Copernicus FICE 2025

Training on
In situ Ocean Colour Above-Water Radiometry towards Satellite Validation

HyperCP Setup Instructions

Dirk Aurin
NASA Goddard Space Flight Center/Morgan State University
dirk.a.aurin@nasa.gov

July 7-19, 2025
Venice, Italy

PROGRAMME OF THE EUROPEAN UNION
Copernicus Europe's eyes on Earth
IMPLEMENTED BY EUMETSAT
FRM4SOC Phase-2
fiducial reference measurements for satellite ocean colour
CNR ISMAR
ISTITUTO DI SCIENZE MARINE
NASA

HyperCP Setup Instructions

Dirk Aurin
NASA/GSFC

Git

- Git is an open-source software revision control system that allows developers to coordinate on projects and end users to stay up to date with the latest revisions

- GitHub is a browser-based staging area for git-managed software (referred to as repositories in git). Using git, repositories can be cloned to your local computer.

- Command line git comes installed on Linux and macOS but needs to be installed on Windows (downloads available at https://git-scm.com/)

- When using git, software is easily kept up to date using the “pull” command.

- If git cannot be installed or used for any reason, the software repository can be downloaded (rather than cloned) from GitHub with all code (open source) or as a bundled package release with an executable file and environment.

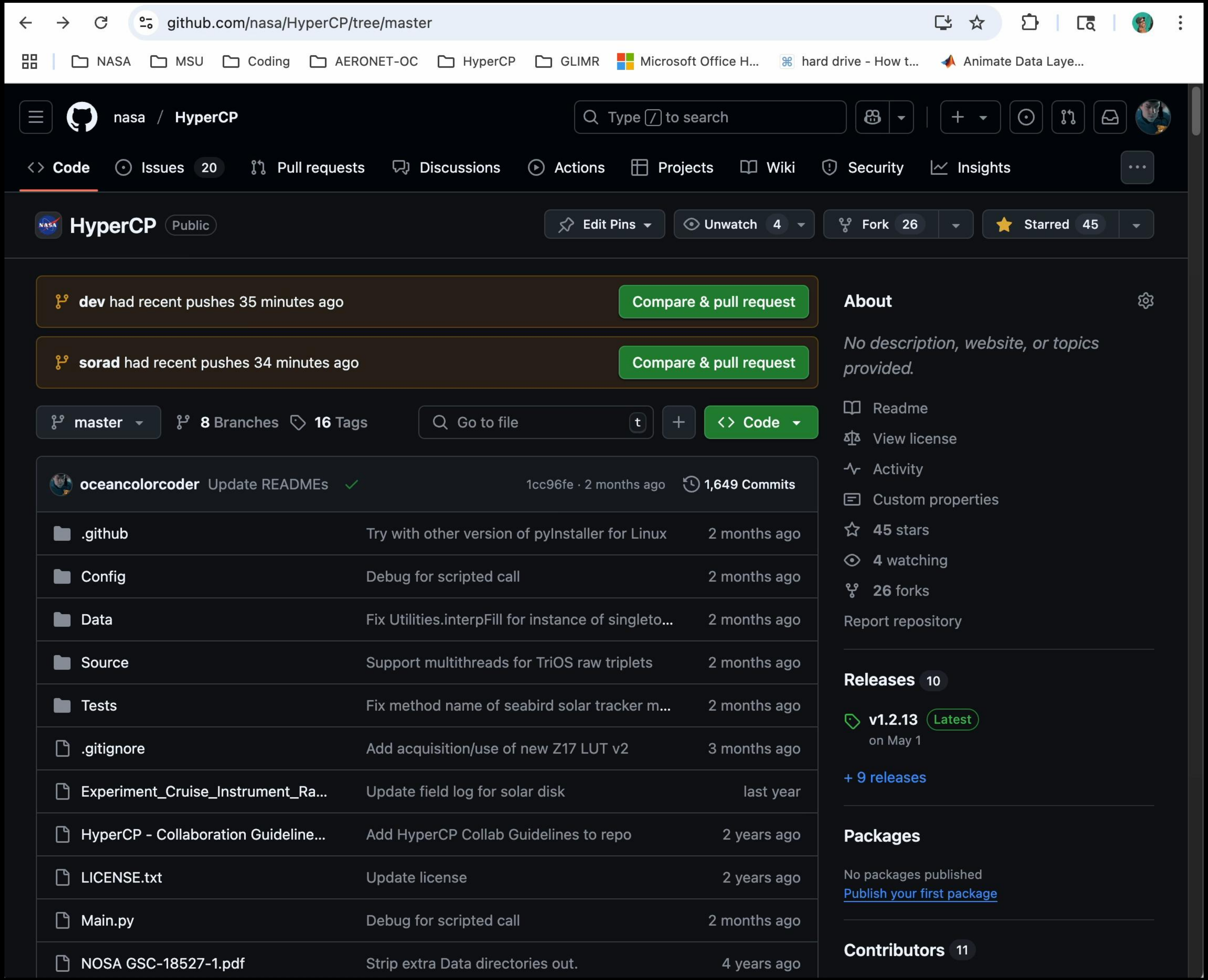
Python and Conda

- Python is an open-source programming language that has rapidly grown in popularity in recent years thanks to its clarity, accessibility, and flexibility. It supports object-oriented computing (HyperCP is object-oriented)

- HyperCP is written in Python so you will need it installed. Linux and macOS come with Python installed by default, but Windows will require you install it.

