Neptune: An Automated System for Dark Ship Detection, Targeting, and Prioritization

Adam B. Byerly, William C. Zhang, Sesan A. Iwarere, Waseem A. Malik, Sheldon F. Bish, Musad A. Haque, and Tamim I. Sookoor

ABSTRACT

The ability to detect dark ships at open-ocean scale requires enhanced space-based intelligence, surveillance, and reconnaissance capabilities. With the boom of commercial space-based sensing, the nation needs an automated process to meet the growing volume and velocity of data. Multimodal data from the variety of existing and proposed space-based sensor networks can be aggregated and fused to produce target-quality tracks on ships. These sensor modalities include synthetic aperture radar (SAR), electro-optical/infrared (EO/IR), and Automatic Identification System (AIS). In this article, we demonstrate the work of a Johns Hopkins University Applied Physics Laboratory (APL) team to automate recognition of target surface vessels from these modalities on a next-generation spaceflight processor to simulate on-orbit detection. These detections can be fused to form quality tracks that can then be used to detect dark ship anomalies via pattern-of-life analysis. Tracks formed over a continental or global scale motivate the need for further automated analysis since a significant amount of human effort would be needed to analyze thousands or tens of thousands of tracks in detail and in real time. To address this challenge, the APL team developed a suite of pattern-of-life tools that extract features from tracks and flag tracks that deviate too far from some learned definition of normality.

INTRODUCTION

Dark Ships

Crews on vessels engaging in illicit activity, such as selling oil to regimes in violation of international sanctions,[1] smuggling drugs and arms,[2] and fishing illegally,[3] routinely attempt to evade detection. A critical national security challenge is finding ways to quickly and reliably locate and identify these ships anywhere in the world. The Automatic Identification System (AIS), a radio frequency (RF) system for identifying and locating

A similar article appears in the proceedings of the Military Sensing Symposia Joint (Battlespace Acoustic, Seismic, Magnetic, and Electric-Field Sensing and Signatures Committee [BAMS] and National Symposium on Sensor and Data Fusion [NSSDF]) Conference, December 13–16, 2021.

maritime vessels, relies on ships being fitted with transceivers and broadcasting information such as their identity, position, course, and speed. These messages can be received by other vessels, as well as by coastal AIS base stations and on satellites. International law requires all voyaging ships over 300 gross tons to participate in this system.[4] Yet, crews attempting to hide their ships' identities or locations can spoof the information of other vessels or simply turn off their ships' transponders. Further, any vessel under 300 gross tons is not required to participate in AIS, giving rise to the possibility of nefarious actors leveraging smaller craft for their activities while appearing benign. In addition to ships going dark in terms of AIS transmissions, they could go radio silent by turning off other sources of RF emission through emission control. Military vessels, for instance, engage in this activity by switching to commercial radar or previously unseen war reserve waveforms to hide from sensors monitoring military waveforms. Therefore, for this article, we expand our definition of dark ships from those that are simply AIS dark to all ships attempting to evade detection or engage in nefarious activity. Furthermore, the nation needs to develop a capability to generate target-quality data on ships of interest by establishing and maintaining accurate tracks and ship identification.

There are a number of constraints to locating and identifying dark ships and developing quality tracking data on them. First, generating targeting solutions and enabling interdiction requires that location data with sufficient precision and confidence be processed, exploited, and disseminated rapidly. Second, targeting these ships anywhere in the world requires global coverage. Third, detecting nefarious vessels, whether transmitting AIS or not, requires that automated processing be enhanced so that it can "see through" the evasion and identify the true high-value bad actors, the "needles in the needle stack." Finally, it must be easy for various stakeholders to field the solution. These stakeholders include analysts and operators tracking vessels in the field, commanders planning missions at headquarters, and captains navigating vessels against adversaries in the open oceans.

Existing Capabilities

A number of commercial and nonprofit organizations are attempting to solve the dark ship problem. For instance, Global Fishing Watch[5] is a partnership among Google, Oceana, and SkyTruth to offer an unprecedented global view of commercial fishing activities. The goal is to reduce illegal and harmful activities, such as overfishing and habitat destruction, by increasing transparency on fishing activity through an online map of fishing vessel tracks from January 2012 through 3 days before the time the map is viewed. The system uses satellite-based optical, synthetic aperture radar (SAR), and Visible Infrared Imaging Radiometer Suite (VIIRS) sensors to supplement AIS and detect dark fishing fleets. While this solution provides global coverage through an easy-to-use interface, shedding light on illegal fishing activity in aggregate, it does not solve the problem of enabling real-time targeting of dark ships. Its interface also provides only the locations and tracks of all vessels and does not flag any activity as being illegal or a ship as being dark, leaving it up to a human analyst to derive these conclusions.

Another solution is HawkEye 360's[6] approach of instrumenting satellites with RF sensors to monitor widely used communications channels that can give away the position of ships that have turned off, or do not have, AIS transponders. ICEYE[7] and Spire[8] extend this approach by augmenting the satellite-based radios with SAR to detect ships engaging in emission control activity, enabling detection of a ship that goes dark in the RF spectrum but is still visible to SAR. Planet,[9] in a similar approach, provides a data feed for vessel detection that locates ships seen from their satellite optical imagery, regardless of their AIS broadcasting status. Again, these systems work well to locate vessels that are not broadcasting AIS, but they fail to adequately sift through the trove of vessel detections to identify bad actors. In particular, Planet's vessel detection feed identifies that users can "monitor [ships'] patterns of life and anomalies," but the company does not claim to have developed a capability to analyze the patterns and anomalies.

Neptune System

Similar to the approaches of Global Fishing Watch,[5] Spire,[8] ICEYE,[7] and Planet,[9] our approach, Neptune, attempts to detect dark ships through multimodal sensor fusion. However, whereas these other approaches focus on a subset of modalities that a determined adversary could attempt to evade, we are building a platform through which all available sensor modalities, including optical, SAR, and electronic intelligence, can be fused, making it harder for a dark ship to evade. Additionally, we developed a pattern-of-life (PoL) system to prioritize anomalous vessel behavior, drawing attention to high-value dark targets sooner. Finally, to optimize the quality of sensor fusion, Neptune directs sensor resources to the highest-priority ships using a game theory assignment algorithm. Figure 1 describes this closed loop activity facilitated by the Closed-Loop Collaborative Intelligence, Surveillance, and Reconnaissance Simulation (CLCSim) scenario simulator, which includes the track fusion engine. CLCSim is a software package that creates high fidelity representations of sensor–target interactions and enables the integration of sensor planning as part of the closed-loop collaborative situational awareness process. The track information generated in CLCSim is passed to our PoL system to generate

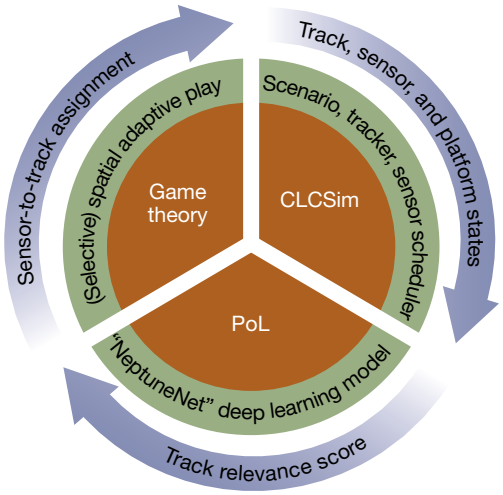

Figure 1. Overall view of the Neptune system. Onboard vessel detections are ingested and fused within the CLCSim test bed, which in turn passes state information to the PoL component for vessel analysis and track prioritization. This prioritization is used by our game theory planner to coordinate and assign sensor coverage.

relevancy scores for each track; these relevancy scores rank the tracks as more (or less) anomalous (described in detail in the section on PoL) and bring operator attention to more anomalous vessels faster. These relevancy scores are then leveraged by a game-theoretic planner (described in detail in the Sensor Fusion section) to optimally perform sensor-to-target assignment.

