Are we alone?

Machine Learning for Detection of Life-Like Motility in Liquid Samples

Steffen Mauceri[1]*, Mark Wronkiewicz[1], Jake Lee[1], Jack Lightholder[1], Gary Doran[1], Lukas Mandrake[1]

[1] California Institute of Technology, Jet Propulsion Laboratory, Pasadena, CA, USA
*Email: Steffen.Mauceri@jpl.nasa.gov

The Search for Life on Ocean Worlds

Jupiter's moon Europa and Saturn's moon Enceladus are expected to contain a liquid water ocean beneath their icy shells [1, 2]. All life we know relies on water, making these moons especially important targets for future missions to explore. To support such a mission, JPL is working on a suite of instruments capable of searching for life in liquid water samples called the Ocean Worlds Life Surveyor (OWLS). Finding life in these oceans could finally answer the question whether we are alone in the universe.

Life Moves

One possibility to search for life in water samples is by visually inspecting them for motility: the ability to move voluntarily. Motility is important to find nutrients and respond to potentially harmful stimuli and a distinct indicator of life. One of the instruments included in OWLS is a Digital Holographic Microscope (DHM) (Figure 2a). The DHM [3] can image a liquid water volume multiple times per second at submicron resolution with a depth of field that is 50 times larger than that of a conventional microscope. This makes it suitable for finding motile microorganisms.

Data Compression

The DHM generates 80 MB/s of data, which quickly exceeds the entire downlink capability (~75 MB) for a typical mission to Europa or Enceladus. Therefore, we must compress 100s of GBs of data and only return the most scientifically salient components to Earth. To solve this challenge, we developed an autonomous tool called Holographic Examination for Life-like Motility (HELM), which finds motile particles within DHM videos and extracts their spatiotemporal paths and small image crops. This effectively reduces the amount of data by orders of magnitude that needs to be transmitted to Earth while preserving most of its information content.

Automated Detection of Motility

HELM uses a multi-step algorithm to differentiate between motile and non-motile microorganisms.

1. Starting with the sequence of raw DHM images (Figure 2b) the median is subtracted. This increases the contrast for moving objects within the sequence and removes the static background.
2. Moving objects in the images (Figure 3a) are tracked (Figure 3b).
3. Track characteristics are extracted to prepare for Machine Learning (e.g., average speed, acceleration, turn radii, etc.).
4. Each track is classified as motile or non-motile using a random forest classifier

Data Prioritization and Downlink

Since the data generated by the DHM far exceeds downlink capabilities the individual samples are sorted by their priority. The priority of each sample is calculated based on its estimated science utility (e.g. number of motile particles) and how it differs from the other analyzed samples. The compressed data products of the most promising samples will then be downlinked to Earth (Figure 4), reducing the 100s of GB of raw data down to less than 100 MB.

Future Work

Going forward we will test the full system on samples from analog sites including Mono Lake, CA and Borup Fiord Pass, Canada. Through those tests, we will demonstrate the capabilities of OWLS and assess its readiness for future space missions.

Citations:

[1] Postberg, F.; Kempf, S.; Schmidt, J.; Brilliantov, N.; Beinsen, A.; Abel, B.; Buck, U.; Srama, R. Sodium salts in E-ring ice grainsfrom an ocean below the surface of Enceladus.Nature2009,459, 1098–1101.2.

[2] Carr, M.H.; Belton, M.J.; Chapman, C.R.; Davies, M.E.; Geissler, P.; Greenberg, R.; McEwen, A.S.; Tufts, B.R.; Greeley, R.; Sullivan,R. Evidence for a subsurface ocean on Europa.Nature1998,391, 363–365.

[3] J. Kent Wallace, Stephanie Rider, Eugene Serabyn, Jonas Kühn, Kurt Liewer, Jody Deming, Gordon Showalter, Chris Lindensmith, and Jay Nadeau, "Robust, compact implementation of an off-axis digital holographic microscope," Opt. Express 23, 17367-17378 (2015)

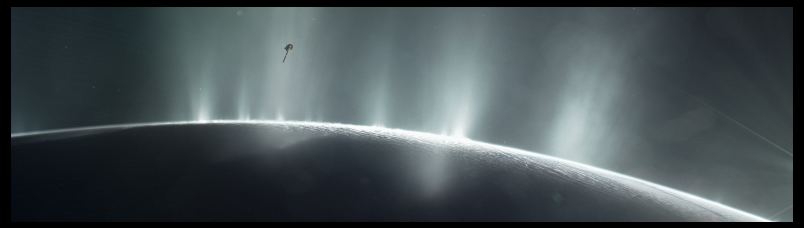


Figure 1: Artist's impression of a mission to Enceladus. A liquid water ocean is expected under its ice shell that might contain life. [image: NASA]

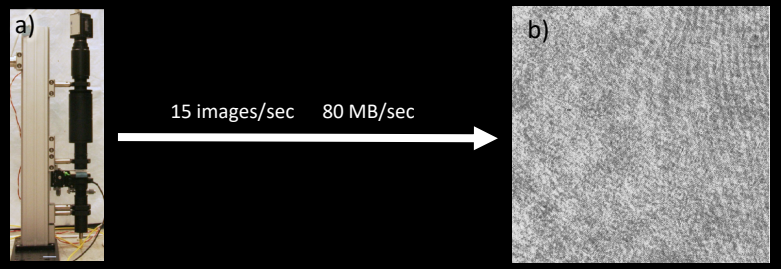


Figure 2: DHM (Digital Holographic Microscope) for visual inspection of liquid water samples: (a) DHM, (b) Raw hologram from the DHM that contains live bacteria.

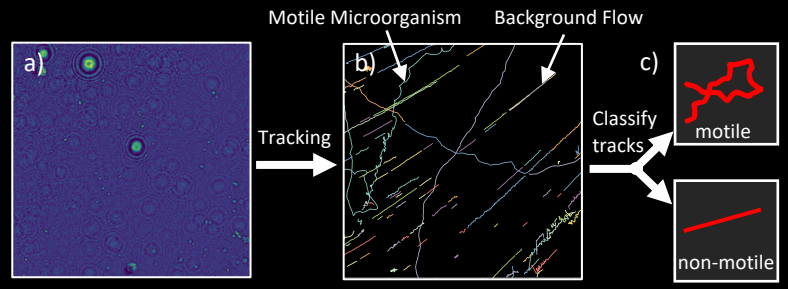


Figure 3: Tracking and classification of bacteria: (a) Hologram with background removed by median subtraction. (b) Tracks for moving objects generated by tracking algorithm. (c) Classification of tracks as motile or non-motile.

Figure 4: Sending the most life-like imagery back to Earth. HELM reduces the 100s of GB of raw DHM data down to ~75MB.

Construction Rovers

Kendall Johnson

kjohns21@masonlive.gmu.edu || Youtube: MoveOverRover ||Instagram: Moveroverrover1

George Mason University | 4400 University Drive, Fairfax, VA 22030

Goddard Space Flight Lab | 8800 Greenbelt Rd, Greenbelt, MD 20771

Abstract

In July 2019, I attended the *Lunar Operations and Technologies to Enable Human Exploration of Mars and the Moon* Workshop. It is a group of NASA scientists who believe that leveraging lunar technologies will allow us to be more successful in getting to Mars. They also believe in creating devices that can work on both celestial bodies will save time, money, and energy in our inevitable journey into the cosmos. One of the main priorities stated for helping to enable Moon and Mars exploration is specified rovers that can complete tasks involving science, maintenance, transportation, and construction (Thronson, 2019). NASA's Martian rover team has done a great job concerning a scientific rover, the classic Lunar rover can transport humans on the surface of the Moon, and currently, there is nothing to maintain on any celestial body, other than Earth, for a maintenance rover. The last of this Lunar Operations and Technologies task is a construction rover, and this is where I have started my theories and experiments to demonstrate actions, hard data, possibilities, shortcomings, and abilities of a future construction rover by building and testing a prototype.

The first task of a construction rover is to best prepare the celestial body's surface for the arrival of humans. The rover is given the task of preparing the surface that can include clearing away regolith, flattening the ground, and moving equipment and debris. This is a demonstration of the proof of concept for the construction rover as an autonomous platform using Artificial Intelligence (AI) and Machine Learning (ML) for the rover moving debris. Adding AI to the construction rover will best open the door to the system being fully autonomous. The AI of the construction rover will consist of separate trained Artificial Neural Network (ANN) for sets of different tasks. An ANN is an AI technique that uses the concept of a biological neuron to weigh parameters for predicting. The first ANN has been trained for the rover's movement based on incoming sensor data producing states for movement. The next ANN is a Convolution Neural Network (CNN) that I am using for object classification, detection, and recognition. A CNN is an ANN, but with the added benefit of a convolutional layer that can take into account the surrounding pixels along with the original pixel into the training of the neural network. The rover's CNN will be trained on images of which objects to carry and which objects to push regarding the AI platform's proposed tasks. The last ANN will be used for reinforcement learning. This ANN is not very different from the first other than it will be trained in a simulation to carry out the important task of path planning. Each ANN gives the AI platform the ability to make its own decisions from what it sees through its camera and reads through its sensors. In the demonstration of the AI construction rover, I hope to show it successfully moving an object using these three AI algorithms as one. I further hope that NASA can use the concepts of these AI algorithms as a tool for the next step of rover programs, and getting humans more opportunities on other worlds.

Objectives and Components

My objectives are to show that the construction rover can complete the tasks of obstacle avoidance, obstacle detection, path planning, and moving an object from one place to another using only its trained autonomy. Briefly speaking these tasks are much easier on Earth, but for a construction rover in space, it must be able to complete its task in a hostile environment and communicate with us humans on Earth from great distances in space. Current rovers also have the added hardship of being moderately controlled by an operator. That's why I'm proposing an autonomous rover using AI to be sent to the rocky bodies that can act in real-time not depending on a team of scientists to catch up to a machine that is 100 million miles away. Now it is easy to use a team to control a rover, but when there are hundreds in space we will need a more solid method.

The rover will be trained with many ANNs that will be used in image recognition, translation of sensor states to movements, and path planning to demonstrate autonomy level 4. Autonomy level 4 behavior is described as high autonomy where that rover is completely autonomous in most situations and conditions with an opening for a human in the loop. NASA Ames's Maria Bualat (2017) Deputy Lead of the Intelligent Robotics Group makes this statement about NASA's relationship with autonomous vehicles saying that we always want the "human in the loop because of experience and powerful cognitive ability". I will be the human component that feeds the ANN real-world and synthetic data made specifically for a rover-like construction robot to complete the tasks above. A better description of this would be, "directed autonomy," to which we can teach the rovers here on Earth then send it anywhere in the Universe, and it can take the human component with it.

Hardware

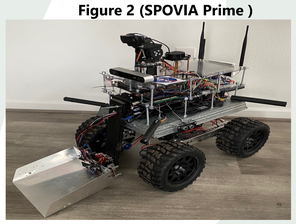
The prototype construction rover-like robot I will be using in this demonstration is a robot that I have built and named Solar Powered Operation Vehicle Intelligently Autonomous (SPOVIA) Prime shown in Figure 1. I took much inspiration from the 2004 EVA test vehicle shown in Figure 2 that was used to simulate possible futures and technology on a foreign rocky body. I used the tractor scoop idea from their demonstration and added it to SPOVIA prime. SPOVIA prime is a 35-pound solar-powered robot with four strong motors, big wheels, and 30 basic analog sensors consisting of ultrasonic, inferred, ultraviolet, photo, accelerometers, and many more. It also includes the more complex components of lidar, GPS, GSM, stereo camera, and USB color camera. The processor, which is optimal for AI tasks and training like these, is a Jetson TX2 with a Dual-Core NVIDIA Denver 2 64-Bit CPU, Quad-Core ARM® Cortex®-A57 MPCore, and 256 NVIDIA CUDA cores. Also, I will be using the google coral TPU to run the object detection model at speeds up to 40 FPS.

Software

The software I will be using is TensorFlow, PyTorch, and Sci-kit learn for building the ANNs. Pandas, Numpy, and Pymata will be used for supporting those modules in Python 3.7. The synthetic data made will contain states of all of the rover's known action possibilities. The real data collected is from tests of the robot's movements and demonstrations of its actions. The data will be normalized from 0 to 1 among the inputs and outputs and organized into a matrix that contains raw sensor data with the last column being the output state determined to act on. The image data is from the COCO dataset and pictures of rocks and smalls balls I took for the demonstration of classifying and moving an object.

Figure 1 (2004 EVA 'Tractor')

Figure 2 (SPOVIA Prime)



Methods and Preliminary Results

This section shows the algorithms chosen for the construction rover's autonomous decision making and why. Quantitative and qualitative results will be displayed some with graphs and others with pictures of use.

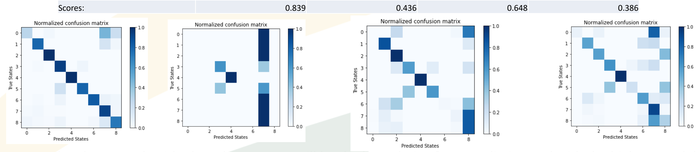
Why Artificial Neural Networks?

As a preliminary effort to choose the best model for the incoming robotic sensor data I compared ANNs with the machine learning techniques Support Vector Machine(SVM), Naive Bias, and Logistic Regression. The resulting model scores and confusion matrixes are shown in Figure 3.

Figure 3 (Model Score (80/20 test train split) and normalized confusion matrix of predicted state vs true states.)

ANN **SVM** **Naive Bias** **Logistic Regression**

Names:	ANN	SVM	NB	LR
Scores:	0.839	0.436	0.648	0.386



From the above results, it's clear that an ANN will be the best choice for a predictive tool with the highest model score and best predictive power shown in the first confusion matrix. The next step taken was to make adjustments to the ANN to get the best results and model score for the predicted robot states from the incoming sensor data so that it can best obstacle avoid. I used around 26,000 points of sensor data. Below in Figure 4 is the normalized confusion matrix of the best ANN model with a strong model score of .94.

Figure 4 (Confusion Matrix of Obstacle Avoidance ANN model).

