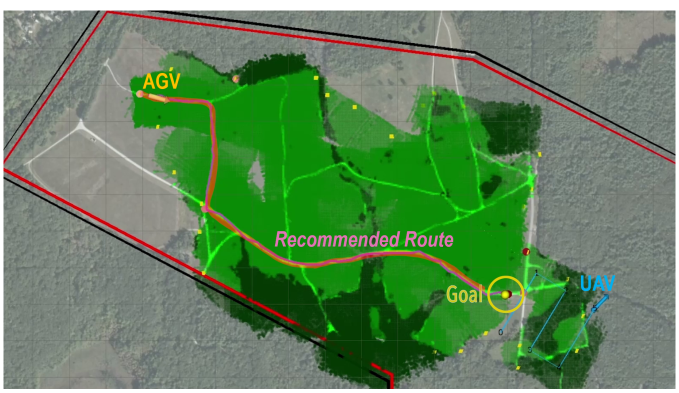

Figure 11: UAV-AGV teaming experiment

5 AI FOR FORMATION CONTROL

Multi-vehicle coordination is an essential capability for successful military operations. As such, formation control of ground vehicles represents a key component in many military tasks involving multiple vehicles for coordinated tasks. Traditional approaches to formation control require knowledge of vehicle pose (i.e. location and orientation) through the broadcasting of GPS coordinates and heading information. However, hostile environments where GPS may be denied or broadcasting of vehicle pose may present a threat to the convoy are particularly challenging for the formation control task. Execution of effective vehicle formations can still be achieved through relative vehicle state estimations in the absence of global information. Therefore, we propose to address these environmental limitations by training a convolution neural network (CNN) to estimate the state of neighboring vehicles from only visual RGB inputs. We demonstrate the effectiveness of our approach on a custom dataset collected at Fort Devens and show its ability to operate in real-time.

5.1 Approach

We train a CNN using M3D-RPN [7] for object detection and pose estimation. This model uses only a monocular RGB image as input and outputs 3D bounding box coordinates for each detected vehicle as well as its orientation relative to the camera on the observer vehicle. We train the model using a custom dataset collected with the DCK from Section 3.

We note that the algorithm currently only performs vehicle detection, while algorithm development for tracking may be done in the future. As it stands, two vehicles of identical appearance would not be distinguished by the algorithm. However, a tracking algorithm could take into account a (recent) history of vehicle movement, so that visually similar vehicles could be individually followed.

Data Collection Public datasets for autonomous driving such as KITTI [8], nuScenes [9], and Waymo [10] provide data for vehicle detection and relative state estimation, but are limited to urban vehicles and environments. To more closely mimic realistic settings for military applications, we collected a custom dataset at Fort Devens with an MRZR and a Chevrolet Suburban as vehicle targets for detection.

The DCK used for our data collection was mounted on the observer vehicle, a Ford Explorer, which captured RGB images of the target vehicles while all three vehicles moved in common formation patterns. Figure 12 shows the setup of the three vehicles used for our data collection. In total, our dataset is composed of about 47k images with diverse background and lighting conditions, as well as a large range of relative vehicle angles so that a variety of perspectives of the target vehicles was collected. For our initial work, we have 2 classes of vehicles: MRZR and Suburban. We hope to expand upon the variety of military vehicles in our collection once more types become available.

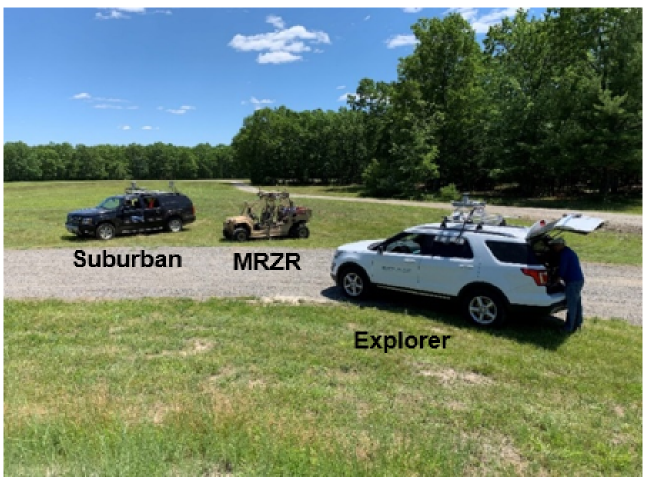

Figure 12: Vehicles used for our data collection at Fort Devens. The MRZR and Suburban were target vehicles, while the Explorer equipped with the DCK was the observer vehicle.

