CS486C – Senior Capstone Design in Computer Science

Project Description

Hawaiian Ecosystem High Temperature Alert System	
Sponsor Information:	**Benjamin Wiebe,** PhD Student in Ecological Informatics School of Informatics, Computing, and Cyber Systems Northern Arizona University bcw269@nau.edu **Dr. Camille Gaillard,** Post-Doctoral Researcher in Ecological Informatics School of Informatics, Computing, and Cyber Systems Northern Arizona University Camille.Gaillard@nau.edu **Dr. Christopher Doughty,** Professor of Ecological Informatics School of Informatics, Computing, and Cyber Systems Northern Arizona University chris.doughty@nau.edu

Project Overview

The islands of Hawaii hold immense ecological and cultural value but are susceptible to increasing hot and dry conditions associated with climate change. This susceptibility was on tragically evident in August 2023 when multiple wildfires across Maui caused at least 106 human deaths, billions of dollars in damages to infrastructure, and vast ecological consequences. Fires in Hawaii often begin in grasslands, where increasing drought and heat decrease fuel moisture and allow small ignitions to grow into massive wildfires before spreading into surrounding forests and urban areas[1]. Other Hawaiian ecosystems are also vulnerable to high temperatures. Marine heatwaves are causing more frequent and significant bleaching events in the islands' coral reef systems[2]. Additionally, tropical forest canopies around the world, including those in Hawaii, are approaching temperatures that are harmful to the fundamental machinery of photosynthesis[3].

Figure 1: Wildfires devastate Maui, August 2023
Photo: Matthew Thayer, The Maui News

With each of these ecological vulnerabilities, high-resolution satellite thermal remote sensing emerging as a crucial tool for ecological monitoring. It holds the growing promise of alerting researchers and natural resource managers when temperatures are approaching critical thresholds, ideally helping in the mitigation of potential damages. For instance, a thermal infrared image of Maui, Hawaii from NASA's Landsat sensor on August 7, 2023 (Figure 2) showed grassland temperatures ranking in the top 4% of the last decade – conditions all too favorable for the devastating fires that ignited and rapidly spread the following day. Thermal satellite data quantity and quality are increasing rapidly, with multiple NASA sensors (e.g. Landsat, MODIS, ECOSTRESS) now collecting global surface temperature data at sub-weekly timescales (Figure 2), and private companies (e.g. Hydrosat) promising to offer multiple high-resolution temperature readings daily across the globe within 3-5 years. The Doughty Lab at NAU concentrates on leveraging these cutting-edge satellite tools to better understand the impact of heat waves and drought on ecosystems. A recent

[1] https://www.nature.com/articles/d41586-023-02571-z
[2] https://www.nature.com/articles/d41586-019-02802-2
[3] https://www.nature.com/articles/s41586-023-06391-z

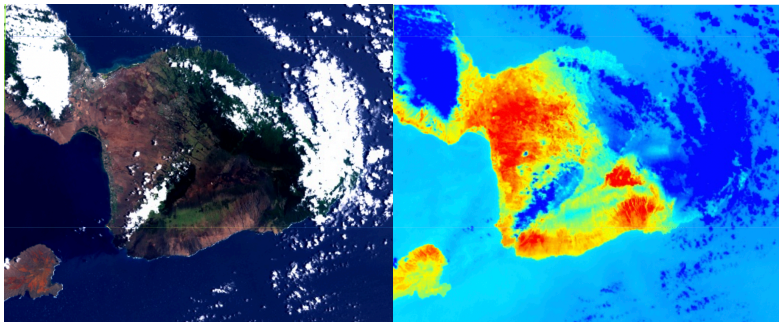


Figure 2: Landsat 8 satellite images of Maui, Hawaii from August 7, 2023, the day before devastating wildfires broke out across the island. *Left:* RGB image. *Right:* Corresponding thermal infrared image shows hot conditions in the grasslands on the western side of the island, where fires broke out the following day.

article in the journal Nature by Dr. Chris Doughty and colleagues uses NASA's high-resolution ECOSTRESS sensor aboard the International Space Station to quantify risk of extreme heat in tropical forests[1].

While satellite thermal data are freely available from NASA, there is currently no system in place for aggregating real-time data from different platforms (especially newer, high-resolution ones like ECOSTRESS), automating cloud masking and quality control, and comparing current data with historical averages to provide alerts of high temperature anomalies on various land surfaces. Further, the access, processing, and utilization of these data still largely require programming knowledge and are largely underutilized by land managers, local governments, and the public. We are seeking to develop a GUI web-app that automatically aggregates and processes satellite thermal data from multiple sources, compares them to historical averages to highlight temperature anomalies, and presents them in a highly accessible and user-friendly interface for a range of resource management applications.