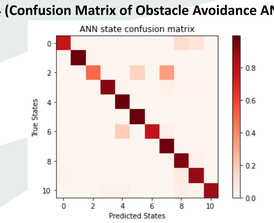
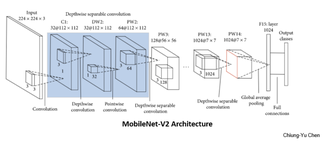


Figure 5(MobileNet SSD v2 CNN structure)



Object Detector using Convolutional Neural Networks

ANNs are not just used for determining sensor states and predicting actions to take. An ANN can also be used to train an object detector, from chosen labeled images, to detect and classify objects. Detecting and classifying gives it the ability to make decisions with the visual information from its surroundings. The ANN I used is similar to the one before except that I added convolutional layers making it a Convolutional Neural Network (CNN). CNNs have the same functions and very similar structures to ANNs, but adding convolutional layers that pool a set of data allows for condensing of important surrounding image features that one would want to be detected. To come up with this result the convolutional layer takes into account the values of close inputs this can be done by simply finding the sum or the average of the incoming inputs aka pooling. CNNs are known to work very well with data in a matrix, like an image, and this is why it is the best choice to be used.

More specifically I am training this object detector using MobileNet SSD v2 using TensorFlow's API. Its structure can be seen in Figure 5 above. I am using TensorFlow to incorporate the new technology of TPUs (Tensor Processing Unit), the google coral, to run my detection algorithm at up to 40 FPS. I have trained it on the COCO dataset plus images of rocks, balls, and other robots. The detection of rocks and balls can be seen in Figure 6. My goal was to train the robot to identify a rock or a ball and initiate a picking up procedure. It uses its scoop and the detected pixel positions to lift the object then travel to the destination of the object and place it down an example of the ball in the scoop is shown in Figure 7.

Figure 6 (Ball & Rock detection)

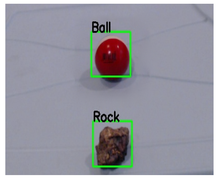


Figure 7(Ball Detection & lift in scoop)

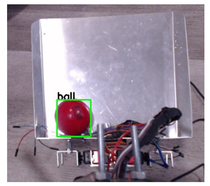
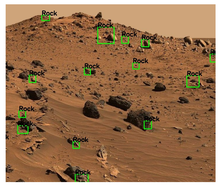


Figure 8(Rock Detection for Mars Surface).



Path Planning

Another important feature that cannot be overlooked is the path planning of the vehicle. This is also a particularly tricky task due to no GPS, roads, and very few land features which creates a very hard problem for an autonomous rover and even a human. The easy solution for a construction rover would be to map the area and create waypoints to travel to and from. I overstepped this problem for SPOVIA Prime using a GPS GUI to plot the points for the rover to travel to, but many obstacles remain especially if it is a large rock in front of the rover that can discourage a straight path. I have attempted to uses the object detection program to detect the rocks in front and possibly avoid them, but seen in Figure 8 this picture of the Martian surface, it is not the most reliable.

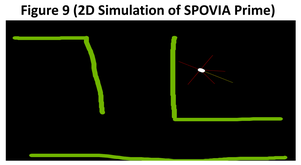
All of my work can be seen at https://www.youtube.com/channel/UCTCYvzO24Ebm68cVsQoy9Eg or MoveOverRover.

Methods and Preliminary Results cont..

Reinforcement learning (RL)

To make the rover go continually straight from one waypoint to another I am using a concept of the algorithm pure pursuit in a reinforcement learning neural network to train the algorithm to do its best to go straight when its sensors don't detect any obstacles. A reinforcement learning neural network is the same as any other ANN, but it is trained in an ongoing simulation with goals, rewards, and point values. Figure 9 is from the 2D simulation of SPOVIA Prime with pure pursuit in mind. The white oval is the rover, the red lines coming outward are the distance sensors, and the yellow line represents a point in front of the rover that is to be followed. The rover gets points based on how close the slope of its initial position and final position is to the slope of the initial position and end of the yellow line. The rover also loses points as the distance sensors (red lines) touch anything. This builds path panning into the fabric of its design allowing for easy, not computationally expensive movement.

Figure 9 (2D Simulation of SPOVIA Prime)



Discussion and Conclusion

Discussion

SPOVIA Prime was very successful at the task of obstacle avoiding using the first ANN. It was a strong rarity that it hit any other detectable objects, although I did have the vehicle travel slower than I would have wanted. SPOVIA Prime was incredibly successful at detecting the objects it was trained to. From the color camera, it was able to detect rocks, balls, people, and cars (from the COCO dataset) like the examples seen in Figures 6 and 7. But, unfortunately when there were many examples to detect the object detection algorithm failed, seen in Figure 8. The more unfortunate news is that the Martian surface from Figure 8 will be the terrain the construction rover will have to mostly deal with, being that it is meant to be sent to rocky planetary bodies.

A limitation I found is that the simulation avatar had minor differences then SPOVIA Prime, and those differences turned into a problem when moving the algorithm in the simulation to the real world. I believe this is why SPOVIA Prime's movements came out sloppy and jerky, but it did complete the task of obstacle avoidance, object detection, and path planning. I will say that I believe the RL most needs to be updated and changed, but the fault probably most falls on the 2D simulation.

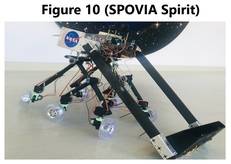
For the main task of picking up a detected object and moving it to another location, it was both a success and a failure. SPOVIA Prime was easily able to detect an object and attempt to pick it. But even when SPOVIA Prime was able to grab the object it was unable to bring it to the chosen waypoint. I ended up settling for success in the robot's ability to detect, lift, and drive away with the detected object still in the scoop.

Conclusion

As a figurative idea, the premise of using AI to make a rover autonomous perform actions is solid, but in the literal, it is much more difficult. Although each algorithm was trained well and was able to perform at its best by itself it was difficult to put together and I rarely got a successful example. The solution to this problem is to better match the algorithm with the physical vehicle. An even bigger problem was deciding which algorithm to carry out and when. I ended up making the obstacle avoidance ANN the priority and when it detected no obstacle the path planning took over followed by the object detector. The solution to this problem would be to make a good task manager for the robotic movements. This was something that I discovered during testing. The takeaways for a more successful autonomous prototype is more training and testing. I believe that in the future this concept will be successful because humanity is curious and will wonder what it is like to live on other celestial bodies, but due to the harsh environments, we will need robotic help to prosper.

Future Work

Figure 10 (SPOVIA Spirit)



I plan to continue this project and demonstrate it again. I will go from a 2D simulation to a 3D, use the object detector CenterNet instead of MobileNet, and create an ANN to be the task manager acting as a decision-maker for the SPOVIA Prime.

After, I plan to use swarm technology for robots to work together. In the object detection section, I mentioned training SPOVIA Prime to see my other robots because I want them to be able to classify each other and work together to get more difficult tasks completed. I have already been working on a partner construction rover seen in Figure 10.

Thank you for your time.

References and Acknowledgements

I would like the thank all the staff of GMU's Physic's department, GMU's The Space Weather Team, and all of the staff at NASA Goddard's Heilophysics lab.

(1) Bualat, 2017, NASA AMES Research center Youtube, https://www.youtube.com/watch?v=P9zmerD04Hk

(2) Rouen et al., (2004), EVA and Mobility Systems Engineering

(3) Saha, 2018, *A comprehensive guide to Convolutional Neural Networks*

(4) Thronson, 2019, *Findings from the Sixth Workshop on Achieving Mars Exploration*

Second Artificial Intelligence and Data Science Workshop for Earth and Space Sciences hosted by NASA Jet Propulsion Laboratory, February 2021 (virtual event)

ID25: Automating validation of satellite-derived ice-cover features: Discriminating ice objects in optical ice images with different degrees of local texture distortions

Ekaterina Kim[1,2,3] (ekaterina.kim@ntnu.no), Nabil Panchi[4], Ole-Magnus Pedersen[3], Sveinung Løset[2,3], Anirban Bhattacharyya[4]

[1]Centre for Autonomous Marine Operations and Systems (AMOS), Norway;[2]Centre for Sustainable Arctic Marine and Coastal Technology (SAMCoT), Norway; [3]Norwegian University of Science and Technology, Trondheim, Norway; [4]IIT Kharagpur, West Bengal, India

Background and motivation: The amount of data from satellite missions is expected to increase. In fact, this number increases faster than the capacity of experts to process, adequately validate and evaluate the uncertainty of the results. Despite rapid progress in machine learning, the methods and standards for automated interpretation of sea ice imagery remain underdeveloped. One such field is the automated interpretation of ice imagery from ground operations, especially under poor visibility conditions (e.g., imagery from surface vessels, shore stations, etc.). There is a strong need for robust and efficient methods enabling the automated processing of close-range sea ice imagery to aid in the derivation of useful characteristics of sea ice cover (ice types, concentration, decay).

How accurately can ice objects in close-range optical images be automatically detected and identified to supplement on-ground operations and validate satellite-derived sea-ice products?

Custom-built datasets for training, validation, and testing: 375 unique close-range optical images of ice cover (three channel RGB, jpeg), manually collected, labeled (following ice object definitions and labelling rules), and verified (with an ice expert). 14 classes: Ice objects (*level ice, deformed ice, brash ice, pancake ice, iceberg, floeberg, floebit, floe, broken ice, underwater ice*) and additional features (*shore, water, sky, melt pond*).

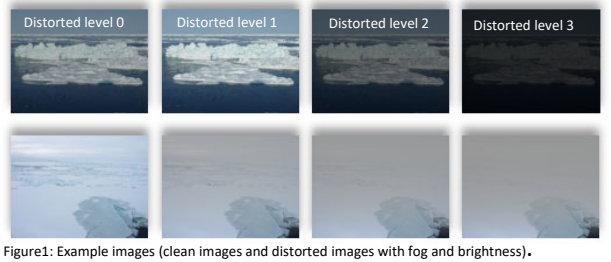
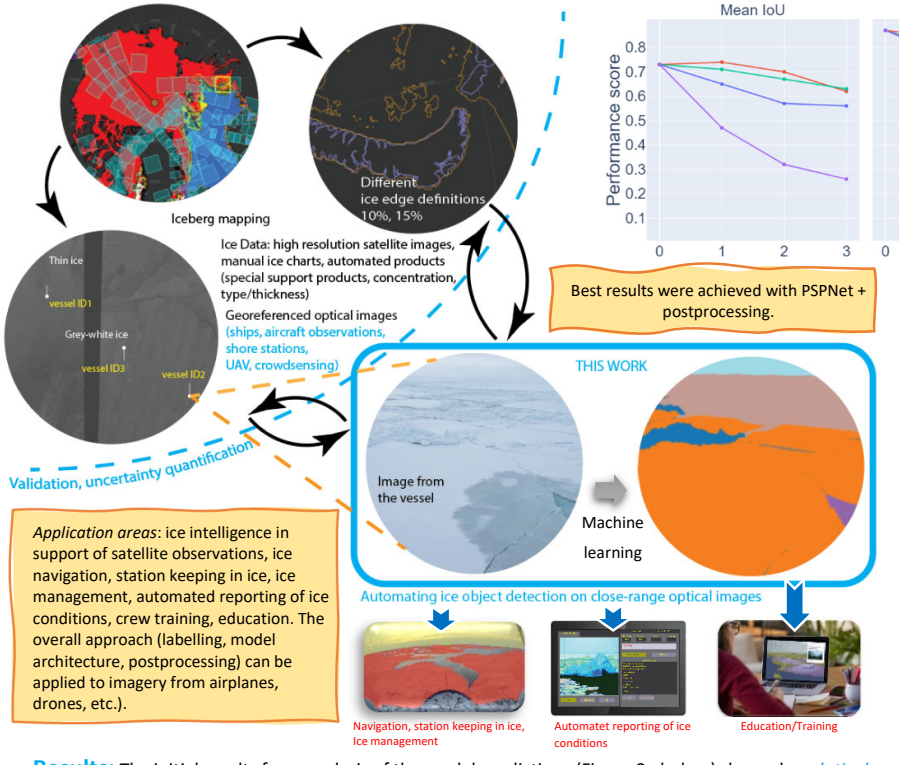


Figure1: Example images (clean images and distorted images with fog and brightness).

Model architecture: Evaluated 12 open-source neural network architectures: PSPNet, PSPDenseNet, DeepLabV3 Plus, UperNet, DUC HDC, FCN, GCN, ENet, UNet, UResNet (UNet with ResNet backbone), SegNet, SegResNet.

Adds-on: Evaluated performance on clean and distorted images (3 distortion levels for fog, dark, blur, noise), with and without postprocessing (fully connected conditional random fields and faster convolutional conditional random fields), various model ensembles, human factors, and compared model predictions with human performance.

Performance metrics: Global and class-wise Accuracy, mean Intersection over Union, Confusion Matrices, F1 score



Application areas: ice intelligence in support of satellite observations, ice navigation, station keeping in ice, ice management, automated reporting of ice conditions, crew training, education. The overall approach (labelling, model architecture, postprocessing) can be applied to imagery from airplanes, drones, etc.).

Best results were achieved with PSPNet + postprocessing.

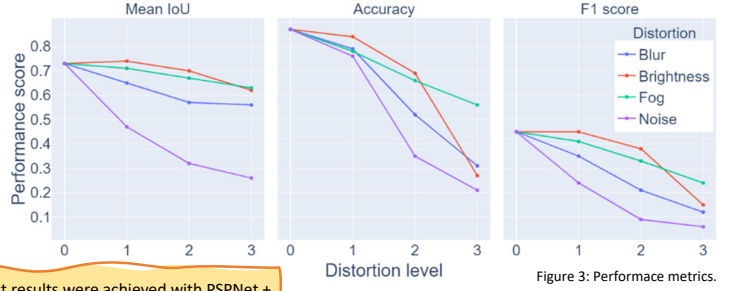


Figure 3: Performace metrics.

Results: The initial results from analysis of the model predictions (Figure 2 - below) showed *a relatively good segmentation of clean images, with the mean intersection over union (IoU) of 0.73* (fully convolutional U-Net model with the a pretrained ResNet101 as encoder and convolutional conditional random fields based post-processing of model outputs).