Label Creation Traditional labeling of objects in images has followed a labor-intensive manual process. However, since all our vehicles (observer and targets) are equipped with GPS and INS modules in addition to cameras, we leverage these data to automatically create our labels. Using GPS time to synchronize across vehicle platforms, our labels include relative position (x, z) and heading (θ) of target vehicles with respect to the observer camera, as well as vehicle class (i.e. MRZR, Suburban) and dimensions. Figure 13 illustrates the relative pose parameters from a birdseye view diagram reflecting the setup in the camera view above it.

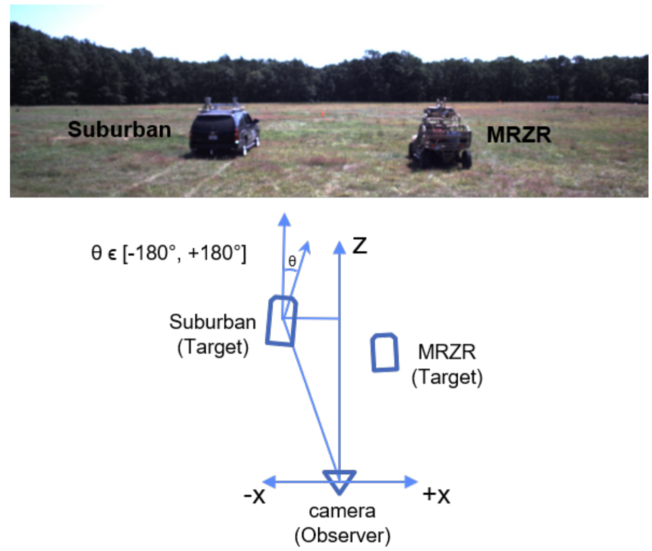

Figure 13: Birdseye view of relative vehicle position (x,z) and orientation (θ) with respect to camera.

Given camera parameters, relative vehicle location is projected into pixel space, and bounding boxes may be drawn on the image to inform the estimated relative state of the target vehicles. Examples of annotated images are shown in Figure 14. Note that only RGB imagery is used during inference, while GPS location and odometry are used only to create labels for training.

Figure 14: Sample annotations of the MRZR and Suburban vehicles from various relative poses.

Design Implementation We train a CNN based on M3D-RPN (Figure 15) to detect target vehicles and estimate their relative position and heading. M3D-RPN enhances upon the region proposal network (RPN) first proposed in Faster R-CNN [11], tailored for 3D. The RPN acts as a sliding window detector which scans every spatial location of an input image for objects matching a set of predefined anchor templates as defined in [7]. The matches are then regressed from the discretized anchors into continuous parameters of the estimated object. An example is shown in Figure 16.

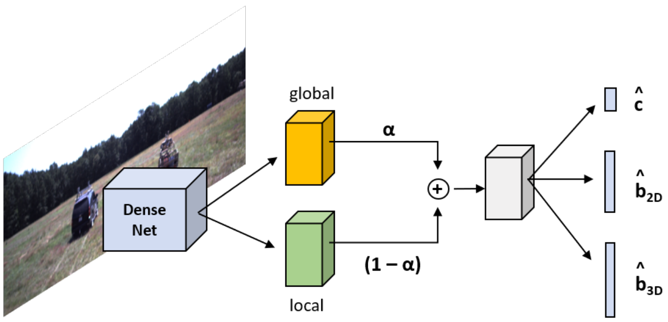

Figure 15: Overview of our system, where $\hat{c}$, $\hat{b}_{2D}$, and $\hat{b}_{3D}$ are the predicted class, 2D and 3D bounding box parameters, respectively, and α is a learned weight.

Figure 16: Sample relative state estimation.

