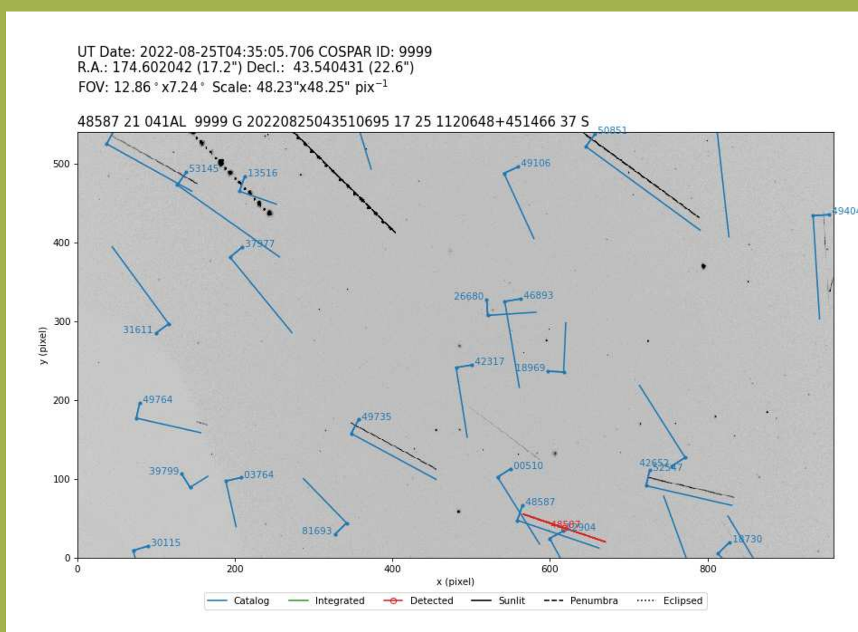
SatNOGS Optical

Unofficial

Guide



by

Jeff Moe

SatNOGS Optical Unofficial Guide

by Jeff Moe

Published by Jeff Moe, Loveland, Colorado, USA.

20220827

Contents

List of Figures

Introduction

SatNOGS Optical

Introduction

The SatNOGS Optical Unofficial Guide documents how to set up and run an optical ground station for the Libre Space Foundation's SatNOGS network. The Libre Space Foundation supports "free and accessible space for all, creating Open Source space technologies."[1]

More rocks in space.[2]

Process Overview

SatNOGS Optical Process Overview.[3] See figure 1, page viii.