A main enabler of the Neptune system is the idea that low size, weight, and power (SWaP) processors with greater computational throughput, such as the NVIDIA Jetson TX2i, will be flown in space, opening up the possibility of pushing advanced processing to the spacecraft. Capella Space,[10] a commercial space-based SAR company, has already demonstrated spaceflight of a Jetson processor in space,[11] and several research universities[12,13] have near-term plans of their own to fly Jetson boards. As more commercial space companies begin flying spacecraft with commercial off-the-shelf accelerators, and processor manufacturers (NVIDIA, Xilinx, etc.) continue to improve processing capability, we can fly more sophisticated algorithms, allowing us to push our detection capabilities to the sensing edge. This advancement allows for reduced exploitation time and enhanced onboard decision-making.

With the capability to detect vessels onboard a spacecraft, we can then investigate tipping and cueing between satellites to ensure that track custody of ships of interest is maintained, even when satellites go out of theater. This conjunction of onboard target recognition with sensor fusion and planning will allow the Neptune system to more rapidly locate vessels and maintain custody of them. We rely on the multiple-hypothesis tracker (MHT) capability of the CLCSim test bed[14] to manage pseudo-tracks of vessels between sensor detections.

Neptune goes a step beyond current solutions by attempting to automate analysis and identify nefarious dark ships based on their behavior. Whereas existing projects rely on machine learning algorithms that have been hand-tuned by teams of analysts to identify certain dark ship behavior, such as Chinese fishing fleets illegally fishing in North Korea's exclusive economic zone, we are attempting to train algorithms that can discriminate anomalous activity without any predetermined patterns. This more automated solution will be useful to a number of sponsors. Our approach for behavior-based dark ship detection relies on the development of algorithms to identify PoL capturing normal vessel behavior in regions across the world. From this background of normal behavior, anomaly detectors can flag abnormal behavior for further scrutiny by an end user. Since Neptune will provide only a subset of data in an easy-to-use interface, end users can interact with the system without the overwhelming experience they might have when using interfaces that present all available data.

Remainder of Article

The remainder of this article is organized as follows: First we further develop the motivation and algorithms Neptune uses for onboard target recognition. Then we describe the sensor and data fusion work and provide an example demonstration description. Next we cover the PoL analysis for prioritizing dark ship behavior and providing contextual information to a human operator. We conclude with a summary and path to future adoption.

ON-ORBIT AUTOMATIC VESSEL DETECTION

Space-Based Observation

Earth-orbiting missions are increasingly designed around large-scale constellations of small satellites with greater heterogeneous sensor capability, data volume, and complexity of joint operations. In parallel, sensors continue to be constructed to consume less SWaP while generating larger volumes of multifaceted data, and the cost to orbit has decreased drastically with the advent of the commercial space revolution.[15] Furthermore, bandwidth limitations and ground station contact schedules often restrict analysis until several hours after an observation has occurred, driving a need for smarter on-orbit processing in the hunt for dark ships.

An increase in the number of data sources (e.g., via commercial space) also leads to a large influx of data, most of which will be on empty ocean. Moving automated detection to the tactical edge (e.g., onboard the spacecraft) enables autonomous processing of multiple data streams and decreases needed bandwidth. Such processing can aid in the faster exploitation and

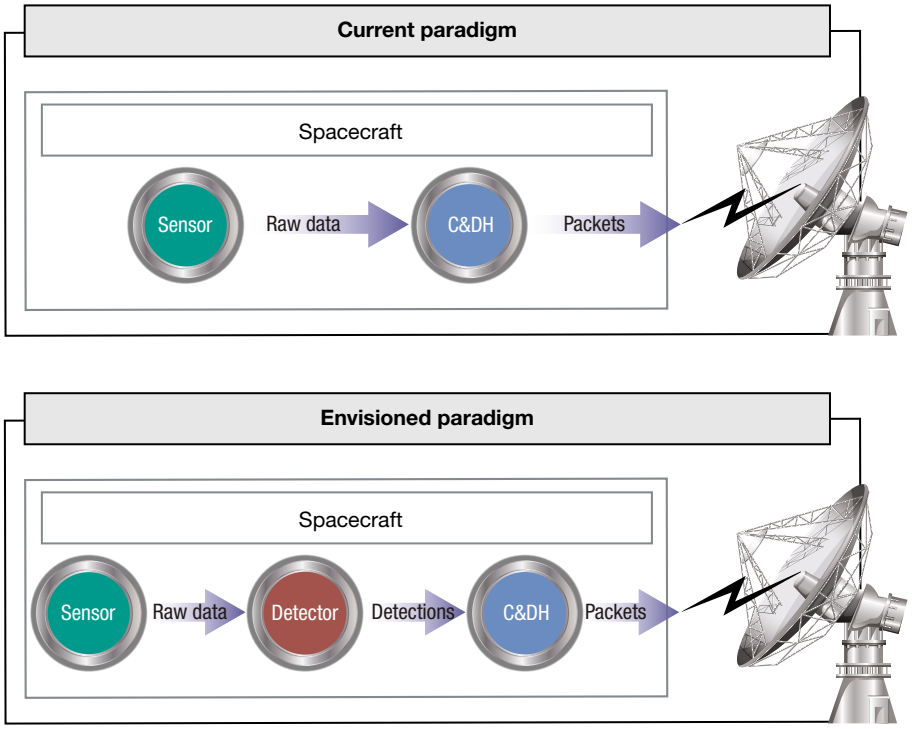

Figure 2. High-level downlink data flow. In a traditional system (top), observations are captured via an onboard sensor that typically passes raw data to the spacecraft command and data handling (C&DH) system. The C&DH system formats these measurements into packets for downlinking to the ground. With the envisioned onboard detection (bottom), these raw measurements would instead first pass through the detector to generate detections, and those detections would be passed to the C&DH system for packaging and transmission. Such detections can then be passed to other spacecraft or sensors to aid the facilitation of tipping and cueing.

dissemination of time-critical information needed to execute real-time targeting. We envision a paradigm where the spacecraft "tells" when it sees something of interest, as shown in Figure 2.

Deep Learning–Based Detection

Deep learning has been demonstrated to address the image detection and classification problem.[16,17] In particular, there has been much interest in applying deep learning to detecting surface ships.[18] Wang et al.[19] collated a data set of calibrated SAR images from two space platforms (Sentinel and Gaofen) and annotated the data with bounding boxes around surface vessels. This data set consists of images from multiple operating modes and resolutions, includes coastal and open-ocean regions, and contains scenes with single and multiple vessels. Wang et al. also demonstrated the capabilities of several deep learning detectors, achieving state-of-the-art performance (mean-average precision, or mAP) as compared to alternative techniques (e.g., constant false-alarm rate). While their goal was to consider the detection accuracy of these models, the question still remains as to the timeliness of processing and exploitation.

To address the operational challenges of pushing detection models to the edge, namely low-SWaP constraints and limited processing resources (as compared to ground-based processing), we must consider the trade-off of detection accuracy with inference time. That is, if a model performs well in detecting all vessels in an observation but fails to generate these detections in a timely manner, those detections may no longer be informative. Likewise, if a model can generate detections instantaneously but fails to identify most vessels in an observation, there may be too many missed targets for operational relevance.

To address this trade-off, we consider the ResNet family[20] of convolutional neural networks. This family has been shown to perform well in image processing tasks[20] and includes a variety of models with increasing depth. Deeper models in the family (e.g., ResNet101) achieve greater accuracy than shallower models (e.g., ResNet34) at the trade-off of inference time; deeper models take longer to execute. By using the ResNet family, we can evaluate several models of varying depth, but identical architecture, on our target hardware. With these ResNet models as our feature extraction backbone, we then consider the RetinaNet detector,[17] a state-of-the-art deep learning detector. The RetinaNet detector is both fast and accurate at identifying and localizing objects of interest in input images, giving us the best of both worlds: fast detection speed and accurate detections.