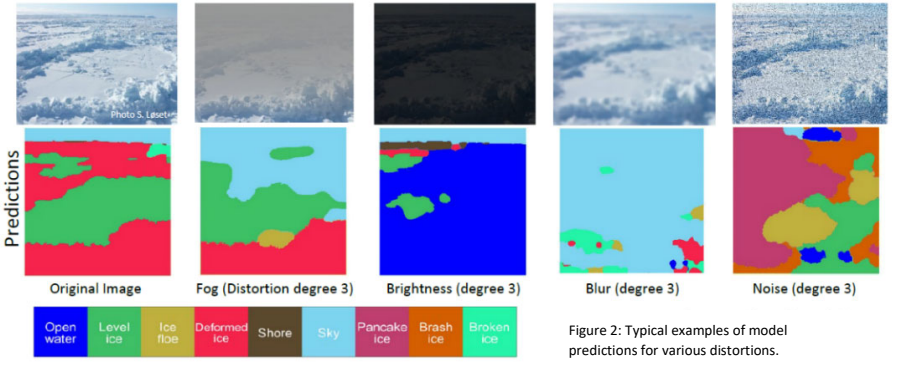


Figure 2: Typical examples of model predictions for various distortions.

We are currently working on improving model performance on distorted images, making it more robust on previously unseen visual conditions. In general, noise distortion has the most profound effect on performance (Figure 3), whereas the fog and darkness affect the predictions least, when the performance is measured in terms of mean IoU.

Remarks: This work opens a possibility for automation of ice object detection and identification from georeferenced close-range optical images (e.g., imagery from surface vessels, shore stations, etc.). The identified and georeferenced ice types that can serve as ground-truth data for validation of satellite-derived ice-cover features.

We have highlighted an area where the current model works and fails, thus giving important indications of where to direct future research efforts.

References:

Kim, E., Dahiya, G.S., Løset, S., and Skjetne, R. 2019 Can a computer see what an ice expert sees? Multilabel ice objects classification with convolutional neural networks, Res. in Eng. https://doi.org/10.1016/j.rineng.2019.100036

Panchi, N., Kim, E., Bhattacharyya, A. 2019. Understanding Polar Environment: Preliminary Results from Deep-Learning-based Segmentation of Optical Ice Images. I: Proceedings of the 6th International Conference on Ship & Offshore Technology. The Royal Institution of Naval Architects, pp. 169-179.

Pedersen, O-M., Kim, E. 2020. Arctic Vision: Using Neural Networks for Ice Object Classification and Controlling How They Fail. Journal of Marine Science and Engineering. vol. 8 (10).

Pedersen, O-M., Kim, E. 2020. Evaluating Human and Machine Performance on the Classification of Sea Ice Images. Proc. IAHR International Symposium on Ice 2020 ISBN 978-82-7598-1

A MACHINE LEARNING APPROACH TO LOW EARTH ORBIT SATELLITE HEALTH AND SAFETY TELEMETRY

Zhenping Li

ASRC Federal, 7000 Muirkirk Meadows Dr #100, Beltsville, MD 20705

Image credit: Japan Meteorological Agency

Introduction

The machine learning (ML) approach includes

- An architecture model that defines the machine learning processes and interfaces.
- A scalable and extensible enterprise architecture for a ML system that provides common infrastructure and services for ML algorithms that are treated as mission specific software components
  - implemented in Advanced Intelligent Monitor System (AIMS).
- A ML algorithm library that provides efficient and accurate data training outputs for telemetry data and instrument calibration data.

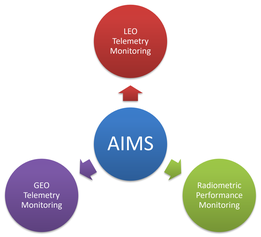


Figure 1: AIMS has been applied to GEO, LEO telemetry data monitoring and satellite instrument Radiometric performance monitoring

Satellite Telemetry Data Challenges

- **The Separation of expected data pattern changes in normal operations from unexpected data pattern changes due to anomalies is essential.**
  - Not all data pattern changes in telemetry lead to anomalies, and they also be result of normal satellite operations or changes in space environments
  - Anomaly detections without the separations of expected from unexpected data pattern changes lead to false positives

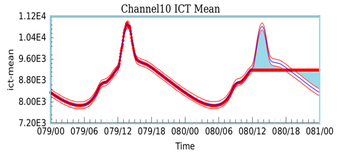


Figure 2 An anomaly example in an instrument radiometric data. The thick red line represents the actual data that became flat. The blue line and orange lines are ML model output and data bound.

- **Correlations among multiple telemetry datasets in multiple subsystems must be considered in separating normal operations from anomalies.**
  - Interactions among subsystems in a satellite lead to strong correlations among telemetry datasets.
  - Data pattern changes due to normal satellite operations or anomalies generally involves multiple telemetry datasets in multiple subsystems.

A Hybrid Approach

Detect data pattern changes through a supervised learning

- Learning normal data pattern in datasets, while detecting data pattern changes.

Separate anomalies from normal operations with an unsupervised learning

- Define a quantitative metric that measures data pattern changes
- Develop a mathematical representation to characterize correlations among datasets
- Correlation patterns determine if data pattern changes are part of normal operations or anomalies

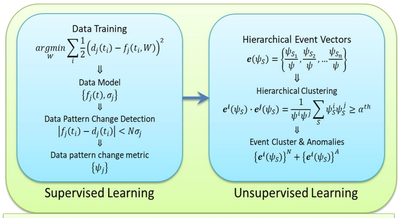


Figure 3 The hybrid approach to anomaly detections in satellite telemetry datasets.

Data Training for LEO Satellite Telemetry Data

The challenges in the data training of LEO satellite telemetry data

- More diverse data types, High complexity in data patterns, and Relationships exists among datasets

The requirements for data training in operational environments

- Efficiency: data training for a dataset should be completed in seconds or minutes instead of hours or days.
- Accuracy: essential in detecting data pattern changes in datasets.
- Robustness: the training data may contain outliers that distort data training outcome

The flexibility in selecting different data models for different patterns and noise level is critical

Figure 4 The data training output for the current and voltage in the power system. The red dots are the actual data and blue lines are the data training outputs. The datasets, such as the one in the power subsystem or the satellite temperature profiles, have typical data patterns that follow the satellite orbital behavior with the same pattern period as the orbital period.

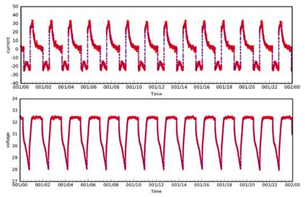


Figure 5 The data training output for the quaternions that determine the satellite attitude. The data patterns for the quaternions follows the satellite orbit period. The quaternions are generally not continuous. Neural networks are generally implemented as the data model for quaternions. Detecting data pattern changes in quaternion is critical in capture the signature of satellite maneuver.

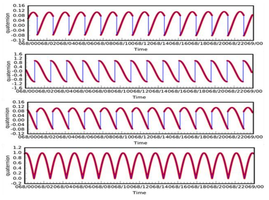


Figure 6 The data training output for the momentum profile of the reaction wheel. The red dots are the actual data and blue lines are the data training outputs. The data patterns of reaction wheel don't follow orbital patterns and highly complex. The Fourier expansion model is implemented.

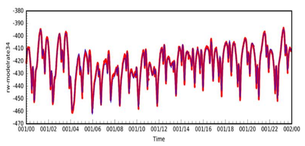
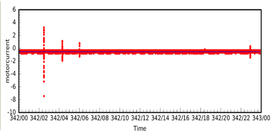


Figure 7 The data training output for the motor-current in the reaction wheel subsystem. There are significant number of datasets that are constant and noisy. The data points in vertical lines are treated as outliers that correlates with the outliers in other datasets.



The Quantitative Metric Measuring Data Pattern Changes

Data pattern change metric:

$$\psi_j = \frac{1}{T}\sum_i \frac{1}{2f_s}\left(\left|O_j^N(d(t_i))\right| + \left|O_j^N(d(t_{i-1}))\right|\right)\delta\left(t_i - t_{i-1} = {}^1/_{f_s}\right)$$

O_j^N is the **normalized outlier** for a single data point:

$$O^N(d(t_i)) = \frac{\delta(|\Delta(t_i)| > N\sigma)}{N\sigma}\{\Delta(t_i) - sign(\Delta(t_i)) \cdot N\sigma\}$$

where

$$\Delta(t_i) = f(t_i, \{S_k\}) - d(t_i)$$

Normalized outlier is a dimensionless quantity so that different telemetry data points can compare with each other.

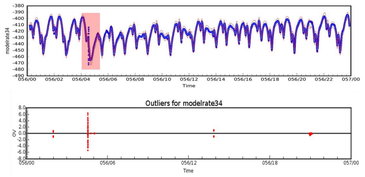


Figure 9 The outliers in the reaction wheel momentum profile (left) are converted into normalized outliers (right). The normalized outlier plots are used to highlight correlations among telemetry datasets.

The Event Vector

Both normal operations and anomalies can be regarded as events. Events are characterized by event vectors consisting of the metrics ψ_i for data pattern changes occurred in the same time period

$$e(t_i, t_f) = \left\{\frac{\psi_1}{\psi}, \frac{\psi_2}{\psi}, \dots \frac{\psi_n}{\psi}\right\}$$

where t_i, t_f: start and time of an event

$$\psi = \sqrt{\sum_i \psi_i^2}$$

- Event vectors characterize correlations among telemetry datasets. Patterns in event vectors provides signatures of normal operations or anomalies
- Event vectors provide a mathematical representation in machine learning for event classifications and anomaly detections

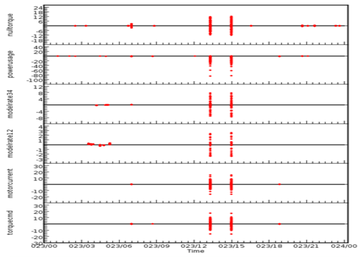


Figure 10 An example of two events with outliers aligned at the exact same time for multiple datasets in the reaction-wheel subsystem. The outlier plot provides insights into the correlations among datasets in multiple subsystems.

Event Vectors Are Hierarchical

- A dataset can be defined by its physical hierarchical path: ***Subsystem/mnemonic/index***
- An event vector can be defined at subsystem level or mnemonic level by aggregation of event vectors at lower levels

$$\psi_S = \sqrt{\sum_M \psi_{S,M}^2}$$

and

$$\psi_{S,M} = \sqrt{\sum_I \psi_{S,M,I}^2}$$

- Event vectors at higher levels reduces the space dimensions while keeping sufficient information for event classification and anomaly detections
- Perform clustering at different subsystem and mnemonic level.

$$e(t_i, t_f) = \begin{cases} \frac{\psi_{S_1}}{\psi} \begin{cases} \frac{\psi_{S_1,M_1}}{\psi} \begin{cases} \frac{\psi_{S_N,M_1,I_1}}{\psi} \\ \dots \\ \frac{\psi_{S_N,M_1,I_l}}{\psi} \end{cases} \\ \dots \\ \frac{\psi_{S_1,M_k}}{\psi} \begin{cases} \frac{\psi_{S_1,M_k,I_1}}{\psi} \\ \dots \\ \frac{\psi_{S_1,M_k,I_l}}{\psi} \end{cases} \end{cases} \\ \dots \\ \frac{\psi_{S_N}}{\psi} \begin{cases} \frac{\psi_{S_N,M_1}}{\psi} \begin{cases} \frac{\psi_{S_N,M_1,I_1}}{\psi} \\ \dots \\ \frac{\psi_{S_N,M_1,I_l}}{\psi} \end{cases} \\ \dots \\ \frac{\psi_{S_N,M_k}}{\psi} \begin{cases} \frac{\psi_{S_N,M_k,I_1}}{\psi} \\ \dots \\ \frac{\psi_{S_N,M_k,I_l}}{\psi} \end{cases} \end{cases} \end{cases}$$

Hierarchical Event Clustering

Clustering Criteria: Two events belong to the same cluster if

$$e^i(\psi_S) \cdot e^j(\psi_S) = \frac{1}{\psi^i\psi^j}\sum_S \psi_S^i\psi_S^j \geq \alpha^{th}$$

- The value for $e^i(\psi_S) \cdot e^j(\psi_S)$ has the range from 0 to 1.
- The value of αth is between 0.95 to 0.98.

The events as part of normal operations are generally repeatable and happen regularly, which form their own clusters.

The events corresponding to anomalies are generally not repeatable and happens rarely with their own characteristics, which are part of noise in event clustering.

Hierarchical Clustering Algorithm

Perform clustering for all events at subsystem level
Output noise events
For each cluster at subsystem level
Perform clustering for events mnemonic level
Output noise events

DBScan Clustering algorithm is implemented at each hierarchical level.

Clustering Outputs for LEO Satellite Data

The outputs of the clustering of 354 events over 75 days of the Suomi National Polar-orbiting Partnership (NPP) satellite telemetry data includes

- The cluster dominated by the data pattern changes in star tracker subsystem and the cluster dominated by reaction wheel subsystems.
- Two orbit maneuver events.
- Two noise events are detected that are potential anomalies.

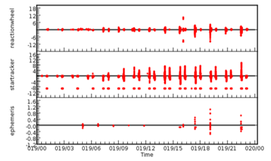


Figure 11 The signature of events as a part of the cluster dominated by data pattern changes in the star tracker subsystem..

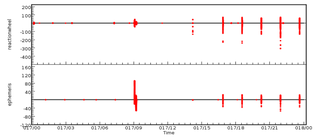


Figure 12 The signature of orbital maneuver event around 9:00z and events as a part of the cluster dominated by data pattern changes in the reaction wheel subsystem..

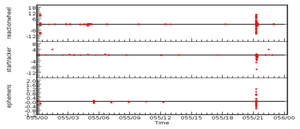


Figure 12 The signature of a noise event that is a potential anomaly.