- There are many ways to install Python but we recommend using a package manager. Anaconda (https://www.anaconda.com/) is now license-based, but conda-forge (https://conda-forge.org/download/) is not.
  - Configuring your Python environment to run HyperCP will be difficult without a conda. The bundled

Cloning, download, or package release



github.com/nasa/HyperCP/tree/master

NASA · MSU · Coding · AERONET-OC · HyperCP · GLIMR · Microsoft Office H... · hard drive - How t... · Animate Data Laye...

nasa / HyperCP

Type / to search

Code · Issues 20 · Pull requests · Discussions · Actions · Projects · Wiki · Security · Insights

HyperCP Public

Edit Pins · Unwatch 4 · Fork 26 · Starred 45

dev had recent pushes 35 minutes ago — Compare & pull request

sorad had recent pushes 34 minutes ago — Compare & pull request

master · 8 Branches · 16 Tags · Go to file · t · + · Code

oceancolorcoder Update READMEs ✓ · 1cc96fe · 2 months ago · 1,649 Commits

.github	Try with other version of pyInstaller for Linux	2 months ago
Config	Debug for scripted call	2 months ago
Data	Fix Utilities.interpFill for instance of singleto...	2 months ago
Source	Support multithreads for TriOS raw triplets	2 months ago
Tests	Fix method name of seabird solar tracker m...	2 months ago
.gitignore	Add acquisition/use of new Z17 LUT v2	3 months ago
Experiment_Cruise_Instrument_Ra...	Update field log for solar disk	last year
HyperCP - Collaboration Guideline...	Add HyperCP Collab Guidelines to repo	2 years ago
LICENSE.txt	Update license	2 years ago
Main.py	Debug for scripted call	2 months ago
NOSA GSC-18527-1.pdf	Strip extra Data directories out.	4 years ago

About

No description, website, or topics provided.

- Readme
- View license
- Activity
- Custom properties
- 45 stars
- 4 watching
- 26 forks