Training Details and Experimental Results

We evaluated the ResNet18, ResNet34, and ResNet50 models on an Ubuntu 18.04 machine with a 4 GB NVIDIA GTX 970 for training and an NVIDIA Jetson Nano and Xavier for inference benchmarking. We trained a PyTorch implementation[21] of RetinaNet on the Wang et al. data set (as described by Wang et al.[19]) with a batch size of eight, using each of the ResNet backbones. The results of this training are shown in Table 1.

Table 1. Vessel detection mAP and inference time

Model	mAP[a]	Inference Time	
		Jetson Nano (pixel^2/s)[b]	Jetson Xavier (pixel^2/s)[b]
RetinaNet-ResNet18	0.778	3,620	12,541
RetinaNet-ResNet34	0.803	2,560	8,868
RetinaNet-ResNet50	0.807	2,486	8,613
RetinaNet-VGG16[c]	0.914	1,252	4,337

[a] See Zhu.[22]

[b] See the NVIDIA site (https://developer.nvidia.com/embedded/downloads) for specs.

[c] From Wang et al.[19]

All three models achieve comparable validation mAP scores, with the deeper ResNet50 model scoring the best of the three. However, all three models do fall short of the reported mAP score for the RetinaNet-VGG16 model evaluated by Wang et al.[19] While these models underperform the VGG16-based model in terms of detection capability, we must also consider the inference time (i.e., the time needed to execute these models on representative low-SWaP processors). In the third and fourth columns of Table 1, we report the image area each model is capable of processing on the Jetson Nano and Jetson Xavier boards, respectively; to account for differences in image sample distance and swatch size, we normalize to square pixel area. It is clear that the ResNet family base of models significantly outperforms the VGG16-based models, with the ResNet18-based detector capable of processing nearly nine times the area per second of the VGG16 model. For context, a representative Capella SAR spotlight image covers 5 km by 5 km with a resolution of 0.5 m, resulting in an image with 10,000 by 10,000 pixels. From our analysis, the VGG16 model would require 5.3 s to process a single image on a Jetson Xavier, while the ResNet18 model would require only 0.64 s and the ResNet50 model only 1.35 s.

SENSOR FUSION

Detection of dark ships faces two challenges: (1) persistent localization and (2) identification of anomalous behavior. To obtain the best chances of persistently localizing a dark ship, track custody must be maintained on all detected ships, whether AIS broadcasting or not based on sensor modalities that do not rely on such broadcasting, like an imaging sensor. For dark ships, the best way to localize is to persistently maintain track custody on such ships from the time that they are either in close proximity to a broadcasting ship (and thus detected via an imaging sensor) or gave their final AIS broadcast. Fusion of different sensor modalities has long been exploited, particularly to produce quality tracks of maritime vessels. These sensor modalities may include SAR, electro-optical/infrared (EO/IR), and AIS and TDOA (time difference of arrival). For Neptune, EO/IR and SAR imaging modalities bring high-resolution and all-weather ship imaging capabilities to bear for the dark ship detection problem. Here, we fuse multisensor data over time to establish track histories and maintain track custodies of a large number of maritime vessels, particularly those that have gone dark. To fuse data from these modalities, we leverage CLCSim as a high-fidelity scenario simulation test bed on which the Neptune dark ship detection and tracking system is being developed.

CLCSim is a C++ software simulation platform for developing, testing, and analyzing closed-loop collaborative ISR, sensor data fusion, estimation, control, and optimization algorithms across maritime, sea, air and space domains. Embedded within CLCSim is the Precision Engagement of Moving Targets (PEMT) tracker, which oversees the filtering and MHT processes that manage track states. PEMT is an unclassified flexible MHT software package that is capable of handling measurements from multiple sources and source configurations. For more information on CLCSim, refer to Newman and DeSena.[14]

CLCSim provides representations of sensors, targets, and the track fusion engine as well as facilities for sensor resource management. The sensor resource management facility of CLCSim enables us to plug in an arbitrary sensor planner that meets specific needs. Here, CLCSim passes track data to Neptune's PoL module for anomaly detection of dark ships, then we integrate a game theoretic framework to establish tipping and cueing between satellite sensors based on the designation of suspected ships, which will allow for enhanced situational awareness of dark ships. We will also be using the existing tipping and cueing algorithms that are built into CLCSim, in particular a greedy algorithm (see the Sensor Resource Management in CLCSim section below).

Table 2. Demo input parameters

Field	Value
Update rate	1 Hz
Orientation	Yaw = 0, pitch = –90, roll = 0
Field of regard (FOR) angle	130°
Probability of detection (P_D)	0.999
False-alarm rate (P_{fa})	1e-8 per unit of detection volume (m^3)
Angular resolution	0.00014°

Table 3. Demo output parameters

Field	Description
Name	Full ship name (i.e., Ship_1)
LineageID	The trackers name for a TrackID
Time	Timestamp in seconds
PositionX	ECEF (Earth Centered Earth-Fixed) X coordinate
PositionY	ECEF Y coordinate
PositionZ	ECEF Z coordinate
PositionLAT	Latitude in degrees
PositionLON	Longitude in degrees
PositionALT	Altitude in meters
Course	Degrees of heading from north
VelocityMAG	Magnitude of velocity

We have demonstrated the Neptune sensor planning strategy in a CLCSim simulation. Our input to CLCSim comprised 57 modified AIS ship trajectories in the South China Sea. These ship targets were then tracked by EO/IR satellite sensors whose parameters were modeled after typical commercial satellite specifications (Table 2) from companies such as Planet or Capella. These parameters dictate the nature of the measurements processed by the tracker and influence track quality.

The output of our demonstration was then target-quality tracks on these ship targets. The track information required for the relevant dark ship analysis is shown in Table 3.

Following is a more detailed description of our demonstration. We assessed the baseline performance of track initialization, maintenance of track custody, and track volume for a constellation mimicking that of Capella, so that hypothetical performance could be assessed with high fidelity. Using CLCSim, we tracked 57 ships on the South China Sea using a constellation of 36 satellite sensor platforms spanning 3 low Earth orbit orbits of 12 satellites each. Ship measurements feeding the simulation came from available reported AIS data. One of the primary goals of the simulation was to determine whether the CLCSim + PEMT (collectively called CLCISR) software tools could maintain track custody among ships that come into close encounters with each other, as may happen with suspicious activity. The sensor platforms used an AngleSensorModel measurement type where the measurement state was composed of only two values, a horizontal angle and a vertical angle. The scan update rate was set to 1 Hz to represent a hypothetical EO/IR sensor, and all ships within the sensor's FOR were accepted into the detection gate for purposes of assessing tracker performance and generating track histories for PoL algorithm development. In the next simulation update, we will assert a more realistic, narrow field of view (FOV) for detection gating paired with the PoL and game theory modules providing sensor planning (aim-point scheduling) functionality.