Poster 27: Absolute Localization for Surface Robotics in GPS-Denied Environments using a Neural Network

A.S. Chung*[1], P. Ludivig*[2,3], R.W.K. Potter*[4,5], T. Seabrook*[6], and B. Wu*[7,8]

*Equal Contribution, [1]Tensorlicious, [2]ispace Europe, [3]University of Luxembourg, [4]GTA Travel, [5]Brown University, [6]University of Oxford, [7]NVIDIA, [8]National Astronomical Observatory of Japan

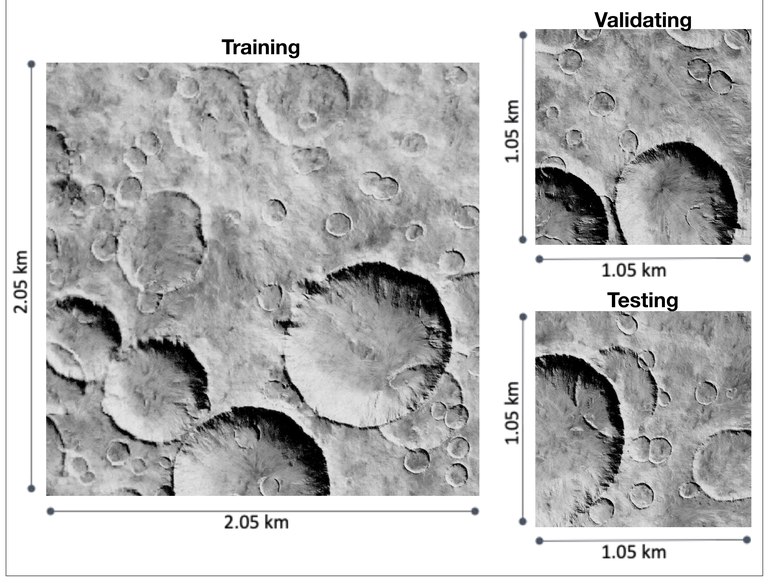
1. Introduction

- Accurate localization in planetary surface robotics is essential for navigation, path planning, and science objectives.
- On Earth, absolute localization can be readily achieved via satellite navigation (e.g., GPS). However, such systems are unavailable on other planetary bodies such as the Moon or Mars.
- Current methods rely on time- and labor-intensive human visual matching of surface perspective features with satellite images. Relative localization via visual and inertial odometry accumulate errors over time and lead to inconsistencies.
- Thus, a method that can quickly, automatically, and accurately reduce the position search space is of great benefit to future planetary exploration missions. **This project[1] presents a new approach to localizing planetary rovers: training an artificial neural network to match surface-perspective imagery to corresponding satellite maps.**

2. Methods

Simulated Environment:

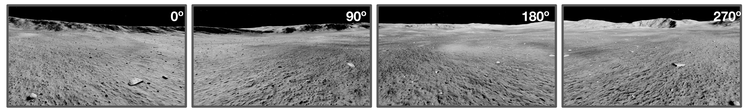
- A simulated environment was used to generate a dataset adequate in size for training a deep neural network.
- The synthetic Lunar surface environment was assembled in Unreal Engine 4[2] using MoonLandscape v3.0[3].
- Distinct zones were set for **training**, **validating**, and **testing**.



3. Methods (cont'd)

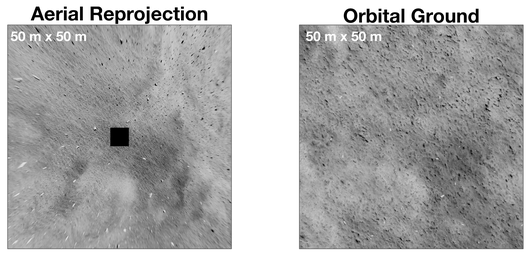
Dataset Generation:

- A virtual rover camera was spawned at distinct random locations (N~600,000) throughout the simulated environment, capturing surface-perspective images in the 4 cardinal directions as well as the corresponding ground truth orbital map.



Dataset Processing:

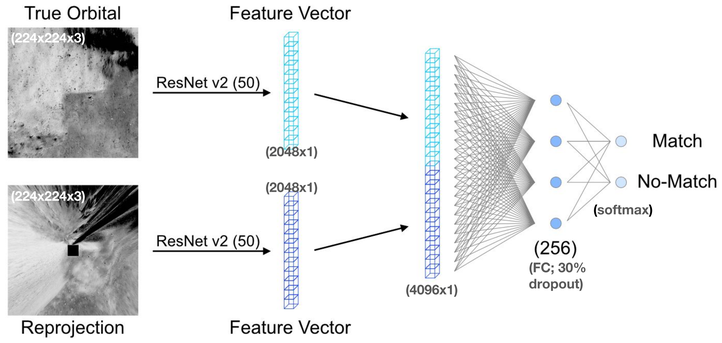
- Each set of 4 surface perspective images was then reprojected into an approximate aerial view using rover camera properties and assuming that the terrain is locally flat.



- The reprojection is paired with an orbital map of the same location (*matching*) or one from a different (*non-matching*) location with 50/50 probability, downsampled, and fed into the neural network.

Neural Network:

- ***PLaNNet*** (**P**lanetary **L**ocaliz**a**tion **N**eural **Net**work), a Siamese neural network, was trained to classify pairs of reprojections and orbital maps as *matching* or *non-matching*:
  - Each image enters a pre-trained ResNet-50 feature extractor
  - The feature vectors are concatenated, fed into a 256-neuron fully connected layer (30% dropout) to produce the final match/no-match logits vector, and softmax is applied to produce the match/no-match probability distribution for a pair of inputs.



4. Results

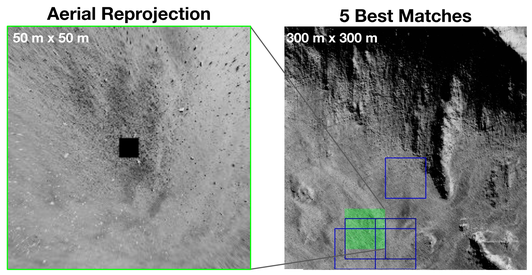
Public Dataset and Code:

- We produced a publicly available synthetic Lunar dataset and open source code for processing, training, and benchmarking localization algorithms[4]. The dataset contains 2.4+ million surface-perspective images at 600,000+ distinct locations split among the training, validating, and testing zones.

Image Type	Number	Physical Scale	Original Resolution	Downsampled Resolution
Surface Perspective	2.42×10^6	90° x 50.6° FoV	1920 x 1080 px	-
Reprojection	6.06×10^5	50 m x 50 m	1000 x 1000 px	224 x 224 px
Orbital Ground Truth	6.06×10^5	50 m x 50 m	1000 x 1000 px	224 x 224 px

Localization:

- The reprojection is compared against an array of candidate locations via a sliding window over any given orbital map. *PLaNNet* calculates probabilities of a *match* with each candidate.



- Experiments using random locations within testing zone:
  - 50 locations in $(300\text{m})^2$ subregion (3600 candidates)
  - 300 locations in full $(1.05\text{km})^2$ testing zone (40401 candidates)
- In general, *PLaNNet* returns a location within 5m of ground truth from the top 10% inferences from available candidate regions (i.e., **90% reduction of search space**). It performs >2x better than standard computer vision benchmarks (SAD/SSD/random).

Conclusions: This proof-of-concept demonstrates promising capabilities for neural network approaches to absolute localization in remote planetary surface environments. Work is in progress to include stereo camera depth information and new architectures.

Acknowledgments:
This work was published at IROS 2019[1] and originated from the NASA Frontier Development Lab (2018). Support was provided by the SETI Institute and NASA Ames Research Center. Funding was provided by Intel, Space Resources Luxembourg, and XPRIZE; compute was provided by Intel. PL thanks the Luxembourg National Research Fund (FNR 11824057).

References:

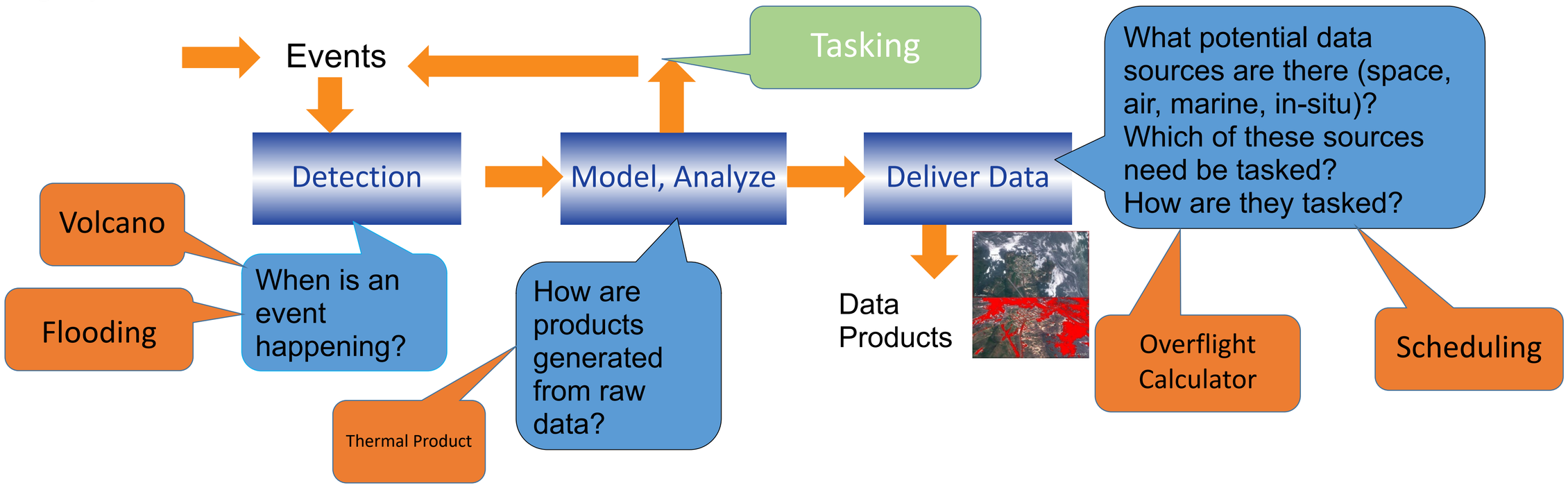
[1] Wu, B. et al. (2019) *IEEE IROS*, 3262. [2] www.unrealengine.com [3] www.unrealengine.com/marketplace/the-moon [4] http://moonbench.space/

29 Using Planet Cubesat Imagery for a Dynamic Environmental Sensor Web

James Mason[1], Steve Chien[1], Jim Boerkoel[2] , Daniel Wang[1], Ashley Davies[1], Joel Mueting[3], Vivek Vittaldev[3], Vishwa Shah[3], Ingacio Zuleta[3]
[1]Jet Propulsion Laboratory, California Institute of Technology. [2]Harvey Mudd College.
[3]Planet Labs, Inc. james.mason@jpl.nasa.gov

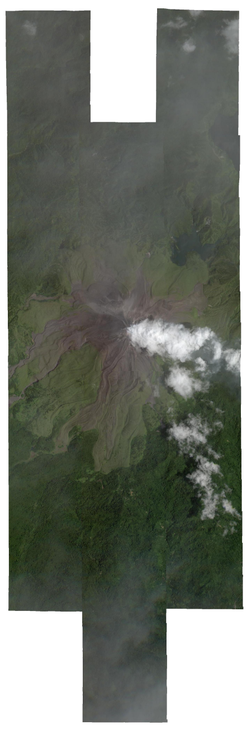
Sensorweb concept and technologies

- Aggregate data from satellite and in-situ sensors to track phenomena.
- Use automated data analysis to generate prioritized alerts
- Use federated scheduling algorithms to plan observations for a given constellation, requesting imagery from both NASA and commercial assets as appropriate

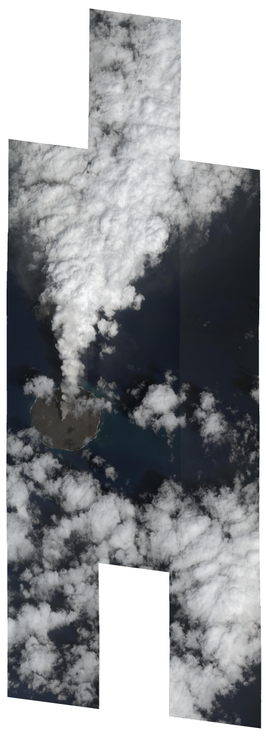


Volcano Monitoring Data Sources

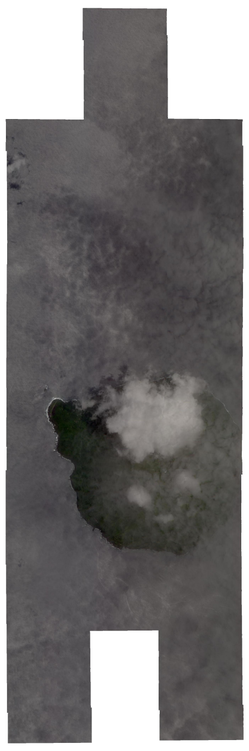
Data Source	Data Type	Spatial Coverage
MODVOLC (MODIS, Terra+Aqua)	Thermal emission	Worldwide
VIIRS Active Fires	Thermal emission	Worldwide
Iceland Met Office	Seismic	Iceland
IGEPN (Ecuador)	Reported	Ecuador
Serganomin (Chile)	Reported	Chile
USGS	Seismic	Worldwide
Volcanic Ash Advisory (VAAC)	Reported Aviation Ashe	Worldwide (7 regions implemented)



Acquired Planet Skysat Scene
13 Feb 2020
Trigger: VIIRS (Active Fire)
Target: Billy Mitchell



Acquired Planet Skysat Scene
14 Feb 2020
Trigger: VIIRS, MODVOLC
Target: Nishinoshima



Acquired Skysat Scene
14 Feb 2020
Trigger: USGS Seismic
Target: Mere Lava (Vanuatu)

End to end demonstration

We have prototyped a volcano observation sensorweb to serve as a driving use case for sensorwor web technology development. In this effort we have operationalized tracking of a number of volcano monitoring sources (see table).

In Spring 2020, we tested the above concepts in an end to end demonstration. Using the above triggers, we enabled automated tasking of the Planet Skysat constellation from a JPL sensorweb node. Three scenes were acquired in early February.