Report repository

Releases 10

v1.2.13 Latest
on May 1

+ 9 releases

Packages

No packages published
Publish your first package

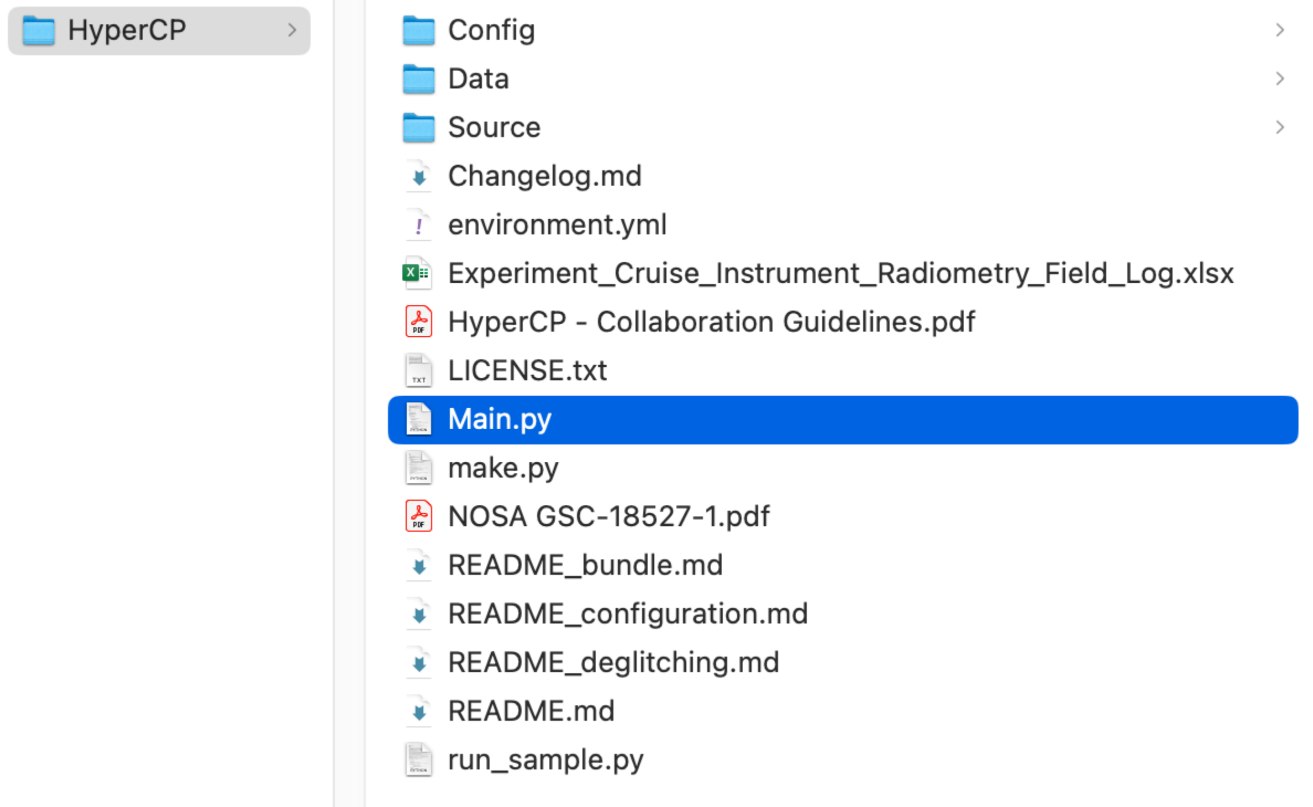
Contributors 11

Setting Up HyperCP

- Once you have git and Python set up, you are ready to clone HyperCP and initialize a dedicated environment in which to run it

- From a directory where you want HyperCP installed (it will create a subdirectory of that name) type:
  - *git clone --depth 1 https://github.com/nasa/HyperCP.git*
  - the remote repository is thereby established and named “origin” by default (see *git remote –v* for details)
  - Without git, you can download a Zip file from https://github.com/nasa/HyperCP under the Code button
  - Without git or conda, the bundled release will work.

- HyperCP will now consist of a directory structure:



```
HyperCP
    Config
    Data
    Source
    Changelog.md
    environment.yml
    Experiment_Cruise_Instrument_Radiometry_Field_Log.xlsx
    HyperCP - Collaboration Guidelines.pdf
    LICENSE.txt
    Main.py
    make.py
    NOSA GSC-18527-1.pdf
    README_bundle.md
    README_configuration.md
    README_deglitching.md
    README.md
    run_sample.py
```

Git Remotes and Branches

- Remotes are repositories on servers (in contrast to local repos on your machine) such as GitHub or GitLab. Each remote has a name and address.

- Branches are copies of the repository used to aid in development. The most stable recent branch is typically called “master” or “main”. Branches such as “dev” are used in development.

- By default, your new local repository should have a remote (“origin”) and and one branch (“master”)

- To make your life easier when pulling updates from the NASA repository on GitHub, you can assign the default remote and branch to use with *pull* and *push* commands:
  - *git push --set-upstream [remote_name] [branch_name]*
  - …after which you can simply type *“git pull”* to get updates before using HyperCP each time

Setting Up HyperCP

- To initialize a conda environment for HyperCP (a place where all necessary software dependencies are sourced), from the terminal type:
  - *conda env create -f environment.yml*
  - Follow the prompts. This could take several minutes.