We recorded a demo video of the tracker's performance via SIMDIS, a simulation display tool set.[23] This video illustrated successful maintenance of track custody between two ships that came into a close encounter, slowed to a possible stop, then departed from one another in a possible example of suspicious activity. The video collage (see Figure 3) consists of a macro view of the overall South China Sea (Figure 3a), which

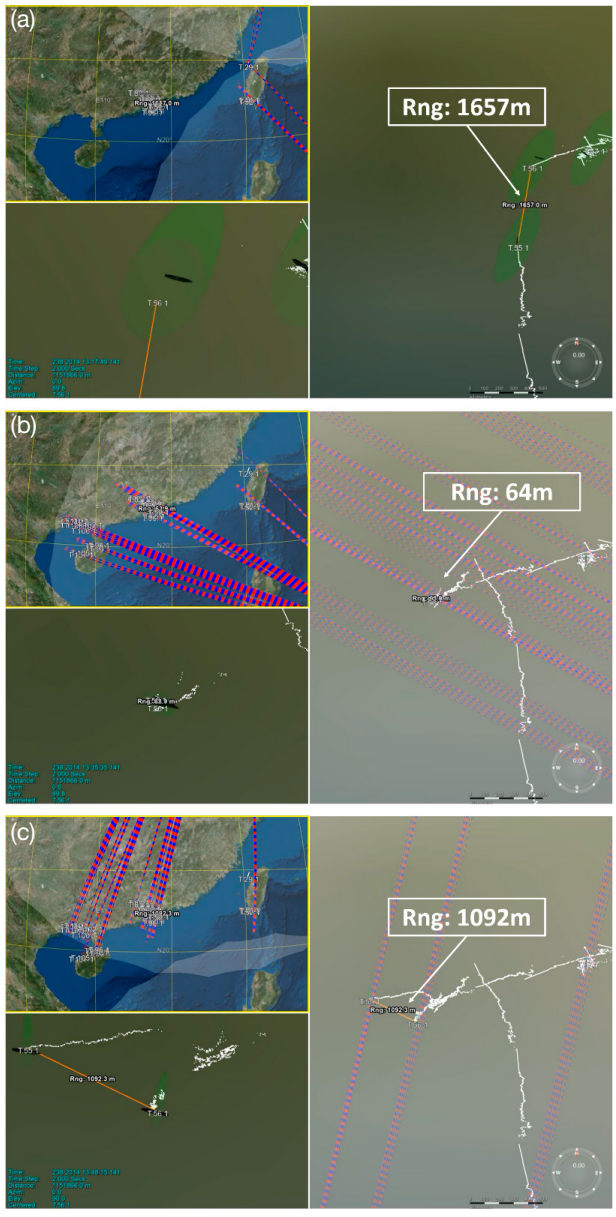

Figure 3. Year 1 demo video screenshots. (a) Tracks 55 and 56 begin to converge paths. (b) Tracks 55 and 56 nearly converge and slow to a crawl less than 50 m apart. (c) Tracks 55 and 56 diverge and separate, with track 55 continuing west and track 56 turning around and proceeding in the opposite direction.

enables the visualization of the sensor FOV as it sweeps across the scene and its relation to the location of tracks on the surface. It also shows a close-up view of a ship bound to track 56 (Figure 3b) and a static, slightly zoomed-out view of the local scene where the ships bound to tracks 56 and 55 interact (Figure 3c). The green ellipsoids represent the track estimates, centered on the track position and sized by the semimajor/minor axes of the covariance ellipsoid. The white traces represent the history of track estimates (where the tracks have been) and make it easier to see how separate track paths relate to each other.

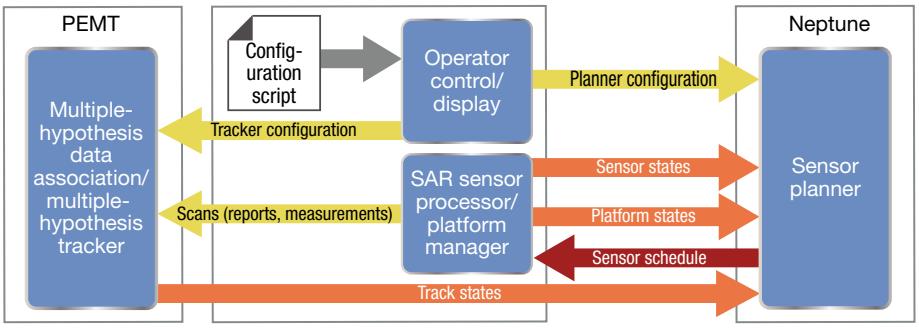

Figure 4. Data flow for Neptune SRM. The SAR sensor processor feeds the multiple-hypothesis tracker (MHT) with scans containing ship reports/measurements. The Neptune sensor planner collects track states from the tracker and sensor/platform states from the SAR sensor platform. The Neptune sensor planner then sends a sensor schedule to the sensor platform, instructing it where to aim next.

Initially, track 56 comes in from the east, while track 55 enters from the south. They cross paths at one point, after which sensor coverage briefly drops. Upon the arrival of the next sensor pass, the tracks reassociate correctly and continue updating. The tracks slowly converge as they decrease in speed until they seem to stop completely at a spacing that is barely discernible (less than 50 m; see Figure 3b). After this stop (for approximately 1 or 2 min), they proceed away from each other with 56 continuing west and 55 changing course to the south/southeast, potentially returning to its original location (see Figure 3c).

This demo shows that the PEMT tracker is capable of maintaining track custody on targets that interact very closely and with tracks that converge in state values. If the sensors were previously cued onto these ships, the tracker would have no problem maintaining them even if they were to go dark. The horizontal and vertical angle measurement uncertainties were set to 50 m, which may or may not be enough to account for uncertainty in sensor orientation and position. We are currently exploring uncertainties that result from ownship errors and image resolution. Incorporating these errors would enable the simulation to more closely estimate the performance of dark ship detection and tracking of dark ships using spaceborne sensors/platforms.

Sensor Resource Management in CLCSim

CLCISR performs sensor resource management (SRM) functions within the ESRM (extensible SRM) module, which is built as a dependency to CLCSim. SRM pertains to the task of managing sensor resources such that the quality of sensor measurements is maximized, especially when the number of target clusters to monitor exceeds the number of sensor resources available. The interface between CLCSim and ESRM is well characterized; thus, the integration of the Neptune sensor planner module is straightforward. The interface provides the following information to a sensor planner module, as shown in Figure 4:

- Track information (kinematic states [position, velocity], classification, covariance, score, probability)
- Platform information (kinematic states, orientation)
- Sensor information (orientation, FOV, FOR, measurement uncertainty, P_D, P_{fa})
- Sensor planner configuration (cost function threshold, event horizon period, sensor update period)

Track, platform, and sensor information is needed to provide the sensor planner with situational awareness required to calculate a cost function that it can optimize against (i.e., to find the "best" pairing between sensors and tracks given constraints). Cost functions are determined based on the objectives of the planner. Often, it is preferable to have a planning model that minimizes the global uncertainty of track states, in which case it would be beneficial to cue sensors to tracks of highest state uncertainty. In other cases, it would be preferable to invest sensor resources in more crowded target regions with lower track hypothesis scores (less certain track custody). Sensor planners are configurable by the user from the CLCSim scenario configuration. Once the sensor plan is made, it is packaged as a sensor schedule containing tipping updates/instructions for the sensors.

A potential scenario that would leverage SRM would simulate AIS measurements from a new sensor model that mimics AIS positional data from a maritime vessel. CLCSim's sensor models have parameters to toggle visibility of a platform (ship) that we could leverage to simulate a ship going dark. This visibility parameter would turn on for the EO/IR/SAR platforms once they go dark, which would require the sensor planner to cue the FOV

to the locations of the dark ship's (fading) track to maintain custody.

The concept of tipping and cueing in the context of our problem refers to sharing track information about a particular target or set of targets among several sensors and then determining a coordinated sensor scheduling action on that target or set of targets. The Neptune sensor planner accepts track data and relevancy scores from CLCSim and PoL (as shown in Figure 1) and uses those data with a game theory algorithm to allow for coordination between satellite sensors.

Sensor Coordination

While the PoL algorithms provide a way to judge the value of a target, there remains a need to coordinate Neptune's collection of sensor platforms. Effective coordination and cooperation among the platforms (and even between sensors on a platform) can take advantage of the information being generated by Neptune's PoL component. Furthermore, if the coordination can be orchestrated in a decentralized manner, the reliance on a central planner is relieved, resulting in a network of sensors that is not rendered useless by the failure of that single, critically important sensor platform. Numerous coordination strategies are described in the multiagent/multirobot literature. Game theory is an established framework for such strategic coordination.[23] This type of coordination is different from the usual kinematic coordination, such as formation control, which is prevalent in the multiagent community. A typical game consists of a set of players, a set of actions each player can take, and individual utilities. The players (or agents) in Neptune are the individual satellites in the constellation. Similar to a typical satellite, each player is assumed to have limited communication, information, and implementation requirements. A global objective is identified from the system-planner point of view. Players are rational agents, interested only in maximizing their own individual utilities. Despite that players behave in a self-interested manner, game theory provides useful convergence guarantees. Before we can delve into that, we need to briefly work through some notations.