This effort is part of the Earth Science Technology Office (ESTO) New Observing Systems (NOS) Testbed. ESTO NOS is bringing together technologists and scientists to integrate disparate modules to demonstrate new observation paradigms.

We have previously used sensorwebs to task the EO-1 spacecraft for the Volcano and Thailand Flood sensorwebs.

Copyright 2021 California Institute of Technology All Rights Reserved. Government sponsorship acknowledged.
Approved for Unlimited Release CL#21-0642

30. Evaluation of Path Planning Algorithms Using a Simulation Platform for Autonomous Surface Vessels

Anete Vagale, Robin T. Bye, Ottar L. Osen

Cyber-Physical Systems Laboratory
Norwegian University of Science and Technology
Ålesund, Norway

1

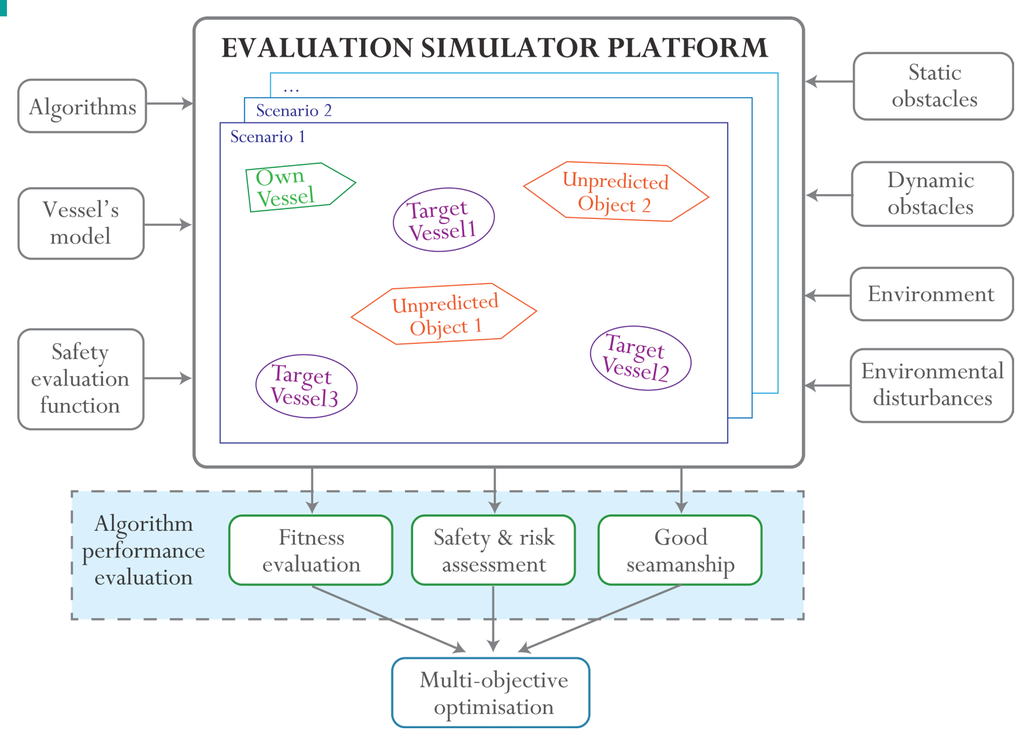


Fig. 1. The concept of the algorithm evaluation simulator platform

2

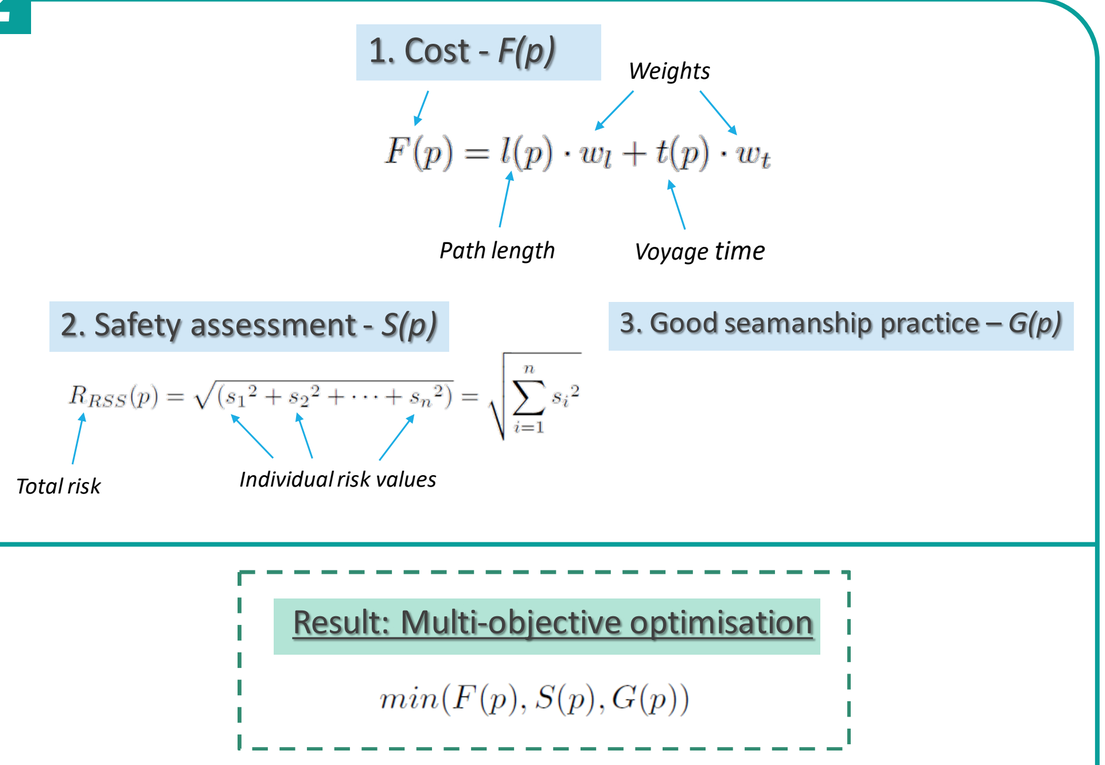


Fig. 2. Algorithm performance evaluation

3

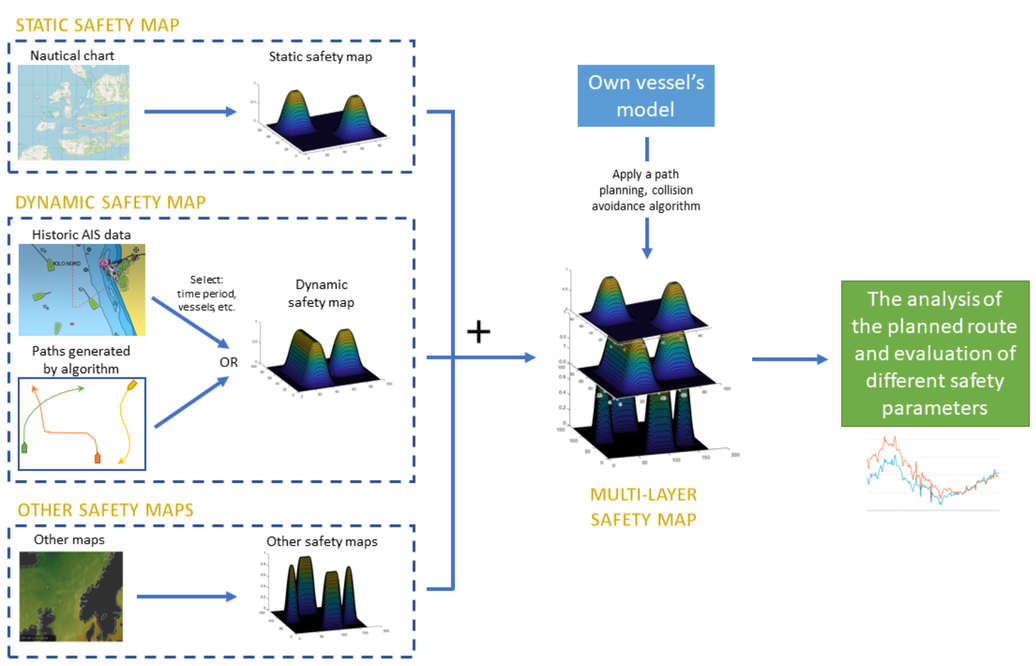


Fig. 3. The generation of each scenario

INTRODUCTION

Problem: Improved safety while navigating on waters and reduction of collision risk is a vital part of the guidance, navigation and control system of an autonomous surface vehicle.
But how to compare the performance of existing path planning and collision avoidance algorithms in a unified way?

Solution: To tackle this problem, a novel evaluation simulator platform (ESP) is proposed for simulation-based testing of algorithms.

METHODOLOGY

1) Evaluation Simulator Platform
The proposed ESP (see Fig. 1.) comprises the following input parameters needed for scenario generation and testing: *map, static obstacles, dynamic obstacles* and their movement, *environmental conditions* (such as wind, waves, current), *vessel's dynamical model*, *algorithms* that are going to be tested, and the *safety evaluation function*.
In each generated scenario, the vessel should successfully navigate in the generated environment from the initial pose to the end pose using different path planning/collision avoidance algorithms.

2) Multi-objective optimization
The ESP outputs are the algorithm performance results, based on: (i) *path evaluation* (here, a cost function), (ii) *safety and risk assessment* of the generated path, and (iii) *good seamanship practice* (the lower the score, the better) evaluation.
We propose to combine these evaluations in a ***multi-objective optimisation (MOO)*** problem (see Fig. 2.). The MOO is applied to evaluate the algorithms based on their performance ratings.

3) Safety and risk assessment
For the safety evaluation purpose, a ***multi-layer safety map*** is formed consisting of the following layers (see Fig. 3.): (i) *static safety map*, (ii) *dynamic safety map*, and (iii) *other types of safety maps*.
Individual risk values are read from the generated multi-layer safety map, and are combined into a total risk measure using a root sum square method.

CONCLUSIONS

Aiming at the problem of there being no unified way of evaluating path planning and collision avoidance algorithms for ASVs, a novel evaluation simulator platform is proposed.
In this poster we have introduced:

- a concept of a novel evaluation simulator platform (ESP),
- safety maps generation approach,
- the safety assessment method based on root sum square method,
- the total algorithm performance evaluation based on (i) *fitness* (here, a *cost function*), (ii) *safety assessment*, and (iii) *good seamanship practice*.

FUTURE WORK

Some of the future work ideas include: (i) validation of the proposed safety evaluation method, (ii) the use of maritime training simulators for validating path planning algorithms, (iii) development of a credible evaluation method for good seamanship practice and qualitative assessment, (iv) automatic scenario generation, and (v) human-in-the-loop evaluation.

Contact information:
anete.vagale@ntnu.no

NTNU
Norwegian University of Science and Technology

Jet Propulsion Laboratory
California Institute of Technology

Automated Scheduling for the ECOSTRESS Mission

Amruta Yelamanchili, Steve Chien, Alan Moy, Elly Shao, Michael Trowbridge, Kerry Cawse-Nicholson, Jordan Padams, Dana Freeborn

firstname.lastname@jpl.nasa.gov

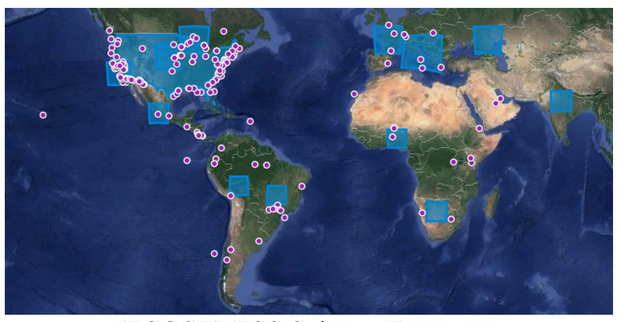
Jet Propulsion Laboratory, California Institute of Technology

Poster 32

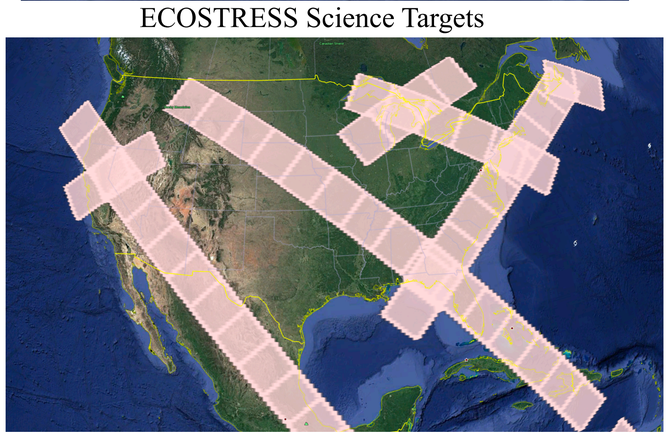
ECOSTRESS

The ECOSystem Thermal Radiometer Experiment on Space Station (ECOSTRESS) mission seeks to better understand how much water plants need and how they respond to stress. ECOSTRESS measures the temperature of plants to understand combined evaporation and transpiration, known as evapotranspiration.

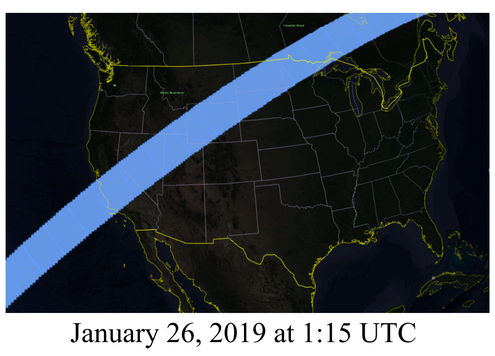
ECOSTRESS launched on June 29, 2018 to the ISS (International Space Station) on a Space-X Falcon 9 rocket as part of a resupply mission. The instrument is attached to the Japanese Experiment Module – Exposed Facility (JEM-EF) on the ISS and targets key biomes on the Earth's surface, as well as calibration/validation sites. Other science targets include cities and volcanoes. From the orbit of the Space Station the instrument can see target regions at varying times throughout the day, rather than at a fixed time of day, allowing scientists to understand plant water use throughout the day. ECOSTRESS is expected to operate until mission end in Summer 2019.