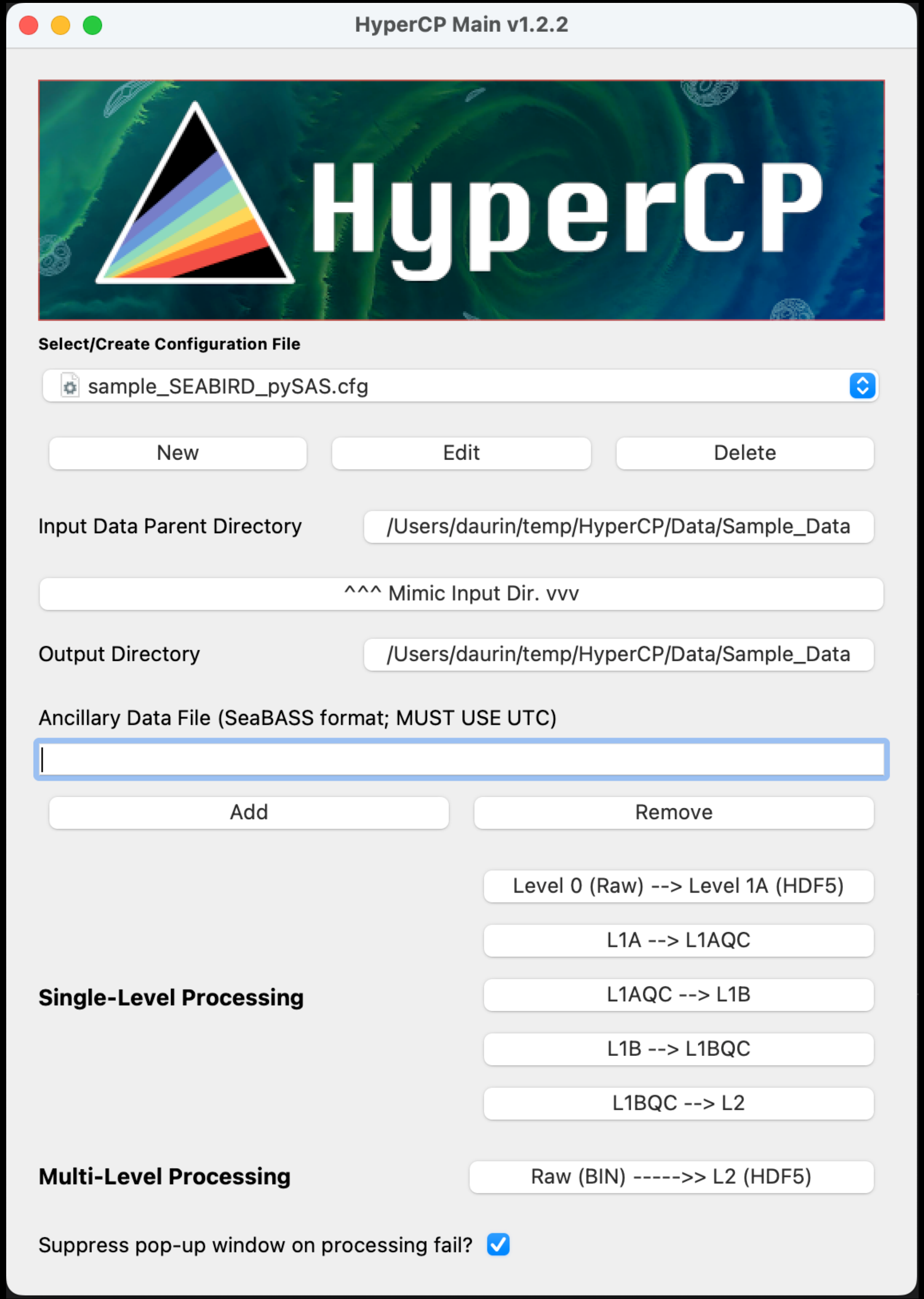
- This will automatically activate the new environment, but you will need to activate it again later when opening a new terminal before launching HyperCP. Type:
  - *conda activate hypercp*

- To update HyperCP with the latest version in the master branch (recommended before each use), type:
  - *git pull*

Setting Up HyperCP

- The final step to initialize HyperCP after installation is the download of additional databases and look-up-tables (LUTs) totally ~3.2 GB. (If these are provided to you locally, you can copy them to the HyperCP/Data directory. Otherwise, the first time HyperCP is launched, they will download automatically.)