There are quite a few convergence concepts in game theory, but the most well known is the Nash equilibrium. Agents in a Nash equilibrium are in a configuration such that there is "no unilateral incentive to deviate."[24] In other words, an action profile is a Nash equilibrium; if asked one by one, no agent would be better off by changing its action, given what everyone else is doing. This feature, of agents agreeing to an arrangement of actions, is desirable considering the type of multiagent system being proposed to tackle the problem of dark ship detection: autonomous and consisting of selfish agents acting in a decentralized manner.

This equilibrium may or may not be optimal—that depends on how we design the individual utilities. There is a notion of "alignment" between the global objective and individual utilities, and Arslan, Marden, and Shamma[24] detail how to pick individual utilities to maximize a global objective. They describe the importance of individual utilities by using a simple example of two autonomous vehicles trying to assign two targets between themselves. There is a high-value target (value of 10) and a low-value target (value of 2). The players (player set $\boldsymbol{P} = \{V_1, V_2\}$) choose a target to engage, which signifies their action. The agents can also choose to do nothing (T_0 in Figure 5). The matrix form shown in Figure 5 allows for a simple representation when a game consists of two players and there is a finite set of actions. For the game on the left, the payoffs to each agent are split evenly when they both engage the same target. This utility model is referred to as the equally shared utility. For the game on the right, individual agents are implementing the wonderful life utility model. In the wonderful life utility, the payoff to a player is the marginal contribution of that player's engagement with a target. Arslan, Marden, and Shamma[24] detail how the choice of individual utility functions affects the multiagent system's ability to maximize a global objective. For the game on the left in Figure 5, when both vehicles choose to engage the high-value target, we are in the (T_H, T_H) cell of the matrix and the payoffs to each player is 5 (the first number in the cell goes to the first vehicle, the second number to the other vehicle). The circles indicate the "best response" of a given player. For example, the best thing for V_1 to do when V_2 does nothing is to engage the high-value target. Hence the 10 in (10, 0) is circled. The Nash equilibrium for this game is both players engaging the high-value target, which yields a value of 5 for each player and a combined value of 10, if we consider the global objective to be the sum of the target values covered. The tweak from the equally shared utility model to

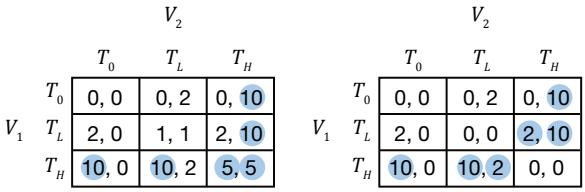

Figure 5. Two uncrewed vehicles, V_1 and V_2, are playing a game. Each vehicle can select an action from the set $\{T_0, T_L, T_H\}$.T_L refers to a vehicle choosing to engage the low-value target; T_H refers to the high-value target; and T_0 refers to neither. The game on the left uses the equally shared utility versus the one on the right, which uses the wonderful life utility. The Nash equilibria for the game on the right also maximize the global objective. This example was presented by Arslan, Marden, and Shamma,[24] and tying back to the problem of dark ship detection, a vehicle can be thought of as a stand-in for a sensor platform.

the wonderful life utility model leads to two Nash equilibria, both maximizing the global objective.

The remaining piece to design for Neptune is the way in which players choose actions. So far, we have reasoned through what a Nash equilibrium is, but how do we arrive there? It turns out, for a certain class of games known as potential games,[24] an agent can use negotiation mechanisms (also known as learning algorithms in the game-theoretic literature) to pick its actions. In a potential game, an improvement in any agent's individual utility that takes place from the agent's changing its action corresponds to an equal improvement in some global potential function that is not specific to any agent. In traffic settings, congestion serves as the potential function, for instance. There are numerous algorithms in the multiagent game-theoretic literature, and the one we implemented first is spatial adaptive play (SAP).[24,25] In contrast to other learning algorithms that require agents to maintain their own history of actions, such as regret monitoring,[24,26] SAP operates on the most recent set of assignments. Because an agent is required to maintain less information throughout the course of a game, lessening computing resource utilization and memory footprint, SAP is an attractive choice for Neptune. A detailed description of SAP is provided in Box 1. Another advantage of SAP is that with high probability, players are guaranteed to converge to a Nash equilibrium—one that maximizes a global objective given by the sum of the target values, given that the wonderful life utility is used.

Agents maintain beliefs and can take actions (based on the CLCSim environment). The key design choices are the individual (or local) utility functions and the negotiation mechanisms. We assign local utility functions in a way that allows agents to act rationally while maximizing global utility. We then pick negotiation mechanisms over local interactions that lead to agreeable action profiles—in particular, actions on ships that we consider dark targets. A significant benefit of this game theory approach for tipping and cueing is that it can deliver convergence guarantees (with high probability) to maximum or near-optimal global utility values, whereas such guarantees cannot be given by heuristics like greedy algorithms.

PATTERN OF LIFE

The final piece of Neptune assigns priority values to ships based on their location and kinematic history using an anomaly detection technique. The Neptune PoL bridges the gap between the on-orbit detection and SRM of Neptune components.

Orbit-based ship detection is a direct response to dark ships that hide their locations. However, darkness and ill intentions are not equivalent. While darkness is arguably a red flag, faulty equipment, poor reception, and signal interference can result in accidental darkness. Another consideration is that bad actors can operate without going dark or while still complying with regulations. For example, crew members aboard the MSC *Gayane* were caught smuggling narcotics in 2019 under the cover of a legitimate voyage that transmitted AIS.[27] Drug smugglers could also use small vessels that are not required to transmit AIS. The main point is that while a detection capability does respond to the dark ship problem, there is additional benefit if an algorithm can use these detections to flag anomalous ship behavior, whether from a dark ship or not.

A capability that tags ships with anomaly scores is immediately applicable to the sensor management strategies described in the previous section. The scores can be used to influence satellite coordination and to direct targeting. As previously mentioned, several commercial space companies have provided data streams to which anomaly detection can be applied; however, no such company has yet to offer such anomaly detection as part of its product suite. If we consider the preceding statement to reflect the general state of commercial industry, then our work advances the state of the art.

BOX 1. SPATIAL ADAPTIVE PLAY

In the spatial adaptive play (SAP) algorithm, described by Arslan, Marden, and Shamma,[24] each player probabilistically picks an action. At every iteration, a player can use the SAP algorithm to assign a probability to its available actions. Each player then selects an action based on the distribution resulting from SAP. Staying consistent with the original notation presented by the authors, the probability distribution over actions for player i is denoted as $p_i(k)$ at step k. If the available actions of player i (from the set $\boldsymbol{A}_i$) are listed as $\alpha_i^1, \alpha_i^2, \ldots, \alpha_i^{|\boldsymbol{A}_i|}$ where $|\boldsymbol{A}_i|$ is the cardinality of the set $\boldsymbol{A}_i$, then $p_i(k)$ is given by

$$p_i(k) = \sigma\left(\frac{1}{\tau}\begin{bmatrix} U_{V_i}(\alpha_i^1,\ a_{-i}(k-1)) \\ U_{V_i}(\alpha_i^2,\ a_{-i}(k-1)) \\ \vdots \\ U_{V_i}(\alpha_i^{|\boldsymbol{A}_i|},\ a_{-i}(k-1)) \end{bmatrix}\right),$$

where, $\tau > 0$ is a small constant and $\sigma(\cdot)$ is the soft-max or logit function. [The soft-max function takes a vector $x = [x_1,\ldots,x_n]$ and produces a vector whose ith entry is $e^{x_i}/(e^{x_1} + \cdots + e^{x_n})$.] A vector is maintained by each player. Each entry in the vector is the utility a player would receive and is derived from playing a particular action against the most recent action taken by the other players (i.e., step $k - 1$). This vector is passed through a soft-max function to constrain the entries to values between 0 and 1 and produce a distribution, which is then used to probabilistically pick an action at step.