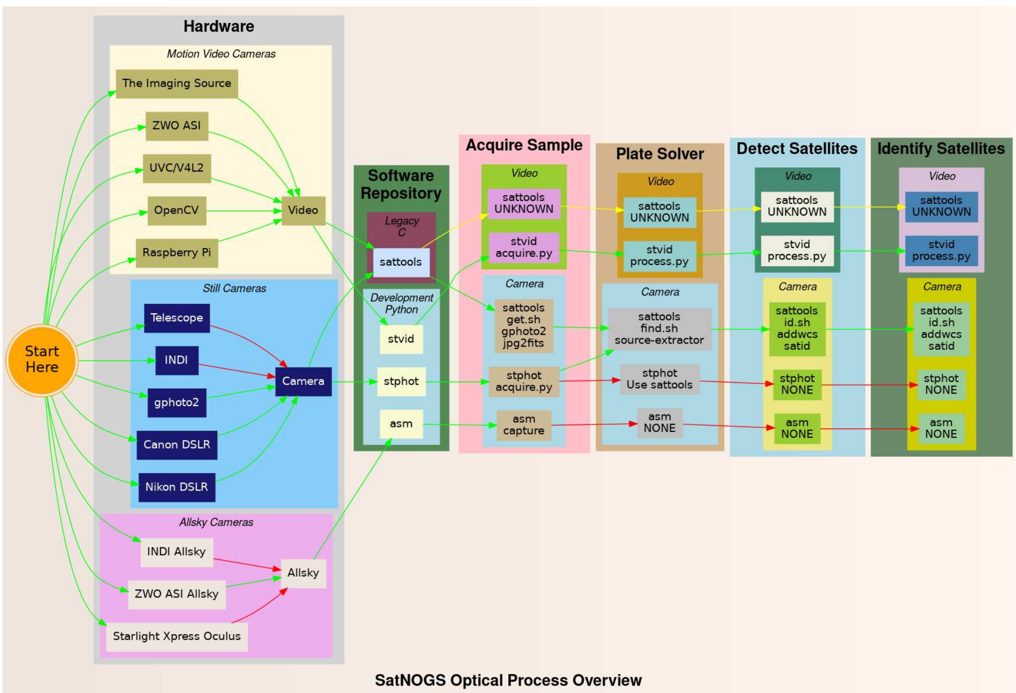


Figure 1: SatNOGS Optical Process Overview

Run `acquire.py`, then `process_new.py`. But don't run rm -rf /*.

[1] https://libre.space

[2] https://starlink.com

[3] https://spacecruft.org/spacecruft/SNOPO

Satellites

Artificial Satellites in Space

1.1 Overview of Satellites

Artificial satellites are rocks with modems.

Ground Stations

Ground Stations on Earth

2.1 Overview of Ground Stations

Ground Stations.

2.1.1 SatNOGS Ground Stations

The Libre Space Foundation built the SatNOGS network using distributed, volunteer-run, radio frequency ground stations, such as in figure 2.1, page 12.[1] It shows a SatNOGS ground station with VHF (right) and UHF (left) antennas on a mast with Alt/Az rotator.

Figure 2.1: SatNOGS ground station with antennas.

2.1.2 SatNOGS Optical Ground Stations

The Libre Space Foundation is developing a system to add optical ground stations to the distributed network.

Prototype optical ground stations are being developed. An example setup, using a Skywatcher EQ6-R Pro telescope tripod and tracking mount, can be seen in Figure 2.2, page 13.

[1] https://spacecruft.org/assets/i/spacecruft-rotator.png

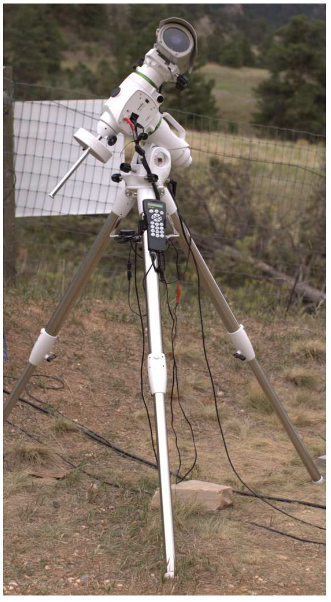


Figure 2.2: SatNOGS Optical ground station prototype.

Hardware

Design and Set Up Equipment

3.1 Overview of Hardware

Hardware considerations for a SatNOGS Optical Ground Station.

Main hardware components in an optical ground station:

- Lens.
- Camera.
- Computer.

Other components:

- Ethernet cable.
- USB cable.
- Enclosure
- Power supply.
- Tripod.
- Manual or tracking mount.
- Power source, grid or alternative.
- Internet, wifi or ethernet.

3.2 Camera

Cameras being tested:

- The Imaging Source IMX174 based.
- ZWO ASI IMC174 based mono.
- DSLR.

> NOTICE:
> ZWO/ASI cameras require proprietary non-libre software on host computer.

3.3 Lenses

Lenses being tested:

- Kowa 50mm f1.4 C-mount.
- Canon EF 50mm f1.2 USM.

3.4 Embedded Computer

Computers, such as Raspberry Pi, that can be used.

- Odroid N2 — Confirmed working.
- Odroid M1 — Testing.
- Raspberry Pi 3 — ?
- Raspberry Pi 4 — ?
- Intel NUC — ?

3.4.1 Comparison

Comparing embedded computers for SatNOGS Optical.

Make	Model	Architecture	Max RAM	eMMC
Odroid	N2	ARM64	4 GB	Yes
Odroid	M2	ARM64	8 GB	Yes

3.5 Example Tracking Ground Station

This is an example of a tracking ground station. It is a prototype, so there are lots of mis-matched, overbuilt/underbuilt parts. The full setup on tripod, can be seen in 2.2, page 13.