Some of the key functions supported by this product should include:

- Minimum Viable Product: A web-app (preferably written in python) providing visualization of near real time land surface temperature data anomaly compared to historical averages
    - Data pipeline for automatically ingesting large amounts of near real time quality-controlled GeoTiffs (10s gigabytes weekly) from NASA APIs with minimal latency (<1day)
        - *Note:* GeoTiff files are image files (Tiffs) with various pieces of georeferencing information (e.g. spatial extent, geographic projection) embedded in the metadata. In this case, these are essentially images where each pixel represents a measurement of the land surface temperature at a specific location on earth.
        - GeoTiffs will be obtained from the NASA APPEARS API. We have already developed a python pipeline that automatically requests, downloads, and does quality control for 100s of thousands of ECOSTRESS files at a time that can be adapted for the purposes at hand.
    - GUI for visualizing satellite land surface temperatures and anomalies. Should include the option to toggle between layers, scroll bar to show data through time, and should include necessary legends. There are many available tools for visualizing satellite image data. We familiar with some on Python and R, and while we are less familiar with those for embedding maps on web-apps, we are confident that good tools exist. It will be at the CS capstone teams discretion as to what to use.
- A Useful System: A product that would allow the above functionality in addition to:
    - Incorporate ≥3 different satellite data sources (e.g. ECOSTRESS, MODIS, Landsat, VIIRs) for more frequent observations and other benefits (e.g. higher spatial resolution). Data pipeline should be built with flexibility in mind to accommodate the future flood of thermal data expected in 2-4 years.
    - More advanced GUI functionality:

[1] https://www.nature.com/articles/s41586-023-06391-z

        - Users will be able to draw areas of interest (AOIs) and define alert parameters for being notified if surface temperatures are being exceeded. For example, they could draw a polygon around a reef system and tell the web app to email them if the surface water temperatures ever exceed 30ºC.
        - Users can define an AOI and visualize the historical temperature data from that AOI. For example, they could see a timeseries of the min / max / mean temperatures from their AOI using all available data.
        - Allow .csv export of data from AOI
    - *Note:* The backend will automatically (e.g. daily) ingest any new quality-controlled data that is available for the region of interest (Hawaii). The GUI will allow alerts to be put in place, so the backend checks any new data to see if the conditions of the alert are satisfied, and if so, the user / creator of that alert will be notified. The GUI will also trigger the basic processing (mean / min / max) and visualization of the data which was already automatically ingested into the backend.

- Stretch Goals: A product that would allow the above functionality in addition to:
    - Ability for users to define an AOI and download all available quality-controlled ECOSTRESS data for further analysis
    - Develop the ability to scale this application to other (larger) areas (e.g. US Southwest fire conditions; Amazon rainforest canopy temperatures)
    - Incorporate ecologically relevant layers onto the platform, such as vegetation species and traits maps (available through collaborators in Hawaii), which are known to impact fire behavior.

This open-source product would be freely available to natural resource managers working in forestry, wildfire prevention, and coral reef conservation in Hawaii. It could also be a prototype that is later scaled to larger regions and other applications.

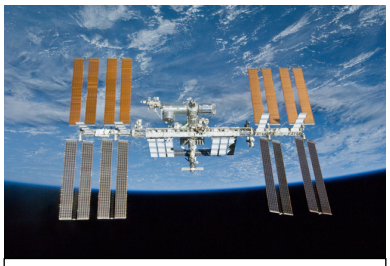
A note on not reinventing the wheel… To our knowledge there is no available tool providing all the outlined objectives. While examples of viewing near real time satellite thermal data exist (see climateviewer.org or worldview.earthdata.nasa.gov), these typically use MODIS, a sensor with ~1/200th the spatial resolution of ECOSTRESS, limiting the utility for many applications, especially in Hawaii, where many diverse ecosystems are packed into small islands. Additionally, available tools do not show anomalies from historical averages, don't allow users to define their AOIs, don't allow export timeseries or area stats, and don't have high-temperature alert systems in place. While we believe that a new webapp would allow for more flexibility and user functionality, it could be possible to collaborate with the creators of other tools to incorporate our intended functionality into their existing applications.

Knowledge, skills, and expertise required for this project:

- An understanding of raster data processing and interpretation
- Knowledge of modern Web2.0 programming techniques, languages, and frameworks.
- Skills in GUI design and evaluation.

Equipment Requirements:

- A cloud-based server will likely be required as a deployment platform. Development could be done on a free-tier server available from AWS, or use of NAU's Monsoon could be possible.
- No specialized software or equipment should be required for this project, beyond a standard software development stations and free IDEs, frameworks, and other tools.



The International Space Station is home to ECOSTRESS, the most sophisticated high-resolution thermal instrument currently providing data from space.
Source: NASA

Software and other Deliverables:

- A strong as-built report detailing the design and implementation of the product in a complete, clear and professional manner. This document should provide a strong basis for future development of the product.
- Complete professionally-documented codebase, delivered both as a repository in GitHub and as a physical archive on a USB drive.
- The web application as described above, deployed and tested successfully with real data.
- Must include a complete and clear tutorial for using the product.
- The created software and GUI must be openly available to and accessible by the scientific community (open-source).