Variational Autoencoders

Neptune PoL is currently based on running ship tracks (detection time series for a single ship) through recurrent variational autoencoders (VAE), an unsupervised deep learning sequence model. The recurrent VAE technique has its roots in Bayesian statistics and variational inference.[28,29] A core theme in Bayesian statistics is the usefulness of defining plausible probability models that emulate the real-world processes that generated a data set. The data set is then used to infer distributions over latent variables, quantities that have meaning with respect to the model but are not part of the measured data set. A common concern with Bayesian methods is that key distributions are often analytically intractable. Traditional approaches such as variational inference or Markov-chain Monte Carlo sampling parameterize approximations of these distributions. Kingma and Welling[29] and Chung et al.[30] demonstrate that neural networks can be

BOX 2. VARIATIONAL DEEP LEARNING

Variational deep learning combines the theory behind variational inference and the function approximation qualities of neural networks. Ultimately, this results in a technique that is similar to variational inference but does not need to rely on extra assumptions. The core concept in variational inference is that the marginal distribution with respect to x can be broken down into the sum of a quantity termed the evidence lower bound (ELBO) and the KL divergence between a distribution q and the posterior p:

$$\log p(x;\theta) = log\frac{p(x,z;\theta)}{q(z|x,\theta;\phi)} + log\frac{q(z|x,\theta;\phi)}{p(z|x;\theta)} = E_{q(z|x,\theta;\phi)}[\log p(x;\ \theta)]$$

$$E_{q(z|x,\theta;\phi)}[\log p(x;\ \theta)] = E_{q(z|x,\theta;\phi)}\left[log\frac{p(x,z;\theta)}{q(z|x,\theta;\phi)}\right] + KL(q(z|x,\theta;\phi)||p(z|x;\theta)) \ .$$

The marginal distribution is constant with respect to $q(.)$ parameters ϕ. Thus, the update rule below indirectly minimizes the KL divergence between q and p:

$$\phi^* = argmax_\phi\ E_{q(z|x,\theta;\phi)}\left[log\frac{p(x,z;\theta)}{q(z|x,\theta;\phi)}\right].$$

Kingma and Welling[29] demonstrated that a reformulation of ELBO is compatible with stochastic gradient methods such as those used to train neural networks:

$$\begin{aligned} E_{q(z|x,\theta;\phi)}\left[log\frac{p(x,z;\theta)}{q(z|x,\theta;\phi)}\right] &= E_{q(z|x,\theta;\phi)}[\log p(x|z;\theta)] + E_{q(z|x,\theta;\phi)}[\log\frac{p(z;\theta)}{q(z|x,\theta;\phi)}] \\ &= E_{q(z|x,\theta;\phi)}[\log p(x|z;\theta)] - KL(p(z;\theta)||q(z|x,\theta;\phi)) \\ &\approx \frac{1}{N}\sum_n \log p(x_n|z_n;\theta) - KL(p(z;\theta)||q(z|x,\theta;\phi)) \end{aligned} \ .$$

This formulation is compatible with neural networks. An encoder network computes $q(z|x,\theta;\phi)$ given measurements x. A decoder network then estimates $E_{q(z|x,\theta;\phi)}[\log p(z|x;\theta)]$ from samples drawn from $q(z|x,\theta;\phi)$. The encoder and decoder are then optimized against the reformulated ELBO objective above. Networks constructed and trained in this way are referred to as variational autoencoders (VAE).

Literature[30] extended the VAE-ELBO formulation of Kingma and Welling[29] to support sequential data as well, by conditioning on previous values in the sequence. This final formulation allows VAE to be used with recurrent neural network architectures. The hybrid architectures are appropriately called variational recurrent neural networks:

$$\begin{aligned} & E_{q(z|x,\theta;\phi)}\left[log\frac{p(x_t,z_t|h_{t-1};\theta)}{q(z_t|x_t,h_{t-1},\theta;\phi)}\right] \\ \approx \frac{1}{N}\sum_n & \log p(x_{t,n}|z_{t,n},h_{t-1,n};\theta) + KL(p(z_t;\theta)||q(z_t|x_t,\theta;\phi)) \\ & where\ h_{t,n} = f(h_{t-1,n},x_t,z_t) \end{aligned} \ .$$

The dependency on previous measurements and latent variables is implemented through a state variable h. This technique is common and well known in recurrent neural network literature. Furthermore, this construction is necessarily non-Markovian as the value of $h_{t,n}$ is influenced by all previous x and z values.

configured to do the same. The claim is that neural networks, as flexible nonlinear function approximators, may arrive at a more accurate approximation. Box 2 presents equations critical to our PoL approach.

In terms of ships' tracks, the Neptune PoL model learns in theory how to encode and decode latent track representations. Distributional constraints[29,30] are softly imposed on the latent representation while requiring it to retain enough information to reconstruct the original track. A perfect reconstruction means that decoding a latent track representation will produce a track that is identical to the original track. This can be quantified using measures such as pointwise cross-entropy or any other reasonable metric. Tracks that the model struggles to encode and decode can be considered anomalies as the model should have learned to encode and decode a wide variety of "normal" tracks during training.

Training Details and Experimental Results

We extracted tracks from data from MarineTraffic,[31] an organization that curates AIS. To reflect the goal of modeling normal kinematics, we considered only tracks that had moved 10 nautical miles from their original position, had reported AIS for at least 90 min, and had transmitted at least once every 10 min. The experiment also focused on tracks in the South China Sea. Under these conditions, 2,423 tracks were available for training on March 15, 2019, and 2,974 tracks were available for validation on March 16, 2019. We then augmented the validation set with 881 randomly generated synthetic loiter tracks as anomalous behavior targets. The synthetic ships were scattered in locations of normal traffic. All tracks were linearly resampled to a 5-min sample rate. Empirically, Neptune PoL struggles to faithfully reconstruct tracks (see the preceding section); however, the amount of reconstruction error still seems to be discriminative in nature.

We evaluated the trained PoL model against two discriminative tasks by thresholding the median reconstruction error. We present the main results as receiver operating characteristic (ROC) curves. A chance line is plotted for reference and characterizes performance for a detector that makes random guesses. We show that our approach can pick out anomalous tracks without overfitting to the training data.

The first task was to discriminate between the training and validation sets. This task checks for the degree of model overfitting. The model should not score "normal" tracks in the validation set differently from those in the training set. Ideally the training and validation distributions over reconstruction error are identical, which would result in a ROC curve that matches the chance line. The second task was to discriminate between the validation tracks and synthetic loiter tracks. For this task, the higher the ROC curve is above the chance line, the better, as this signifies that the model can pick out anomalies. As shown in Figure 6, our approach demonstrates capability to detect anomalies but has not overfit.

In addition, Figure 6 presents a number of example tracks from the extended validation set for one of the trafficked regions in the South China Sea.