A close up of the setup can be seen at 3.1, page 18, showing the Skywatcher telescope tracking mount, a Bosch PoE camera enclosure, and through the glass the camera lens. In the background is a white antenna for GNSS (GPS) and a solar power setup.

The camera lens is protected by the enclosure glass, which is IP67 (XXX) rated. See a close up of the front of the enclosure and camera lens in figure 3.2, page 18.

As seen in figure 3.3, page 19, the left side of the enclosure has a hinge for opening. The bottom white component is part of the telescope mount.

Figure 3.4, page 19, shows the right side of the enclosure. Two mounting bolt access points can be seen on each end. These are unscrewed with a hex head tool (supplied) to open the enclosure.

The enclosure is opened from the right side, as shown in figure 3.4, page 19.

Inside the camera enclosure, as shown in Figure 3.6, page 20, is:

- The Imaging Source DMX camera with Sony IMX174 CMOS.
- Kowa 50mm f1.4 C-mount lens.
- Odroid N2 running Debian.
- Odroid N2 plastic enclosure, large half, hole drilled for ad-hoc mounting.

Figure 3.1: SatNOGS Optical ground station prototype.

Figure 3.2: Camera enclosure, front side, showing glass and lens.

Figure 3.3: Camera enclosure, left side, showing hinge.

Figure 3.4: Camera enclosure, right side.

- Blower fan on top, with power cable (came with Bosch enclosure).
- Fan, maybe not so useful, with power cable (came with Bosch enclosure).
- Camera mounting plate (came with Bosch enclosure).
- Camera mounting screws, M6x25 (?).
- Ethernet cable, internal, short white (came with Bosch enclosure).
- PoE ethernet cable, external, plugged into PoE switch for data and power.
- USB 3 cable, internal, way too long, needs replacing, from Odroid to camera. XXX flat connector
- USB 3 cable, external, from Odroid to telescope mount. XXX large rectangle connector
- “Custom” 12V DC power cable from Bosch PoE to Odroid.

Figure 3.5: Camera enclosure, right side, opened.

- Assorted nuts, bolts, and washers for an ad-hoc standoff height.

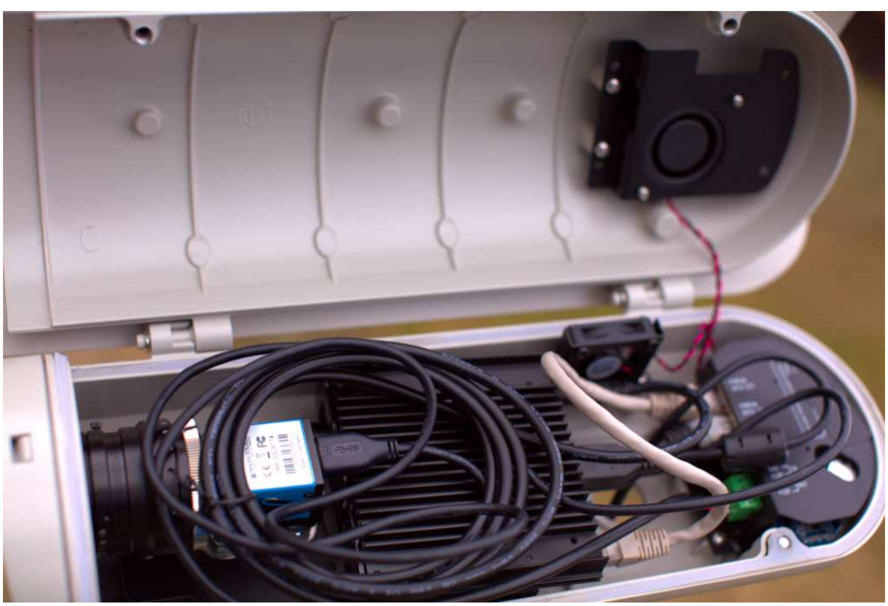


Figure 3.6: Camera enclosure, opened.

The top of the enclosure shows weather protection and a sun visor. See figure 3.7, page 21.

Figure 3.7: Camera enclosure, top.

Setup

Install and Configure Software

4.1 Setup Optical Ground Station Software

HOWTO set up and configure a SatNOGS Optical ground station.