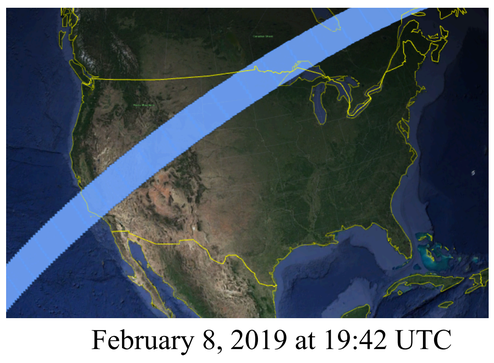
ECOSTRESS Science Targets



Observations scheduled over North America on February 28, 2019



January 26, 2019 at 1:15 UTC



February 8, 2019 at 19:42 UTC

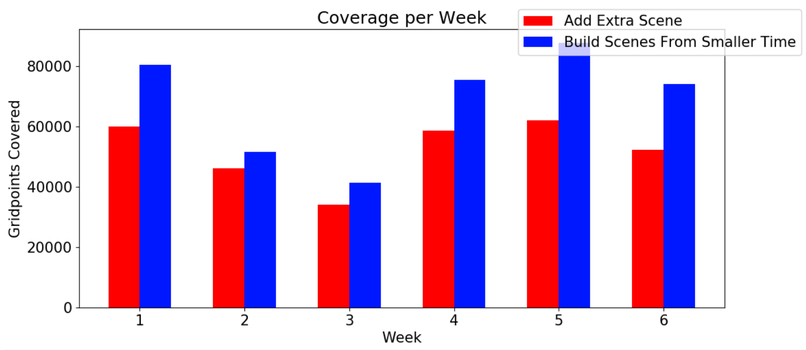
Two Orbital Tracks of the ISS that view the same regions but at different illuminations

Orbit Uncertainty

Command sequences are uploaded weekly to ECOSTRESS. Being in Low Earth Orbit (LEO), the ISS experiences some drag from the atmosphere, causing it to drift over time from its predicted location at the beginning of the week. This can cause a planned observation to miss its intended target.

Two strategies were developed to deal with the uncertainty. They needed to account for the fixed size of observations that ECOSTRESS takes and at least a few seconds of extra time to ensure a target is not missed.

1) Static Padding - Add ½ of an observation to the start and end of each contiguous observation
2) Targeted Padding - analyzes the target regions and calculates padding around the targets



When working with the same fixed data volume constraints, Targeted Padding uses less data volume on padding, therefore enabling it to cover more science targets as seen above.

Mass Storage-less Operations

After further operations, both Mass Storage Units on ECOSTRESS became non-functioning. Updates to the instrument firmware were developed to allow data to be acquired and downlinked avoiding the MSU. CLASP was adapted again to be able to schedule with a new set of constraints.

New constraints included:

- Reduced onboard storage capability
- Greater downlink capability with removal of certain spectral bands
- All data onboard must be downlinked at once to regain use of storage

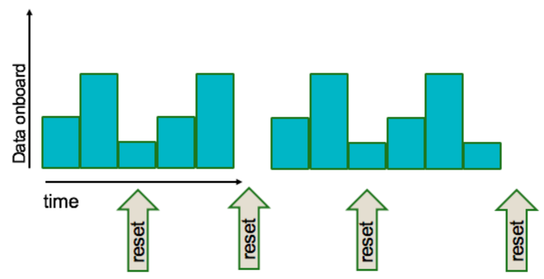
CLASP

The Compressed Large-scale Activity Scheduler and Planner (CLASP) has been in use in various ways for ECOSTRESS. Coverage planning technology in the CLASP scheduler has been used in the aid of:

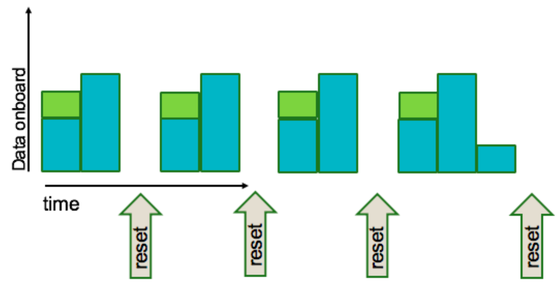
- evaluating designs of the overall science campaign implementation prior to launch
- Generating command sequences for operations
- designing new science coverage strategies
- updated scheduling approaches to address hardware challenges on orbit

Ring Buffer Issue

The Mass Storage Unit firmware onboard ECOSTRESS implemented a ring buffer. However, the read and write pointers did not correctly handle the buffer boundary, causing lost data. Rather than perform a firmware update to the instrument, CLASP was updated to automatically schedule resets of the ring buffer to avoid the issue.



High priority targets are initially scheduled to determine natural places where resets of the ring buffer should occur – when the amount of data onboard will be low and the end of the buffer has not been reached.



High and low priority targets are then scheduled, accounting for the times the ring buffer resets occur.

Radiation Sensitivity

One of the possible reasons for stalls of the Mass Storage Unit on ECOSTRESS was radiation sensitivity. It was found that most of the stalls were occurring in areas that had higher levels of radiation, like over the South Atlantic Anomaly and Low Latitudes. Many strategies for mitigating this were analyzed, and CLASP assessed the impact of the strategies on coverage.

Strategies included:

- Taking data and downlinking only on orbits overflying Australia
- Turning off the MSU over the South Atlantic Anomaly and Southern High Latitudes
- Not downlinking data over High Latitudes

Regions where the MSU needed to be powered down due to radiation

References

1. Yelamanchili, A.; Chien, S.; Moy, A; Shao, E.; Trowbridge, M.; Cawse-Nicholson, K.; Padams, J.; and Freeborn, D.; "Automated science scheduling for the ECOSTRESS mission", International Conference on Automated Planning and Scheduling (ICAPS 2019) Scheduling and Planning Applications Workshop (SPARK), Berkeley, CA, July 2019, in preparation. Also appears in International Workshop for Planning and Scheduling for Space (IWPSS 2019).
2. Hook, S. 2019. Ecostress https://ecostress.jpl.nasa.gov.
3. Knight, R., and Chien, S., "Producing large observation campaigns using compressed problem representations", International Workshop on Planning and Scheduling for Space (IWPSS 2006), 2006.
4. Knuth, D. E. 1997. The art of computer programming. volume 1, 244–245
5. Rabideau, G.; Chien, S.; Mclaren, D.; Knight, R.; Anwar, S;, Mehall, G. and Christensen, P.; "A tool for scheduling themis observations", International Symposium on Space Artificial Intelligence, Robotics, and Automation for Space (ISAIRAS 2010), 2010.
6. Troesch, M.; Chien, S.; and Ferguson, E. 2017. "Using automated scheduling to assess coverage for europa clipper and jupiter icy moons explorer", International Workshop on Planning and Scheduling for Space (IWPSS 2017), 2017.

JPL Clearance CL 21-0369

National Aeronautics and Space Administration
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California
www.nasa.gov

Copyright © 2019 California Institute of Technology.
U.S. Government sponsorship acknowledged

Automated Scheduling for the Orbiting Carbon Observatory-3 Mission

Amruta Yelamanchili, Christopher Wells, Alan Moy, Steve Chien, Annmarie Eldering, Ryan Pavlick

firstname.lastname@jpl.nasa.gov

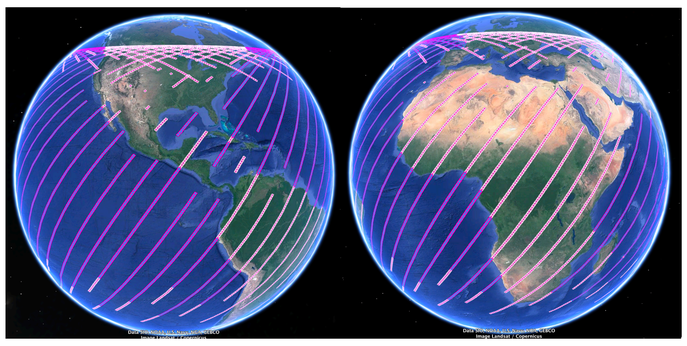
Jet Propulsion Laboratory
California Institute of Technology

Poster 33

Summary

The Orbiting Carbon Observatory-3 (OCO-3) is a NASA instrument for measuring atmospheric CO_2. OCO-3 launched to on May 4, 2019 to the ISS (International Space Station) on a SpaceX Falcon 9 rocket as part of a resupply mission. It is mounted on the International Space Station on the Japanese Experiment Module – Exposed Facility (JEM-EF). It is expected to begin nominal science operations in August 2019 and its planned mission duration is three years. OCO-3 will enable identification of CO_2 sources and sinks and study changes in CO_2 levels over time.

Automated scheduling is being deployed for operations of OCO-3. The OCO-3 scheduling process begins with a mostly-automated dynamic science priority assignment that is input to an automated scheduling of area targets, calibration targets, nadir, and glint mode. It is also being used to schedule observations for the calibration of the pointing mirror.

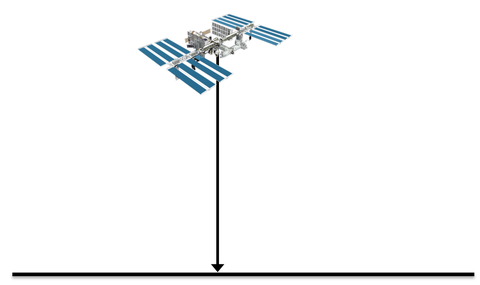


A visualization of part of a generated schedule

Operational Modes

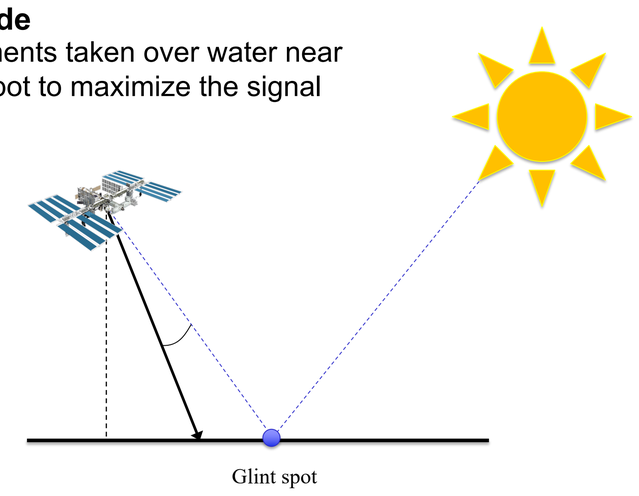
Nadir mode

Default mode over land in the daytime



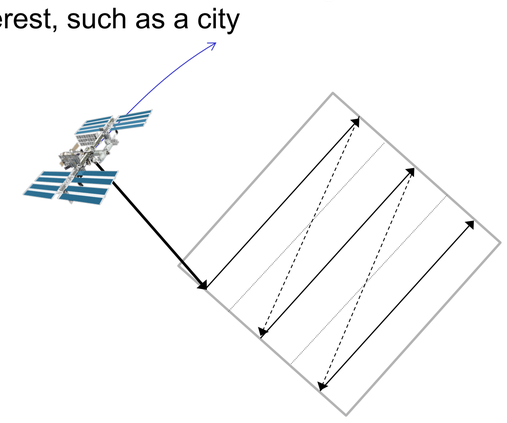
Glint mode

Measurements taken over water near the glint spot to maximize the signal



Area Mapping mode

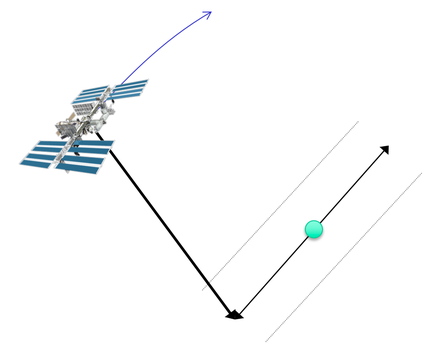
Measurements taken over regions of interest, such as a city



Observe several non-overlapping stripes over the target area

Target Mapping mode

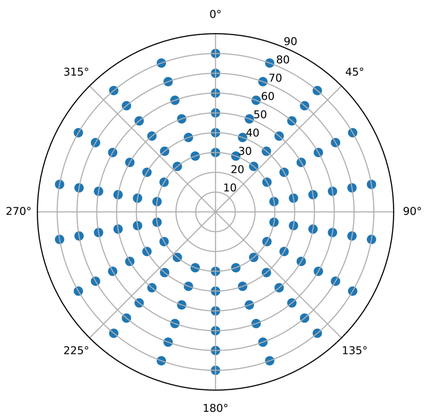
Measurements taken over a specific point, such as a validation site



Observe a single stripe over the target point repeatedly

Pointing Mirror Assembly Calibration

To calibrate the Pointing Mirror Assembly (PMA), observations are taken from a set of pointings relative to the instrument body. The observations must be taken over land in daytime to be compared to reference images to determine the error in the pointing based on the ISS location. Corrections are then applied to minimize the pointing error for future observations.



Azimuth, elevation target points that are observed to calibrate the PMA

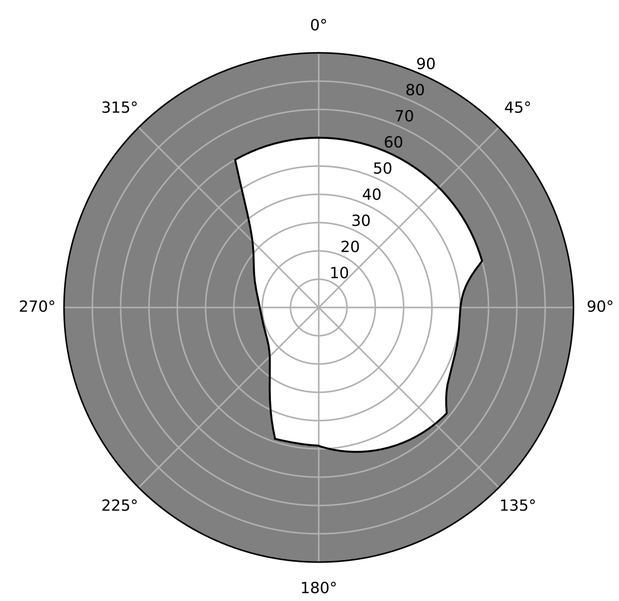
Checking Visibility

OCO-3's field of view is limited due to obstructions by other components of the ISS. The scheduling software must account for the limitation without sacrificing too much runtime or precision. The occlusion mask is defined as a set of polynomials for several longitude segments. We currently check visibility for area map mode, target mode, and glint mode.