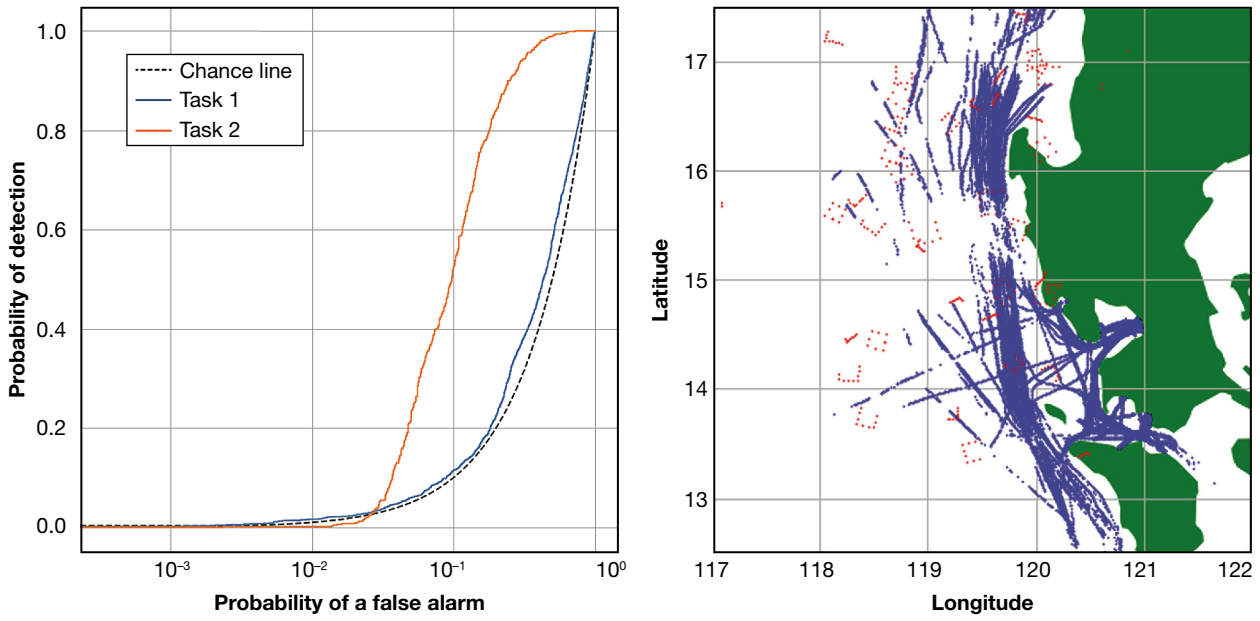

Figure 6. Loiter track flagging. Shown are ROC curves associated with two discriminative tasks, as well as a number of example tracks from the extended validation set for one of the trafficked regions in the South China Sea. The first task was to discriminate between the training and validation set to check for the degree of model overfitting. The second task was to discriminate between the validation tracks and synthetic loiter tracks.

CONCLUSIONS

The downselection and filtering of raw data—from on-sensor data returns, to fused target tracks, to PoL contextual labeling—gives Neptune the opportunity to quickly and reliably locate and identify dark ships anywhere in the world. Bolstered by APL's position as a trusted agent for the government, Neptune has the ability to enable the Lab's sponsors to augment national tactical means with the large number of commercial satellites being launched by a number of companies to achieve situational awareness across the world's oceans. It will be impossible for the government to own sufficient assets to achieve the global coverage necessary to target dark ships anywhere in the world. Therefore, Neptune's approach of leveraging commercial capabilities, in conjunction with sensor fusion and PoL analytics, will help the government solve this critical challenge. Success in this domain potentially motivates Neptune applications to other domains such as pedestrian or ground transportation automated target recognition (ATR) and behavior modeling.

From the prototype Neptune system currently in place, we want to address several key next steps. First, the on-orbit target recognition work has thus far been conducted with a processor-in-the-loop test bench, but the actual models analyzed have yet to be flown, either in space or an appropriate surrogate (e.g., a high-altitude balloon). Demonstrating a full flight article will ensure that the entire onboard ATR approach is not only feasible, as we have begun to show here, but also efficient and usable. A second step will be to incorporate a wider variety of sensor modalities, beyond the AIS transmissions and SAR imagery we have used to date. Finally, the demonstration of the PoL analysis in detecting an example known dark ship will be necessary to demonstrate the full capability of the PoL analysis.

ACKNOWLEDGMENTS: We are grateful to Erik Johnson, Glenn Mitzel, Andy Newman, Benjamin Frazier, Craig Carmen, Conrad Grant, Catherine Colangelo, Jason Fayer, and the entire APL Propulsion Grant team for their support. We are also grateful to Jon Ward, David Jansing, and Erin Richardson for their comments and feedback.

REFERENCES

[1]S. McFarlane and B. Faucon, "Ships exporting Iranian oil go dark, raising sanctions red flags," *Wall Street Journal*, Jul. 6, 2017, https://www.wsj.com/articles/ships-exporting-iranian-oil-go-dark-raising-sanctions-red-flags-1499333402.

[2]J. Crandall, "MSC Gayane crew member pleads guilty to cocaine trafficking stemming from one of the largest drug seizures in U.S. history," press release, US Department of Justice, Washington, DC, Jun. 15, 2020, https://www.justice.gov/usao-edpa/pr/msc-gayane-crew-member-pleads-guilty-cocaine-trafficking-stemming-one-largest-drug.

[3]D. Long, "China named in ambitious new anti-illegal fishing strategy for the U.S. Coast Guard," *Global Security*, Sep. 18, 2020, https://www.globalsecurity.org/military/library/news/2020/09/mil-200918-rfa01.htm.

[4]International Maritime Organization. *SOLAS, 1974: International Convention for the Safety of Life at Sea, 1974*. London: International Maritime Organization, 1974.

[5]Global Fishing Watch. https://globalfishingwatch.org/ (accessed Sep. 10, 2021).

[6]HawkEye 360. https://www.he360.com/ (accessed Sep. 10, 2021).

[7]ICEYE. https://www.iceye.com/ (accessed Sep. 10, 2021).

[8]Spire. https://spire.com/ (accessed Sep. 10, 2021).

[9]Planet. https://www.planet.com (accessed Sep. 10, 2021).

[10]Capella Space. https://www.capellaspace.com/ (accessed Sep. 10, 2021).

[11]Capella Space Corporation, "Application for new or modified radio station under part 5 of FCC rules - experimental radio service (other than broadcast), Appendix A: Response to question 7 of FCC form 442 (purpose of experiment)," WJ2XJE, Feb. 9, 2020, FCC.gov, https://apps.fcc.gov/els/GetAtt.html?id=247843&x=.

[12]"SERPENT." University of Texas at Austin. https://sites.utexas.edu/tsl/serpent/ (accessed Sep. 10, 2021).

[13]C. Adams, A. Spain, J. Parker, M. Hevert, J. Roach, and D. Cotton, "Towards an integrated GPU accelerated SoC as a flight computer for small satellites," in *Proc. 2019 IEEE Aerosp. Conf.*, pp. 1–7, https://doi.org/10.1109/aero.2019.8741765.

[14]A. J. Newman and J. T. DeSena, "Closed-loop collaborative intelligence, surveillance, and reconnaissance resource management," *Johns Hopkins APL Tech. Dig.*, vol. 31, no. 3, pp. 183–214, 2013, https://www.jhuapl.edu/Content/techdigest/pdf/V31-N03/31-03-Newman-DeSena.pdf.

[15]H. Jones, "The recent large reduction in space launch cost," in *Proc. 48th Int. Conf. Environ. Syst.*, Albuquerque, NM, Jul. 8–12, 2018, ICES-2018-81.

[16]A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017, https://doi.org/10.1145/3065386.

[17]T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020, https://doi.org/10.1109/TPAMI.2018.2858826.

[18]Y. Zhou, Z. Cai, Y. Zhu, and J. Yan, "Automatic ship detection in SAR Image based on multi-scale faster R-CNN," *J Phys.: Conf. Ser.*, 1550, 042006, 2020, https://doi.org/10.1088/1742-6596/1550/4/042006.

[19]Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "A SAR dataset of ship detection for deep learning under complex backgrounds," *Remote Sens.*, vol. 11, no. 7, 765, 2019, https://doi.org/10.3390/rs11070765.

