Guide: Using PGC GitHub: An Overview of PGC Tools
URL: https://www.pgc.umn.edu/guides/pgc-coding-and-utilities/using-pgc-github-an-overview-of-pgc-tools/
Last Modified: October 4, 2023
Export Date: November 7, 2026

Introduction

This is an overview of the tools that are offered on the Polar Geospatial Center (PGC) GitHub webpage. All tools are free to download and use. A total of 53 repositories are hosted by the PGC. However, only 11 repositories will be touched on in this overview due to relevance for PGC users. These programs are distributed without any warranty; without the implied warranty of merchantability or fitness for a particular purpose. Specific details pertaining to licenses can be accessed on each tool's GitHub repository.

The Polar Geospatial Center is an NSF-funded research organization at the University of Minnesota that provides geospatial support, mapping, and GIS/remote sensing solutions to researchers and logistics groups in the polar science community. More information about the PGC can be found here.

Requirements

A majority of the PGC tools can run within a personal computing environment (desktop computer) or in a cluster computing environment. The code is built to run primarily on a Linux HPC cluster running Maui/Torque for queue management. This tool will also work on a windows platform. Please note that the code is tightly coupled to the systems on which it was developed. You should have no expectation of it running on another system without some patching.

Software

The PGC tools designed to work with satellite imagery are built on the GDAL/OGR image processing API using Python. GDAL 2.1 is required for this tool to function.

If you are using a Linux system you will need to download the PGC optimized GDAL toolchain. The list of software installed with the optimized GDAL toolchain can be found here. A script is provided to install all required packages. If you have not ran a shell script in a Linux terminal follow this guide here. After installation of the PGC optimized GDAL toolchain, the mosaicking tool can be ran through the Linux terminal. There are plenty of free, online tutorials for Linux terminal if you are new to command line interfaces.

If you are using a Windows system, it is recommended that you use OSGeo4W. This will provide a Windows environment to use the tool. You can get the installers here. The express installation will provide the most high profile OSGeo4W packages. However, it will not allow for control over install location, proxies, and cache directory selection. The advanced install will allow for more control. The PGC mosaic tool will run with either install type. After installation of OSGeo4W, the orthorectification tool can be ran through the OSGeo4W Shell. As with Linux, there are numerous online resources for using a Windows command line interface.

Some of the tools are built to run using JavaScript in Google Earth Engine (GEE). GEE does not require any software download and is freely available.

Available Tools

Dockerfile: Miniconda Install Container

A number of the scripts created at the PGC for program installations use the Conda package, and some tools require other packages installed with Miniconda. This repository simply provides a way to install Miniconda.

PGC GitHub: Miniconda Install Container

Dockerfile: PGC Optimized GDAL Toolchain

Many of the tools used to work with satellite imagery and DEMs created from both SETSM and ASP require the installation of GDAL. The PGC has an optimized GDAL toolchain for download and installation in a Docker environment. This was built to run using Miniconda.

PGC GitHub: GDAL-docker *There is also a shell option for the GDAL download.*

JavaScript: Google Earth Engine Scripts

Google Earth Engine (GEE) is a web-based remote sensing application hosted by Google. It has access to all freely available imagery including: Landsat series, Sentinel series, NAIP, MODIS, etc. This program is open source, and only requires an account for access.

The PGC has created scripts for mosaicking Landsat 8 imagery over Alaska and the Himalayas. It uses the Top of Atmosphere (TOA) reflectance images to create a cloud-free composite. Input imagery spans multiple years to maximize cloud-free data points. Image appearance is modified so use for anything other than visualization and base mapping is inappropriate.

The geoms used to determine image output extent are also provided. Like the mosaics, the geometries are provided as a reference, and are not designed to represent cadastral boundaries.

PGC GitHub: GEE

JavaScript: PGC Coordinate Converter Webpage

The PGC provides a coordinate converter online. This tool can convert coordinates between decimal degrees (DD), degrees decimal minutes (DDM), and degrees minutes seconds (DMS). It also includes a map interface allowing for a point and click option to receiving coordinates in WGS84 Antarctic Polar Stereographic or WGS NSDIC Sea Ice Polar Stereographic North.
This repository is NOT an offline version of this coordinate converter. Instead, it is the source code to put the coordinate converter on another website. Feel free to use and modify the application as you see fit.

PGC GitHub: Coordinate Converter

JavaScript: PGC Info Webpage

The PGC Information Pages is a stand-alone web application that displays a continuous slide of selected content, much like a slideshow. It runs in a web browser, but meant for display on a High-Definition (HD) display, such as a television or computer screen. The application is somewhat responsive to different screen resolutions, but is optimized for 16:9 displays.

This application can be run through the URL. This will always the latest release, and it will not display your location. Instead it will display St. Paul, Minnesota.

There is an option for you to host the application on your own web server. This requires you to download the latest release. Instructions on how to edit the location are at the bottom of the application download page.