To check if a target is visible at a particular point in time, we project the target onto the unit sphere around the satellite. This gives us an azimuth/elevation point that can be checked for inclusion in the visibility set.

For glint mode, this approach is satisfactory. But for area map and target mode, we do not know exactly where the PMA is pointed at any time. We consider three approaches to addressing this complication:

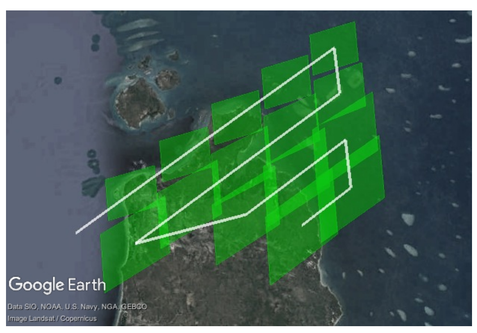
- **Centroid:** Project the centroid of the target and check visibility set
- **Corners:** Project the corners of the target and check visibility set
- **Configuration space:** Define a configuration space that characterizes all points on the unit sphere that represent the centroid of a visible target. Project only the centroid of the target on the unit sphere and check if that the point falls within the configuration space.



A visualization of the "keepout zone" (in grey) that the scheduling software must avoid pointing the PMA at

Agile pointing

While the current OCO-3 scheduler does not schedule fine instrument pointing (this is controlled in flight software), technology exists in the Eagle Eye system[12] to construct detailed pointing plans to cover arbitrary polygons.



Agile pointing of 2d sensor to cover area[12]

References:

[1] Moy, A.; Yelamanchili, A.; Chien, S.; Eldering, A.; and Pavlick, R. 2019. Automated Scheduling for the Orbiting Carbon Observatory 3 Mission. International Workshop on Planning and Scheduling for Space (IWPSS 2019) in preparation.

[2] Acton, C. 2019. Naif https://naif.jpl.nasa.gov/naif.

[3] Chien, S.; Doubleday, J.; Thompson, D. R.; Wagstaff, K.; Bellardo, J.; Francis, C.; Baumgarten, E.; Williams, A.; Yee, E.; Stanton, E.; and Piug-Suari, J. 2016. Onboard autonomy on the intelligent payload experiment (ipex) cubesat mission. *Journal of Aerospace Information Systems (JAIS).*

[4] Doubleday, J., and Knight, R. 2014. Science mission planning for nisar (formerly desdyni) with clasp. In *SpaceOps 2014*.

[5] Doubleday, J. R. 2016. Three petabytes or bust: Planning science observations for nisar. In *SPIE 9881*.

[6] Hurkala, J. 2015. Time-dependent traveling salesman problem with multiple time windows. In *Position Papers of the 2015 Federated Conference on Computer Science and Information Systems*, *volume 6 of Annals of Computer Science and Information Systems*, 71–78.

[7] Knight, R., and Chien, S. 2006. Producing large observation campaigns using compressed problem representations. In *International Workshop on Planning and Scheduling for Space (IWPSS-2006)*.

[8] Rabideau, G.; Chien, S.; Mclaren, D.; Knight, R.; Anwar, S.; Mehall, G.; and Christensen, P. 2010. A tool for scheduling themis observations. In *International Symposium on Space Artificial Intelligence, Robotics, and Automation for Space (ISAIRAS 2010)*.

[9] Roussel, S.; Pralet, C.; Jaubert, J.; Queyrel, J.; and Duong, B. 2017. Planning the observation of exoplanets: the ariel mission. In *International Workshop on Planning and Scheduling for Space (IWPSS 2017)*.

[10] Schubert, E.; Zimek, A.; and Kriegel, H.-P. 2013. Geodetic distance queries on r-trees for indexing geographic data. In *Advances in Spatial and Temporal Databases*, 146–164.

[11] Troesch, M.; Chien, S.; and Ferguson, E. 2017. Using automated scheduling to assess coverage for europa clipper and jupiter icy moons explorer. In *International Workshop on Planning and Scheduling for Space (IWPSS 2017)*.

[12] Shao, E.; Byon, A.; Davies, C.; Davis, E.; Knight, R.; Lewellen, G.; Trowbridge, M.; and Chien, S. 2018. Area coverage planning with 3-axis steerable, 2D framing sensors. In *Scheduling and Planning Applications Workshop , International Conference on Automated Planning and Scheduling (ICAPS SPARK 2018).*

National Aeronautics and Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

www.nasa.gov

JPL Clearance CL 21-0368

Copyright © 2019 California Institute of Technology.
U.S. Government sponsorship acknowledged.

National Aeronautics and Space Administration

Poster #16

Front Delineation and Tracking with Multiple Underwater Vehicles

Andrew Branch[1], Mar M. Flexas[2] Martina Troesch[1], Brian Claus[3], Andrew F. Thompson[4], Yanwu Zhang[4], Evan B. Clark[1], Steve Chien[1], David M. Fratantoni[5], James C. Kinsey[3], Brett Hobson[4], Brian Kieft[4], Francisco P. Chavez[4]

[1]Jet Propulsion Laboratory, California Institute of Technology
[2]California Institute of Technology
[3]Woods Hole Oceanography Institution
[4]Monterey Bay Aquarium Research Institute
[5]Remote Sensing Solutions

Corresponding Author: andrew.branch@jpl.nasa.gov

Project Summary:

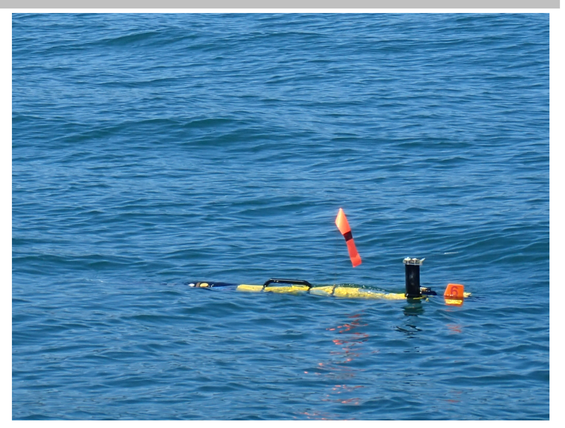
Space based remote sensing provides great information about ocean dynamics. However, remote sensing information is generally limited to measuring the ocean surface or the upper layer of the ocean. Ocean models can further augment this information. However, in order to probe the immense volume of the ocean most accurately generally requires marine vehicles such as autonomous underwater vehicles (AUVs), Seagliders, profiling buoys, and surface vehicles sampling in-situ. Deploying and operating these assets is very expensive. This means there is a very limited number of marine vehicles compared to the massive size of the ocean. Knowing where the assets should be deployed and operated is very difficult.

One strategy is to deploy in-situ assets to study specific scientific features such as fronts, eddies, upwellings, harmful algal blooms, or other features of interest. A typical strategy would be to deploy marine assets to measure transects across the feature of interest at a scale that covers the feature, as well as a baseline signal around the feature. However, asset capabilities (e.g. mobility, endurance) and prevailing ocean currents may render these science goals unachievable. Our project targets automatic detection and tracking of these features and generation of mission plans for assets to follow these science derived templates relative to projected feature motion.

From 22 October to 30 October 2016 the project team deployed an underwater glider executing feature prediction using a predictive ocean model and adaptive sampling near Monterey Bay, CA. The glider was commanded along a given transect, at each surfacing features were extracted from the model based on the vehicles expected path. The vehicle was then commanded along the path that resulted in the observation of the most interesting science features.

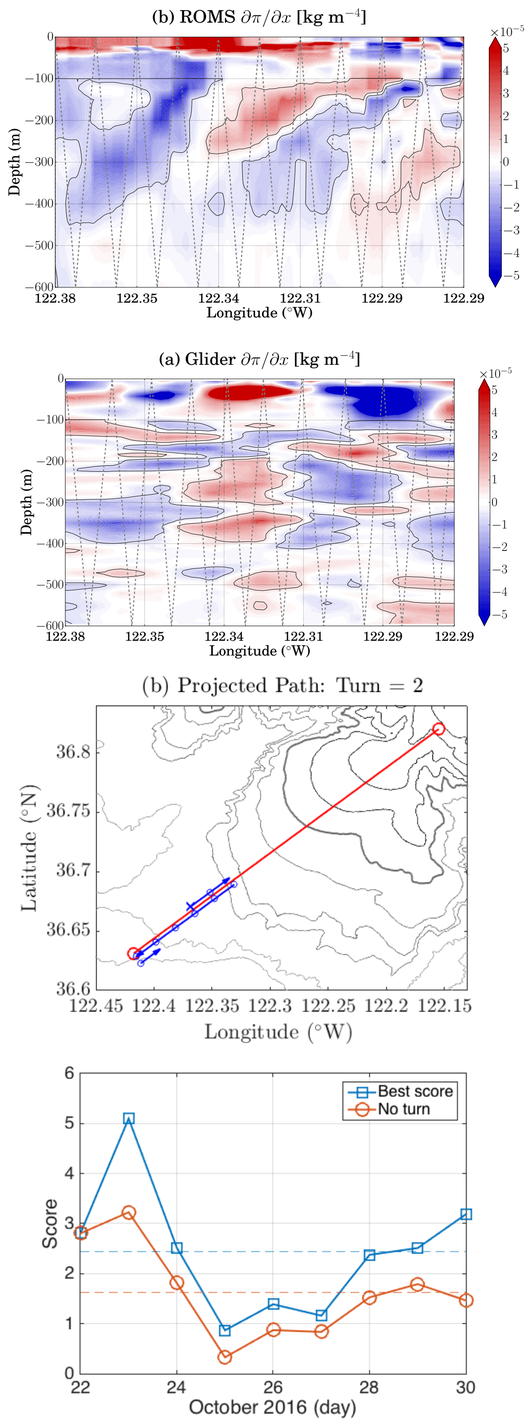
From 27 April to 11 May 2017 the project team, along with MBARI, deployed a heterogeneous fleet of marine vehicles including Tethys and Iver AUV's and gliders executing linear front detection and retasking near Monterey Bay, CA. Multiple vehicles were used performing parallel transects to autonomously detect an ocean front, fit a linear estimate to those detections, and re-command the vehicles to repeatedly cross the front.

Iver AUV on deck



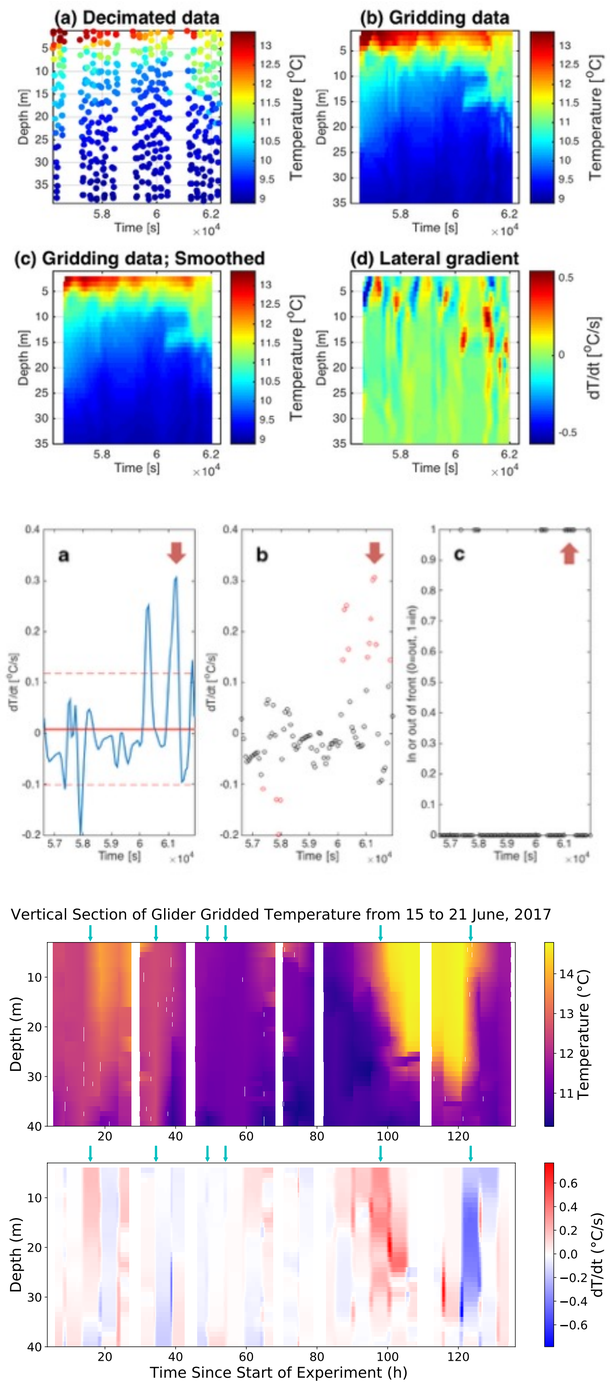
Deployed Iver AUV

Glider Feature Chasing



2017 Deployment Map

Linear Front Detection and Tracking



National Aeronautics and Space Administration
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

www.nasa.gov

Portions of this work were performed by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

Copyright 2021. All rights reserved

Publications:

A. Branch, E. Clark, S. Chien, M. Flexas, A. Thompson, B. Claus, J. Kinsey, D. Fratantoni, Y. Zhang, B. Hobson, B. Kieft, and F. Chavez, Front Delineation and Tracking with Multiple Underwater Vehicles. *Journal of Field Robotics (JFR)*, 0(0). December 2018.

M. Flexas, M. Troesch, S. Chu, A. Branch, S. Chien, A. Thompson, J. Farrara, Y. Chao, Autonomous sampling of ocean submesoscale fronts with ocean gliders and numerical model forecasting. *Journal of Atmospheric and Oceanic Technology*, 35 (3). March 2018.