[20]H. Kaiming, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 *IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[21]A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, et al., "PyTorch: An Imperative style, high-performance deep learning library," in *33rd Conf. Neural Inf. Process. Syst. (NeurIPS 2019)*, Vancouver, Canada, 2019, https://papers.nips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

[22]M. Zhu, "Recall, precision and average precision," Working Paper 2004-09, Department of Statistics & Actuarial Science, University of Waterloo, 2004.

[23]SIMDIS. US Naval Research Laboratory. https://simdis.nrl.navy.mil/index.aspx (accessed Sep. 10, 2021).

[24]G. Arslan, J. R. Marden, and J. S. Shamma, "Autonomous vehicle-target assignment: A game theoretical formulation," *J. Dyn. Syst., Meas. Control*, vol. 129, no. 5, pp. 584–596, 2007, https://doi.org/10.1115/1.2766722.

[25]D. Monderer and L. S. Shapley, "Potential games," *Games Econ. Behav.*, vol. 14, no. 1, pp. 124–143, 1996, https://doi.org/10.1006/game.1996.0044.

[26]J. Durieu and P. Solal, "Adaptive play with spatial sampling," *Games Econ. Behav.*, vol. 43, no. 2, pp. 189–195, 2003, https://doi.org/10.1016/S0899-8256(03)00012-5.

[27]J. Crandall, "Federal government conducts unprecedented seizure of massive cargo ship after finding almost 20 tons of cocaine on board," press release, US Attorney's Office, Eastern District of Pennsylvania, Jul. 8, 2019, https://www.justice.gov/usao-edpa/pr/federal-government-conducts-unprecedented-seizure-massive-cargo-ship-after-finding.

[28]A. Greenwald and A. Jafari, "A general class of no-regret learning algorithms and game-theoretic equilibria," in *Learning Theory and Kernel Machines*, *Lecture Notes in Computer Science*, vol. 2777, B. Schölkopf and M. K. Warmuth, Eds. Berlin, Heidelberg: Springer, 2003, pp. 2–12, https://doi.org/10.1007/978-3-540-45167-9_2.
[29]D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," arXiv, submitted Dec. 20, 2013; last revised May 1, 2014, https://arxiv.org/abs/1312.6114.
[30]J. Chung, K. Kastner, L. Dinh, K. Goel, A. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," arXiv, submitted Jun. 7, 2015; last revised Apr. 6, 2016, https://arxiv.org/abs/1506.02216.
[31]"Data Services (API)." AIS API Service, MarineTraffic, www.marinetraffic.com/en/ais-api-services. (accessed Sep. 10, 2021).

Adam B. Byerly, Space Exploration Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Adam B. Byerly is in the Ground Applications Group of APL's Space Exploration Sector. He has a BS in mathematics from the University of Maryland, Baltimore County and an MS in applied mathematics from Johns Hopkins University. Adam has contributed software engineering and data analytics expertise across a portfolio of civil and national security space missions. His research interests include machine learning, high dimensional data analysis, and space systems. He served as co-principal investigator and leader of the target detection work for Neptune. His email address is adam.byerly@jhuapl.edu.

William C. Zhang, Force Projection Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

William C. Zhang is an algorithm developer in the Electromagnetic Systems Group in APL's Force Projection Sector. He has a BS in electrical engineering, an MS in electrical engineering, and an MS in applied mathematics, all from Johns Hopkins University. Will has developed and applied machine learning techniques to critical challenges pertaining to radar and passive acoustic signal detection and characterization. His research interests primarily involve modifying methods from computer-vision deep learning to suit nonconventional data domains. He serves as a task leader on the Neptune project, guiding both algorithm development and software integration tasks. He was a key contributor to Neptune's pattern-of-life model. His email address is william.zhang@jhuapl.edu.

Sesan A. Iwarere, Force Projection Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Sesan A. Iwarere is in the Electromagnetic Systems Group in APL's Force Projection Sector. He has a BS in electrical engineering and a BS in computer science from the University of Maryland, College Park, an ME in electrical engineering from the University of Florida, and a PhD in electrical engineering from the University of Wisconsin. Sesan has worked in areas including multiple hypothesis tracking algorithms, image classification for radar returns, and probabilistic algorithms for electronic warfare. His research interest areas are probability and random processes, machine learning algorithms, control systems theory, image processing, and optimization. For the Neptune project, he helped lead the data and fusion team members who integrated CLCSim (a platform simulation environment) with game theory (for sensor coordination) and pattern of life (for dark ship anomaly detection). He is a graduate instructor for the Johns Hopkins University Whiting School of Engineering and is also a member of IEEE. His email address is sesan.iwarere@jhuapl.edu.

Waseem A. Malik, Force Projection Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Waseem A. Malik is a section supervisor in the Guidance, Navigation, and Controls Group in APL's Force Projection Sector. He has a BS in electrical engineering, a BS in mathematics, an MSc in electrical engineering, and a PhD in electrical engineering, all from the University of Maryland, College Park. Waseem leads a team that works on fundamental methodologies underlying strategic systems, advanced estimation algorithms, optimization methods, control algorithms, and strategic technology development roadmaps. For the Neptune project, he helped lead the data fusion team and also led outreach efforts across the Lab. He is a member of the adjunct faculties at the Johns Hopkins Whiting School of Engineering and the A. James Clark School of Engineering at the University of Maryland, College Park. His email address is waseem.malik@jhuapl.edu.

Sheldon F. Bish, Force Projection Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Sheldon F. Bish is a software engineer in APL's Force Projection Sector. He has a BS in biomedical engineering from the University of Connecticut and a PhD in biomedical engineering from the University of Texas at Austin. Sheldon develops sensor fusion simulations of combat engagements in land, sea, air and space domains. In particular, he has developed sensor models for bistatic radar, synthetic aperture radar (SAR), and general improvements to sensor fusion software. He has also developed SAR image characterization tools and is a major contributor to a high-fidelity SAR imagery simulator. For the Neptune project, Sheldon constructed the network interfaces between CLCSim (scenario simulator), pattern-of-life (dark ship anomaly model) and game theory (sensor coordinator) modules. His task also involved system analysis of tracking performance under different configurations. His email address is sheldon.bish@jhuapl.edu.

Musad A. Haque, Space Exploration Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Musad A. Haque is in the Embedded Applications Group in APL's Space Exploration Sector. He has a BS in electrical engineering from the University of Texas at Arlington and an MS and a PhD in electrical and computer engineering from the Georgia Institute of Technology. His doctoral work focused on the coordination of multiagent systems inspired by the collective behavior of social animals. At APL, Musad primarily works on the NASA Double Asteroid Redirection Test (DART) mission. He started as a member of the SMART Nav team and currently supports DART's integration and testing, having worked on the mission's guidance, navigation, and control; autonomy; and fault management teams along the way. Musad has led a number of independent research and development efforts in artificial intelligence (AI)/machine learning, multi-agent systems, and intelligent constellations. He teaches AI at the University of Maryland, Baltimore County as an adjunct professor. His email address is musad.haque@jhuapl.edu.

Tamim I. Sookoor, Asymmetric Operations Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Tamim I. Sookoor is in the Critical Infrastructure Protection Group in APL's Asymmetric Operations Sector. He has a BE in computer engineering from Vanderbilt University and an MS and a PhD in computer science from the University of Virginia. Tamim has led a number of internally and directly funded projects including the Institute for Assured Autonomy (IAA)–funded Runtime Assurance for Distributed Intelligent Control Systems (RADICS) and the Safety Assurance-Integrated Learning with Operator-assisted Retraining (SAILOR) Propulsion Grant. His research interests include cyber-physical systems (CPS), internet of things (IoT), smart cities, and assured autonomy. He is leading the development of technologies at the frontier of secure CPS and resilient smart cities. Tamim serves as a co-principal investigator and project manager on the Neptune project. He teaches graduate courses at the Johns Hopkins University Whiting School of Engineering and is the coordinator of the Assured Autonomy track. He is a member of IEEE and ACM and the Secretary for the Baltimore Area chapter of ACM. His email address is tamim.sookoor@jhuapl.edu.