PGC GitHub: Info Pages App

Shell: PGC Tuned – Ames Stereo Pipeline

Ames Stereo Pipeline is a 3-Dimensional surface reconstruction tool developed by the Intelligent Robotics Group (IRG) at the National Aeronautics and Space Administration (NASA). It was originally developed for the Mars Global Surveyor mission. This tool has advanced stereo image processing capabilities, and the PGC uses this tool to create stereo DEMs. The PGC has created both a bash script and Dockerfile to install the Ames Stereo Pipeline tool. This uses a list of installers to build the ASP toolset. A warning from the ASP developers: ASP is a research product and may contain bugs or incomplete features.

PGC GitHub: Ames Stereo Pipeline

Additionally, the ASP documentation offers a tutorial on how to process Maxar imagery. This can be accessed here. This documentation also includes citation instructions and in-depth documentation of the tool itself.

Python: ArcGIS Server Feature Service Query

The PGC has a number of web mapping services offered in ArcGIS Desktop, ArcGIS Pro and QGIS. This repository is a standalone python script for querying the ArcGIS Server Feature Service. This will allow users to access the Maxar Stereo Index (pairnames). This will require authentication via an approved PGC account.

PGC GitHub: ArcGIS Server Feature Service Query

Python: Mosaic Dataset Configuration Script

Mosaic datasets allow users to manage, view, and query collections of image data. For more information on mosaic datasets visit the ArcGIS help page. This repository hosts a Python script that takes in parameters and an xml file to perform various tasks to create and populate a mosaic dataset. This tool has a number of features including:

- Automate the creation of multiple mosaic datasets
- Configuration of multiple mosaic datasets using configurations files
- Built-in verbose reporting and logging system
- Command line usage via batch files

PGC GitHub: Mosaic Dataset Configuration Script

Python:PGC DEM Tools

The PGC DEM tools are a bundle of 11 tools for indexing, shelving, copying, and modifying both SETSM and ASP DEMs. These tools can do:

- Build a geodatabase or shapefile index of a single SETSM DEM or directory of DEMs
- Build a geodatabase or shapefile index of a single SETSM DEM mosaic tile or directory of tiles
- Package all auxiliary files into a tar.gz archive. This can be done on DEMs or DEM mosaic tiles
- Move or copy DEMs by date, and create a representative file structure in the output directory. The file structure is ///.
- Move or copy a SETSM DEM or a directory of DEMS into folders based on the geocell that intersects the DEM centroid. Identified by the lower left geocell center
- Move or copy a SETSM DEM or a directory of DEMs into folders based on a custom shapefile index. The

files will be shelved based on the centroid location relative to the shapefile index
- Move or copy ASP deliverable files
- Apply the offset and outputs a new raster if there is GCP registration information included with the SETSM DEM. The strip ArcticDEM files contain IceSAT altimetry offset information within the metadata, and this tools allows for the adjustment of SETSM DEMs with that information
- Resample a SETSM DEM to a lower resolution
- Divide SETSM mosaics into subtiles

PGC GitHub: PGC DEM Tools

Python: PGC Imagery Utilities

The PGC Imagery Utilities is a collection of tools used at the PGC for satellite imagery manipulation. These tools can be used for batch processing and include:

Orthorectification

Correct for terrain displacement and radiometric settings. This tool can also change bit depth of the imagery. Requires the use of a DEM.

Mosaicking

Takes multiple input images and mosaics them into a set of non-overlapping output tile images. It can sort images according to cloud cover, sun elevation angle, off-nadir angle, exposure, and proximity to a date.

Pansharpening

Pansharpening is a practice of using the panchromatic band of a satellite image to increase the spatial resolution of the image. This tool applies the orthorectification process followed by pansharpening using the GDAL tool gdal_pansharpen.

Normalized Difference Vegetation Index (NDVI)

NDVI is a spectral index used to identify areas of vegetation. This index uses the near infrared and red bands of a satellite image. This tool is designed to run on imagery that has already been processed using the orthorectification utility.

Each of the tools hosted in the PGC Imagery Utilities repository have detailed guides for their use. These guides can be accessed through the PGC website.

PGC GitHub:PGC Imagery Utilities

Python: SETSM Post-Processing

This utility offers methods for filtering and merging DEMs produced by SETSM with stereo pairs of Maxar imagery. The end result is the creation of strips from individual scenes. The code for filtering and mosaicking is largely from the MATLAB code written by Ian Howat, Professor of Earth Sciences at Ohio State University.

The readme file in the GitHub repository offers an in-depth explanation of the dependencies, installation recommendations, description on how to use the script, and how the tool works.

PGC GitHub: SETSM Post-Processing

Other Potentially Useful GitHub Repositories

COMNAP Antarctic Vessels

The Council of Managers of National Antarctic Programs (COMNAP) creates a list of Antarctic vessels administered by and for National Antarctic Programs. This repository hosts a comprehensive list of vessels operating below 60oS latitude for science, science support, and logistics purposes. This dataset was provided by each National Antarctic Program to COMNAP.
PGC GitHub: COMNAP Vessels

COMNAP Antarctic Facilities

The COMNAP Antarctic Facilities is a comprehensive list of Antarctic facilities with a status of ‘Open’ or ‘Temporarily Closed’. This repository hosts a comprehensive list of Antarctic facilities, and was provided by each National Antarctic Program to COMNAP.
PGC GitHub: COMNAP Facilities