E. Clark, A. Branch, S. Chien, F. Mirza, J. Farrara, Y. Chao, D. Fratantoni, D. Aragon, O. Schofield, M. Flexas, and A. Thompson, Station-Keeping Underwater Gliders Using a Predictive Ocean Circulation Model and Applications to SWOT Calibration and Validation. *Journal of Oceanic Engineering (JOE)*. January 2019.

A. F. Thompson, Y. Chao, S. Chien, J. Kinsey, M. M. Flexas, A. Branch, S. Chu, M. Troesch, B. Claus, J. Kepper, J. Farrara, D. Fratantoni, "Satellites to Seafloor: Towards Fully Autonomous Ocean Sampling" Oceanography. 30 (2) June 2017.

D. Fratantoni, A. Branch, Y. Chao, F. Chavez, S. Chien, S. Chu, E. Clark, B. Claus, Z. Erickson, J. Farrara, M. Flexas, J. Kepper, B. Kieft, J. Kinsey, B. Hobson, A. Thompson, M. Troesch, W. Yuan, Y. Zhang, Towards fully autonomous ocean observing: Coupling heterogeneous robotic arrays with data-assimilating models and autonomous path planning, Ocean Sciences Meeting, Portland, OR, February 2018.

Acknowledgements to entire KISS Team:
Caltech: Andrew Thompson, Mar Flexas, Zachary Erickson
Woods Hole Oceanographic Institution: James Kinsey, Brian Claus, James Kepper, Stefano Suman
Remote Sensing Solutions: Yi Chao, John Farrara, David Fratantoni
Special thanks to MBARI (F. Chavez, B. Hobson, B. Kieft) for providing the Tethys Class LRAUVs
Special thanks to Rutgers University (David Aragon) for providing the Slocum Underwater Gliders

Jet Propulsion Laboratory
California Institute of Technology

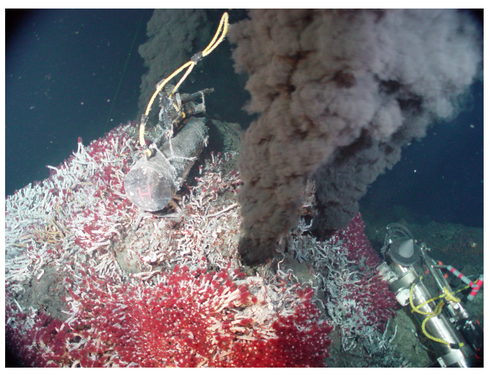
Autonomous Nested Search for Hydrothermal Venting

Andrew Branch[1], James Mason[1], Guangyu Xu[2], Michael V. Jakuba[2], Christopher R. German[2], James McMahon[3], Steve Chien[1], James C. Kinsey[2], Andrew D. Bowen[2], Kevin P. Hand[1], Jeffrey S. Seewald[2]

[1]Jet Propulsion Laboratory, California Institute of Technology
[2]Woods Hole Oceanographic Institution
[3]Naval Research Laboratory

Poster #17

Image Credit: Marcel Nicolaus, AWI

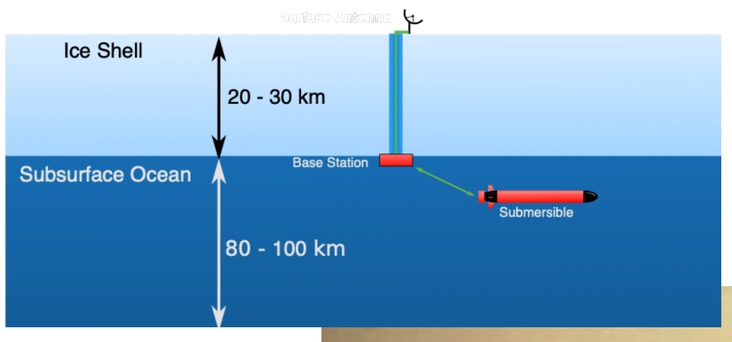


Credit: IFE, URI-IAO, UW, Lost City Science Party; NOAA/OAR/OER; The Lost City 2005 Expedition

Motivation

- 8+ bodies in the Solar System are thought to harbor sub-surface liquid oceans, including Europa and Enceladus
- Earth-based hydrothermal vents harbor unique life and are potentially crucial to the origin of life
- Potential hydrothermal venting on Europa
- Evidence for hydrothermal venting on Enceladus [4,5]

Problem Definition



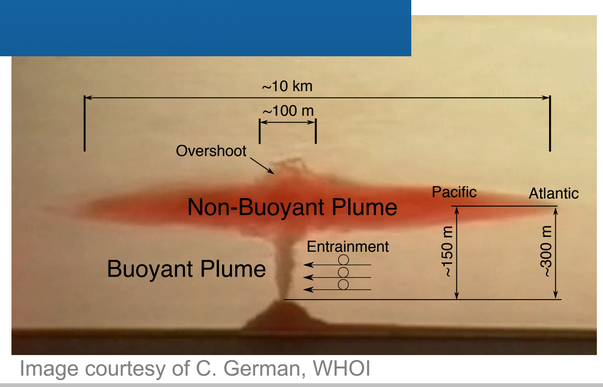


Image courtesy of C. German, WHOI

Ocean Worlds Submersible

- Single under-ice base station provides communication to Earth
- Travel 100s of km from base station
- Limited communication with Earth due to orbital occlusions and underwater acoustic communication range
- Fully autonomous operations required for weeks or months at a time
- **Goal: Autonomously detect, locate, and sample hydrothermal venting**

Hydrothermal Plumes

- Chemically altered seawater detectible through temperature, redox, and optical backscatter
- Hot, low density plume fluid exits vent forming buoyant plume [2]
- Density equilibrium reached and non-buoyant plume formed [2]
- Source vent can be tracked using the hydrothermal plume

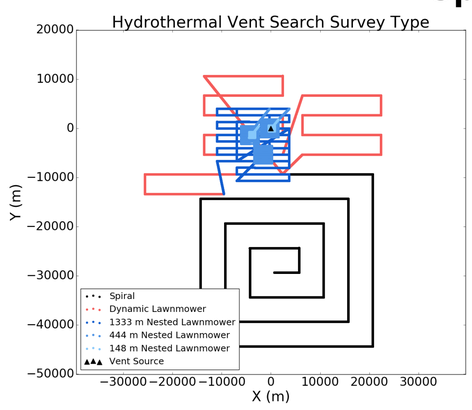
Objectives

- Perform high resolution survey of region immediately surrounding hydrothermal venting
- Autonomous adaptation of proven human-in-the-loop search method [3]
- Must maintain robustness to local maxima and small-scale turbulence

Approach

- Progressively higher resolution nested surveys to pinpoint maximum vent fluid concentration
- Three Search Phases:
  - Initial spiral survey
  - Dynamic lawnmowers
  - Nested lawnmowers

Spatial Nested Search

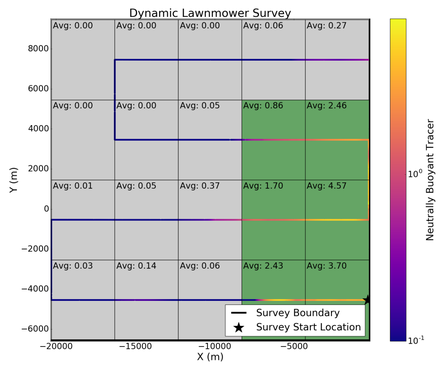


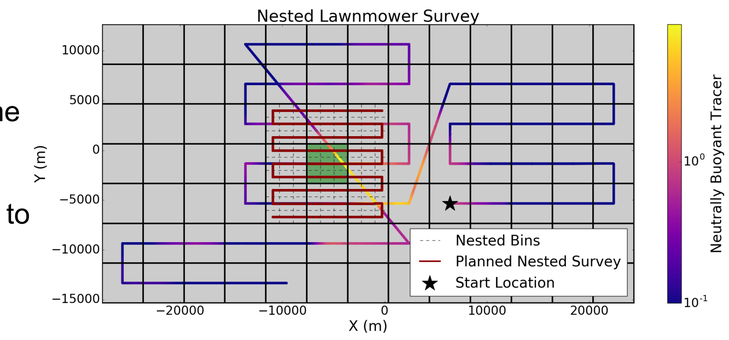
Spiral Survey

- Spiral centered at the base station
- Yo-Yo pattern from surface to seafloor to locate non-buoyant plume
- Terminates on first contact with plume above specified threshold

Dynamic Lawnmower

- Performed at first contact with plume
- Variable size lawnmower survey to determine the extent of the plume
- Data binned at resolution of survey to determine local maxima





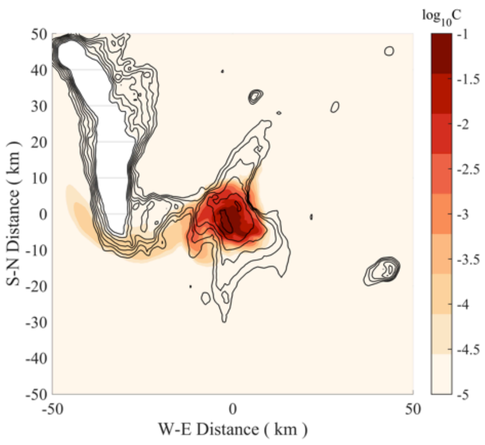
Nested Lawnmowers

- Perform recursively higher resolution surveys of previously searched regions
- Each survey encompasses local maxima and surrounding bins
- Prioritized based on average plume strength of bin and survey resolution

Simulation

Hydrothermal Venting Model

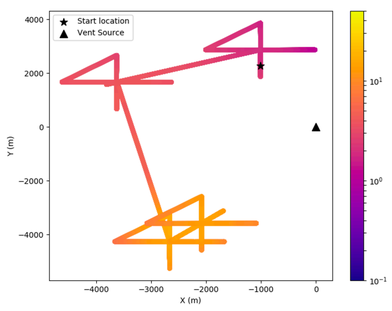
- FVCOM based circulation and hydrothermal plume model of Axial Seamount
- 1 GW heat source in the Axial caldera
- Initial forcing constructed with HYCOM and OSU Tidal Inversion models
- 300x300 km, 60 day simulation
- Model variables: temperature, salinity, currents, passive tracer



Baseline Search Methods

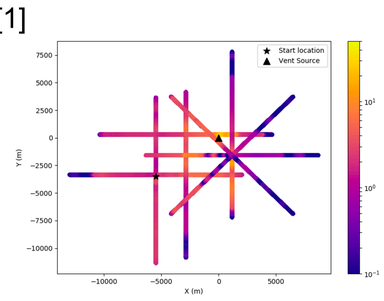
Gradient Ascent

- Determine the plume gradient at a location and follow it towards stronger plume fluid [1]

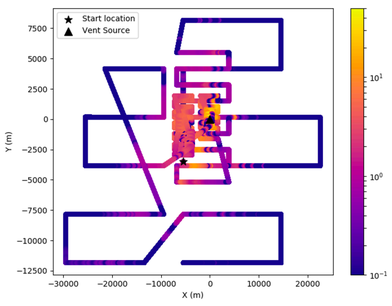


Greedy Transect

- Direction Set: Perform fixed pattern searching for increased plume strength. Repeat this pattern at new maxima until no new max plume values are seen [1]



Results



Example Nested Bin Search Result

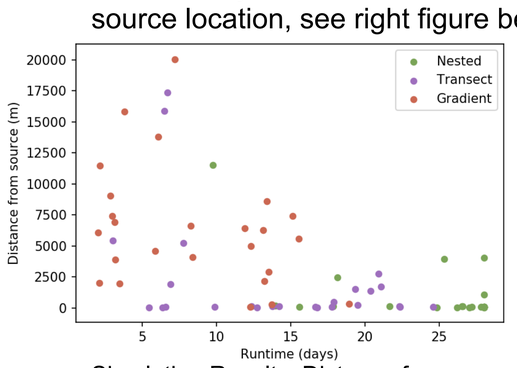
Simulated Results

- 25 simulation runs per algorithm with starting x,y uniformly distributed from [-30,30] km
- Nested Search better estimates the vent location over baseline methods, however with longer search times. See table and left figure below.

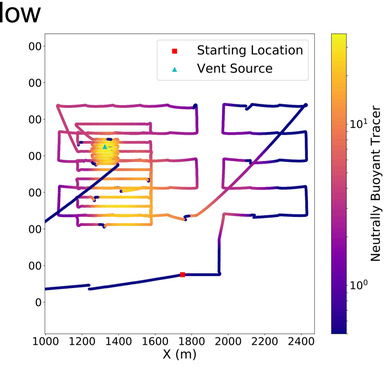
Algorithm	Nested Search	Gradient Ascent	Greedy Transect
Success Rate (< 200m of ground truth)	80%	56%	4%

Deployment

- Deployed Nested Search algorithm onboard an Iver AUV in Chesapeake Bay with NRL in June 2019
- Successfully demonstrated the vehicle locating the simulated vent source location, see right figure below



Simulation Results: Distance from True Vent Location Vs. Time



Demonstration onboard Iver AUV

[1] Burian, Erik, et al. 1996. Gradient search with autonomous underwater vehicles using scalar measurements." *Proceedings of Symposium on Autonomous Underwater Vehicle Technology*. IEEE.
[2] German, C., and Seyfried, W. 2014. Hydrothermal processes. Treatise on geochemistry 8:191–233.
[3] German, C. R. el al. 2008. Hydrothermal exploration with the autonomous benthic explorer. Deep Sea Research Part I: Oceanographic Research Papers 55(2):203–219.
[4] Hsu, H. W. et al. 2015. Ongoing hydrothermal activities within enceladus. Nature 519(7542):207.
[5] Waite, J. H. et al. 2017. Cassini finds molecular hydrogen in the Enceladus plume: Evidence for hydrothermal processes. Science 356(6334):155–159.

National Aeronautics and Space Administration
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California
jpl.nasa.gov

Acknowledgments
Portions of this work were performed by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

© 2021 California Institute of Technology. U.S. Government sponsorship acknowledged.

Predecisional, for planning and discussion only.