


CS486C – Senior Capstone Design in Computer Science

Project Description

Project Title: AirFlow Processing Pipeline	
Sponsor Information: 	Trent Hare, Cartographer USGS Astrogeology thare@usgs.gov

Project Overview:

U.S. and international spacecraft missions have acquired enormous numbers of images and data about the planets and their satellites. While these data sets support furthering our knowledge of the universe, they are also used for supporting planetary rover missions, helicopter missions and the planned human exploration back to the Moon and Mars. The USGS Astrogeology Science Center (ASC) plays a critical role in providing these foundational data products for current and future missions. For example (Figure 1), a recent mosaic (called the orbital map in the figure) was created at ASC and will be used to safely land the Perseverance Rover onboard the Mars 2020 mission.

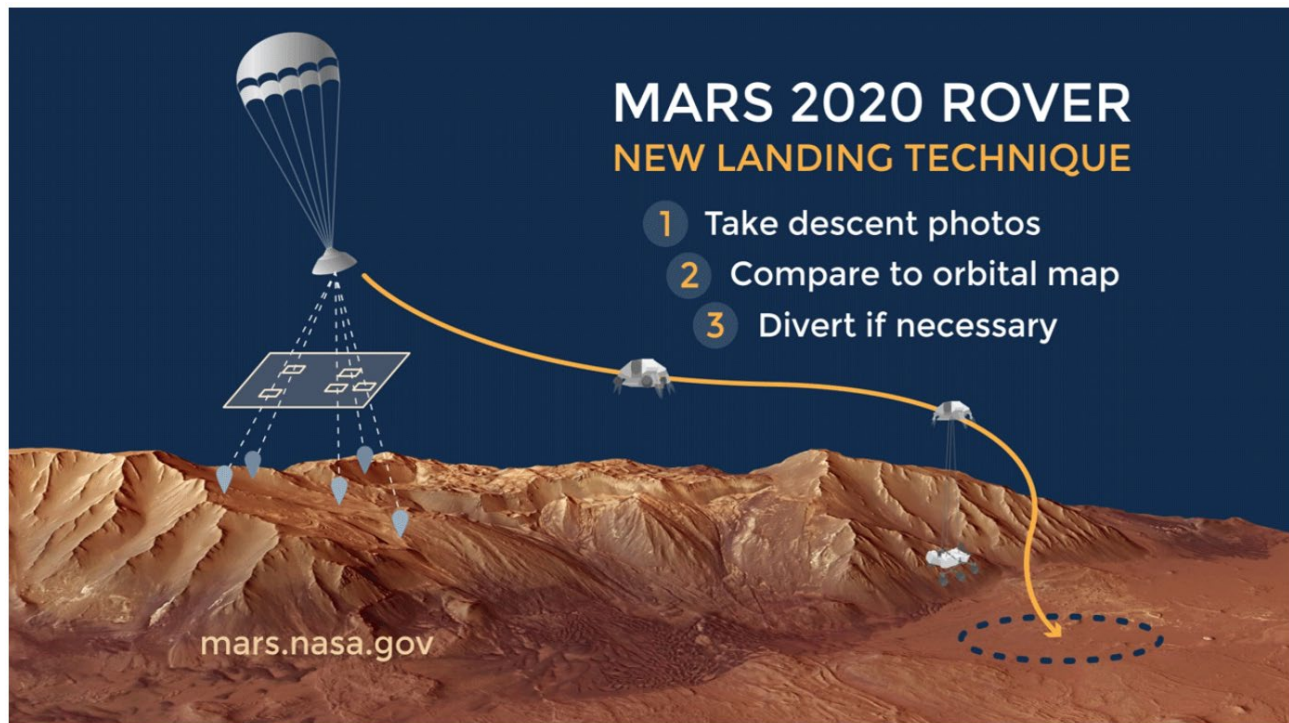


Figure 1. Showing the new Terrain-Relative Navigation to help land safely on Mars. For more see: <https://www.usgs.gov/news/mars-2020-mission-be-guided-usgs-astrogeology-maps>. Image: NASA

To help researchers and the public make similar mosaics, ASC already hosts an on-demand image processing pipeline for planetary images which allows images to be spatially registered to the surface (map projected) and converted to the user's preferred file format (<https://astrocloud.wr.usgs.gov>). However, this pipeline is currently locked into one path. To enable a much more valuable tool, we would like to see this pipeline highly configurable and allow researchers and the public to optimize the processing steps available in our planetary software suite.

Thus, the goal of this project is to develop a graphical workflow specification and monitoring tool; specifically, this tool will allow users to graphically specify a processing pipeline within the Apache

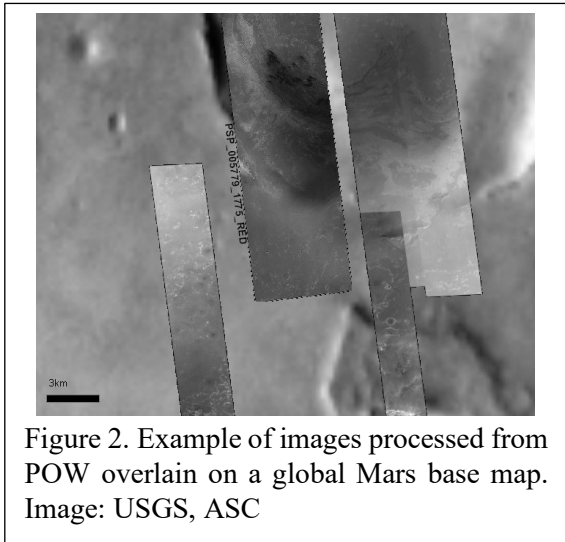


Figure 2. Example of images processed from POW overlain on a global Mars base map. Image: USGS, ASC