- To launch HyperCP, navigate to the HyperCP directory and type:
  - *python Main.py*



HyperCP Main v1.2.2

Select/Create Configuration File

sample_SEABIRD_pySAS.cfg

Input Data Parent Directory /Users/daurin/temp/HyperCP/Data/Sample_Data

Output Directory /Users/daurin/temp/HyperCP/Data/Sample_Data

Ancillary Data File (SeaBASS format; MUST USE UTC)

Single-Level Processing

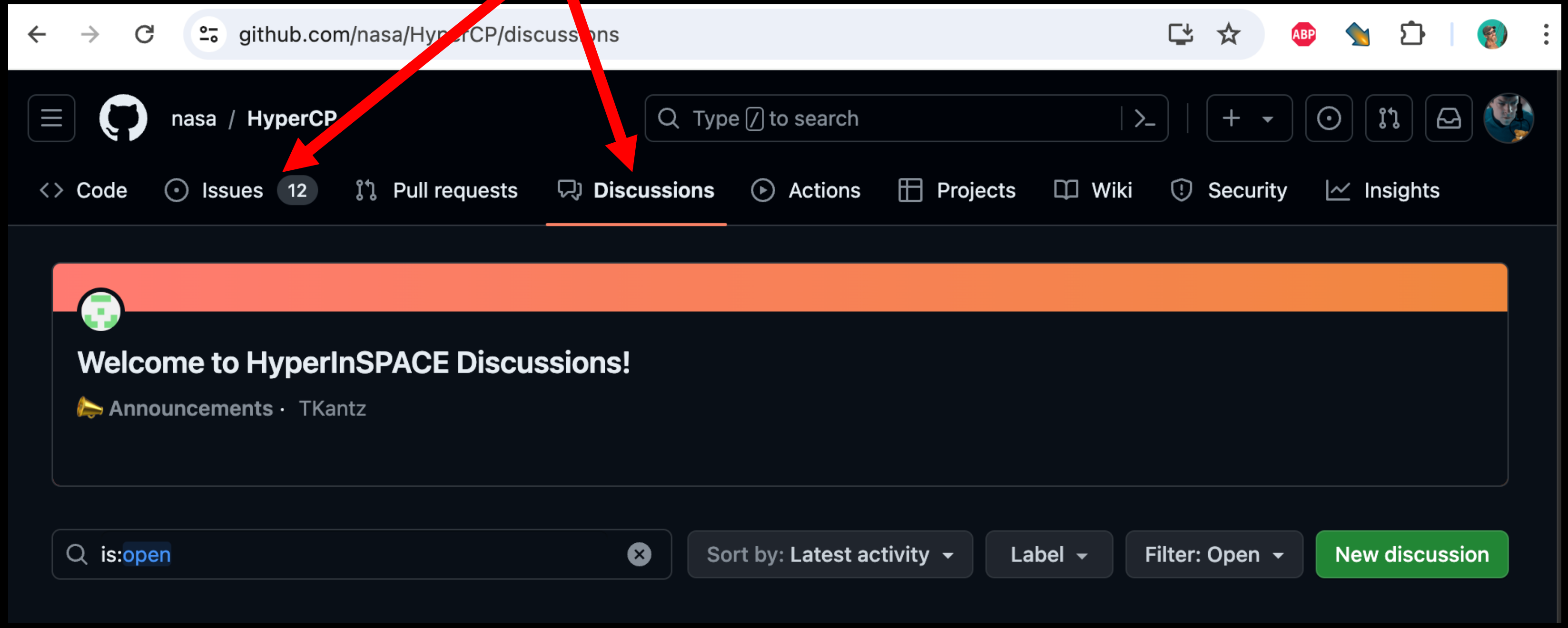
Multi-Level Processing

Suppress pop-up window on processing fail? ☑

End HyperCP Setup

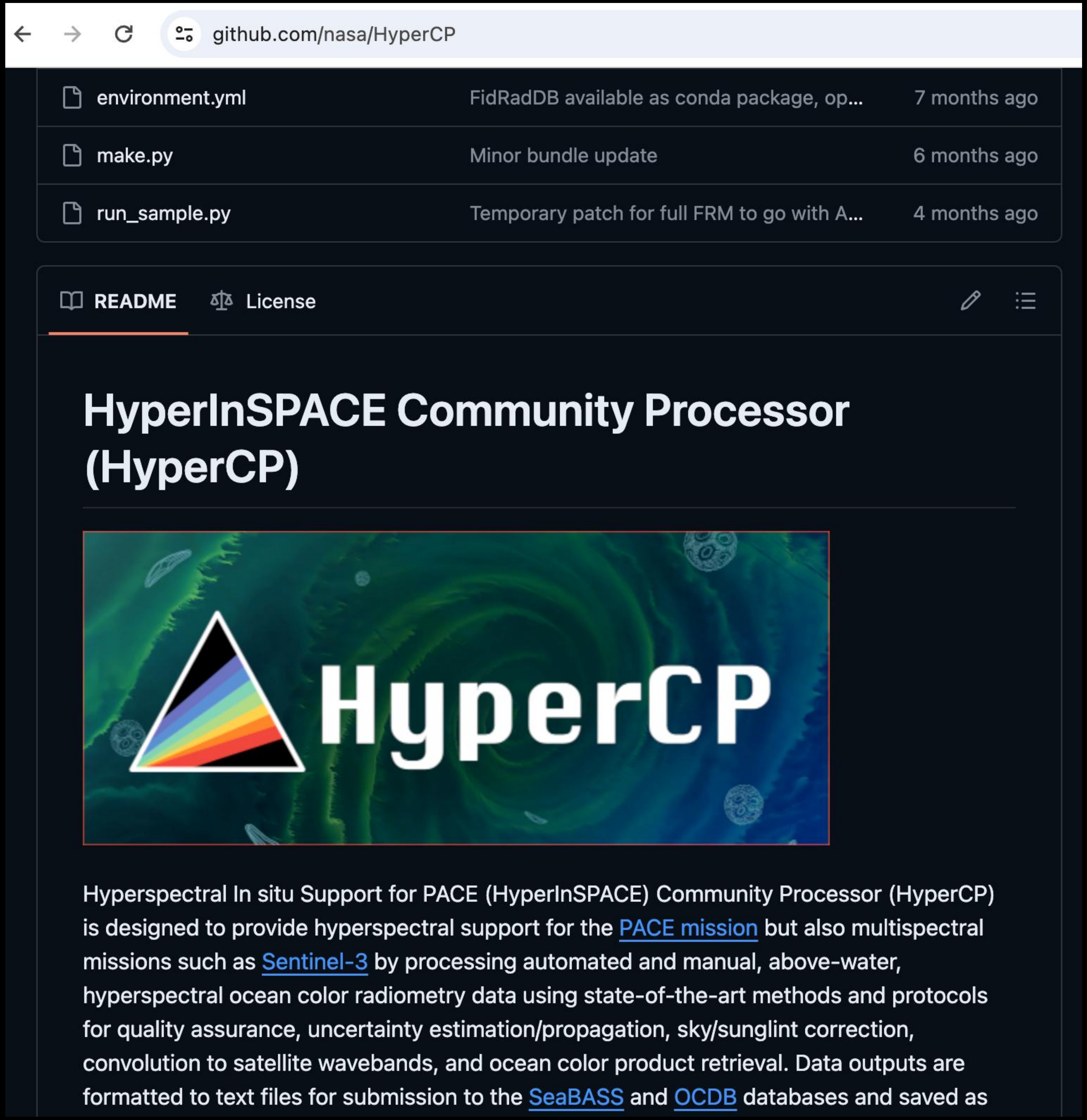
These instructions are available for later reference in the README.md at https://github.com/nasa/HyperCP

For support, please reach out on GitHub in the Discussions tab, or, for bugs, the Issues tab.



Welcome to HyperInSPACE Discussions!

Announcements · TKantz



environment.yml	FidRadDB available as conda package, op...	7 months ago
make.py	Minor bundle update	6 months ago
run_sample.py	Temporary patch for full FRM to go with A...	4 months ago

HyperInSPACE Community Processor (HyperCP)

Hyperspectral In situ Support for PACE (HyperInSPACE) Community Processor (HyperCP) is designed to provide hyperspectral support for the PACE mission but also multispectral missions such as Sentinel-3 by processing automated and manual, above-water, hyperspectral ocean color radiometry data using state-of-the-art methods and protocols for quality assurance, uncertainty estimation/propagation, sky/sunglint correction, convolution to satellite wavebands, and ocean color product retrieval. Data outputs are formatted to text files for submission to the SeaBASS and OCDB databases and saved as