The M3D-RPN architecture is comprised of a DenseNet [12] backbone followed by two parallel paths for global and local feature extraction. The global features use regular spatial-invariant convolution, while the local features denote depth-aware convolution that uses non-shared kernels in the row-space. These features are combined via a learned weight α and used for the multi-task framework where we simultaneously learn the vehicle class as well as 2D and 3D bounding box parameters as defined in [7]. The multi-task loss function is as follows:

$$L = L_c + \lambda_1 L_{b_{2D}} + \lambda_2 L_{b_{3D}} \quad (1)$$

where L_c is a multinomial logistic loss, $L_{b_{2D}}$ is a logistic loss for 2D bounding box learning, and $L_{b_{3D}}$ is a smooth L1 regression loss for 3D bounding box learning; λ_1 and λ_2 are hyperparameters.

5.2 Experimental Results

Our dataset is composed of about 47k images, which we performed a 90/10 split for training and test. The input image size is 512x1760 pixels, while the size of the sliding window (i.e. the input to the CNN) is 224x224. We use stochastic gradient descent (SGD) as our optimizer with a learning rate of 0.002, which we adopt from the original M3D-RPN paper. We are able to perform real-time inference at 0.2 sec per image on a Titan X GPU.

We achieved a detection rate (DR) of 92% and 93% for the MRZR and Suburban, respectively, as shown in Table 1, which also includes positional and angular accuracy measures. We suspect that the elongated body shape of the Suburban may have led to noticeably lower angular error than that for the MRZR. In general, errors observed are more related to the visual appearance of the vehicles (i.e. when the observed vehicle is partially off camera, occluded, or too far), than specific to any particular formation. Regardless, we believe the accuracy achieved for both vehicles is adequate for formation control tasks.

	DR	e_x (m)	e_z (m)	e_θ (°)
MRZR	92%	0.78	1.21	10.42
Suburban	93%	1.68	1.96	6.82

Table 1: Accuracy results on our custom dataset.

5.3 Discussion and Future Work

We addressed the challenge of formation control in hostile environments where location broadcasting is forbidden by learning to estimate relative state estimation of neighboring vehicles with a CNN using only visual imagery. To attain representative military settings, we collected a custom dataset at Fort Devens that included an army vehicle, the MRZR, to train our network. Having achieved good accuracy and as our model is able to run inference in real-time at about 0.2 sec per image, we plan to feed the estimated state of target vehicles to a planner and operate a closed loop system for formation control. In the near term, we are planning a vehicle following task. Eventually, we hope to adapt to more complex formations and environments.

6. CONCLUSION

The AMIC data hub contains a large volume high-quality data collected from sites similar to anticipated AGV operating environments. We have demonstrated the ability to use the data to train machine learning models to provide autonomy capability for air-ground teaming and formation control. We believe that continued growth in the volume and diversity of data stored in the data hub will enable the community to build and validate innovative algorithms to realize the Army's RAS vision.

7. REFERENCES

[1] M. Boulet, T. DeWeese, A. Bird, R. Kreiter, and C. Cheung, "An open data architecture for ground vehicle data-driven autonomy development and validation," in *Ground Vehicle Systems Engineering and Technology Symposium (GVSETS)*, August 2020.

[2] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba, "Semantic understanding of scenes through the ade20k dataset," *International Journal on Computer Vision*, 2018.

[3] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020. [Online]. Available: https://github.com/MIT-SPARK/Kimera

[4] P. Krüsi, P. Furgale, M. Bosse, and R. Siegwart, "Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments," *Journal of Field Robotics*, vol. 34, no. 5, pp. 940–984, 2017.

[5] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Batch informed trees (bit*): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 3067–3074.

[6] R. Bähnemann, N. Lawrance, J. J. Chung, M. Pantic, R. Siegwart, and J. Nieto, "Revisiting boustrophedon coverage path planning as a generalized traveling salesman problem," in *Field and Service Robotics*. Springer, 2021, pp. 277–290.

[7] G. Brazil and X. Liu, "M3d-rpn: Monocular 3d region proposal network for object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, Seoul, South Korea, 2019.

[8] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: the kitti dataset," *The International Journal of Robotics Research*, vol. 32, pp. 1231–1237, 09 2013.

[9] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.

[10] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov,

"Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[12] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Weinberger, "Convolutional networks with dense connectivity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.