AirFlow framework (<https://airflow.apache.org>) with a customizable directed acyclic graph (DAG) that allows users to select/specify all processing steps in a pipeline. The current processing pipeline provided by the Map Projection on the Web (POW) service has a fixed set of steps, but this project would be to make the steps dynamic to complete processing up through any step as defined by the user. For example, people working with the images may only want the initial step of converting the files from the original mission format to the ISIS3 cube format or to have them converted and SPICE kernels applied which defines the spacecraft location and pointing information (<https://naif.jpl.nasa.gov/naif/spiceconcept.html>). An option for output will also be to support Cloud-optimized GeoTIFF (COG) with a SpatioTemporal

Asset Catalog (STAC) JSON record describing the output and a map viewer for viewing these records. Both writing COG and a preliminary STAC record is already supported but could be streamlined. Also connecting the images to an existing web map interface, based on the existing web-map interface, called CartoCosmo (a previous NAU Capstone), might also need small updates by this team.

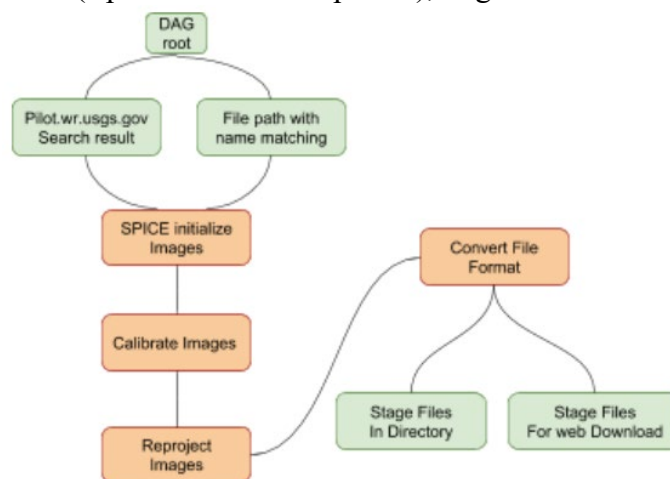


Figure 3. Example workflow for the DAG where the user starts with known images and selects the processing steps (orange). Often someone might want to update the parameters used in the “calibration” or they may want to even stop processing after calibration. Allowing researchers to create (and share) their own pipeline, by connecting these steps, could be extremely valuable.

Project Requirements

Summary - the goal of this project is to take the existing POW pipeline and break down the processing steps into separate AirFlow DAG processes to allow users full control over the steps that are run in their processing pipeline. The web application will provide a graphical drag-n-drop interface in which users can select desired processing steps, graphically arrange them into the desired processing workflow, then edit/configure each step with various parameters (if applicable). The capabilities this interface needs to support are:

- Specify the files to be processed either through a file path or from a CSV download from a Pilot search result (<https://pilot.wr.usgs.gov>). This is already solved.
- Select which of the processing steps in the POW pipeline to apply to the list of files.
- Specify the processing options for each step such as what map projection to use and what file format to convert the finished products to.
- Specify either an output directory (for USGS Astrogeology Science Center team members) or to have the finished files to be packaged for download using a cloud-based bucket (e.g., S3).
- Stretch: One form for output will be the Cloud-optimized GeoTIFF (COG) with a SpatioTemporal Asset Catalog (STAC) JSON record describing the files and connecting to the Leaflet map viewer for viewing these records.

Our team here at USGS can provide the existing POW solution, the steps we have taken to implement AirFlow for the processing, and a development environment for running the pipelines. We expect these overall specifications to become more precise as part of the early design and requirements process. The real work here is understanding how to connect these ideas together and creating the separate processing nodes via Apache Airflow. To restate this, the process capabilities are already available, but exposing these within Apache Airflow is the goal.

Knowledge, skills, and expertise required for this project

- Experience with Python will be essential as the POW pipeline and AirFlow are all written in Python.
- Some familiarity with the underlying architectures of distributed systems including processing clusters and Docker Container to deploy the environment.
- Data will likely be made available using cloud-hosting options like an Amazon S3 bucket. Fortunately, emulating an S3 bucket can be done using S3 Ninja or MinIO.

Equipment Requirements:

- We will provide access to the image processing environment for running the existing pipelines via Anaconda or Miniconda.

Software and other Deliverables:

Basic deliverables include (tagged as MVP, minimal viable product, or as stretch goals):

- MVP: AirFlow framework with an interface as described above. The interface can be an API, web tool, or other method the team comes up with.
- MVP: Design/Architecture documents demonstrating the system being run in as a single instance.

- MVP: 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.
- MVP: Complete professionally-documented codebase delivered as a repository in GitHub, BitBucket, or some other version control repository.
- Stretch: Export support for Cloud Optimized GeoTIFF (using the existing Geospatial Data Abstraction Library, GDAL) and STAC metadata file.
- Stretch: connect final output to Leaflet web-map interface (e.g., CartoCosmo).

References:

- Apache Airflow, <https://airflow.apache.org/>
- Hare, T.M., S.W. Akins, R.M. Sucharski, M.S. Bailen, and J.A. Anderson, 2013, Map Projection Web Service for PDS Images, LPSC XXXXIV. <https://www.lpi.usra.edu/meetings/lpsc2013/pdf/2068.pdf>
- STAC and Cloud Optimized TIFF:
- https://www.eclipse.org/community/eclipse_newsletter/2018/december/geotrellis.php
 - Github URL: <https://github.com/AustinSanders/PDS-Pipelines>