Setup an embedded computer, such as an Odroid N2, with Debian stable (11/Bullseye) or testing (Bookworm).

See each repository for latest documentation.

Install dependencies from Debian repository:

```
sudo apt update
sudo apt install python3-virtualenv python3-pip python3-dev \
    source-extractor astrometry.net \
    libeigen3-dev giza-dev libx11-dev \
    astrometry-data-tycho2 astrometry-data-tycho2-07 \
    astrometry-data-tycho2-08 astrometry-data-tycho2-09 \
    astrometry-data-tycho2-10-19
```

Next, install the `hough3d-code` dependency.

```
git clone https://gitlab.com/pierros/hough3d-code
cd hough3d-code
make all
sudo cp -p hough3dlines /usr/local/bin/hough3dlines
```

Install `satpredict` from using either the cbassa or spacecruft repo.

```
cd ../
git clone https://spacecruft.org/spacecruft/satpredict
# Or
#git clone https://github.com/cbassa/satpredict
cd satpredict/
make
sudo make install
```

Now install `stvid`, the main acquisition and processing application. It is written in Python. Either use the spacecruft `git` repository or the cbassa one.

```
cd ../
git clone https://spacecruft.org/spacecruft/stvid
# Or use upstream:
#git clone https://github.com/cbassa/stvid
```

Optionally, set up a Python virtual environment:

```
cd stvid
virtualenv -p python3 env
source env/bin/activate
pip install --upgrade pip
```

Install dependencies. Numpy needs to be installed first, or the install fails.

```
pip install --upgrade numpy
pip install --upgrade -r requirements.txt
```

If the system `python3-numpy` conflicts, you could try installing numpy in the virtualenv thusly:

```
pip install --upgrade --ignore-installed numpy
```

Debian in bug `#941466` changed the name of the `sextractor` dependency to `source-extractor`, so a symlink is needed for now:

```
sudo ln -s /usr/bin/source-extractor /usr/local/bin/sextractor
```

4.2 Configure

Configure the embedded computer.

```
cd stvid/
cp -p configuration.ini-dist configuration.ini
# Edit
# Note for fields st-username and st-password, *don't* use quotes.
# Set:
vi configuration.ini
```

Edit various `configuration.ini` fields:

```
# For example:
observer_cospar = 9999         # COSPAR number of observer's site
observer_name = Jeff Moe       # Name of observer
observer_lat = 40.568912       # Latitude of location in decimal degrees
observer_lon = -105.225852     # Longitude of location in decimal degrees
observer_height = 1860          # Elevation of location in meters

observations_path = ./data/obs # Path for raw acquisitions
tle_path = ./data/tle          # Path where TLEs are stored (and updated)
results_path = ./data/results   # Path for results of processed obs

st-username = foo
st-password = bar

# Path to source-extractor
sex_config = /usr/share/source-extractor/default.sex
```

Store downloads here:

```
mkdir -p ./data/obs ./data/tle ./data/results
```

The `ST_DATADIR` variable is still used, from `sattools`. Maybe set all these in `~/.bashrc` (suit to fit directory setup):

```
export ST_DATADIR=$HOME/sattools
export ST_TLEDIR=$HOME/TLE
export ST_OBSDIR=$HOME/satobs
export ST_LOGIN="identity=foo@no:pass"
```

Set `astrometry.net` to run in parallel, assuming you have enough RAM: (This doesn't appear to work? Breaks?).

```
sudo vim /etc/astrometry.cfg
# Uncomment:
inparallel
```

Operation

HOWTO Run a Station

5.1 Overview of Operation

HOWTO run a SatNOGS Optical ground station, after it has been set up and congfigured.

5.2 Acquire Data with **stvid**

Acquire data with **acquire.py** from **stvid**. Enter **virtualenv**, if used:

```
cd stvid/
source env/bin/activate
```

Get the latest TLEs:

```
./update_tle.py
```

Acquire live data from camera.

```
cd stvid/
./acquire.py
```

Figure 5.1: Satellite images acquired by stvid.

5.3 Process Data

Process data with using stvid. Adjust dated directory to actual directory name.

```
cd stvid/
./process_new.py -c config_new.ini -d data/obs/20220825_1/020502/
```

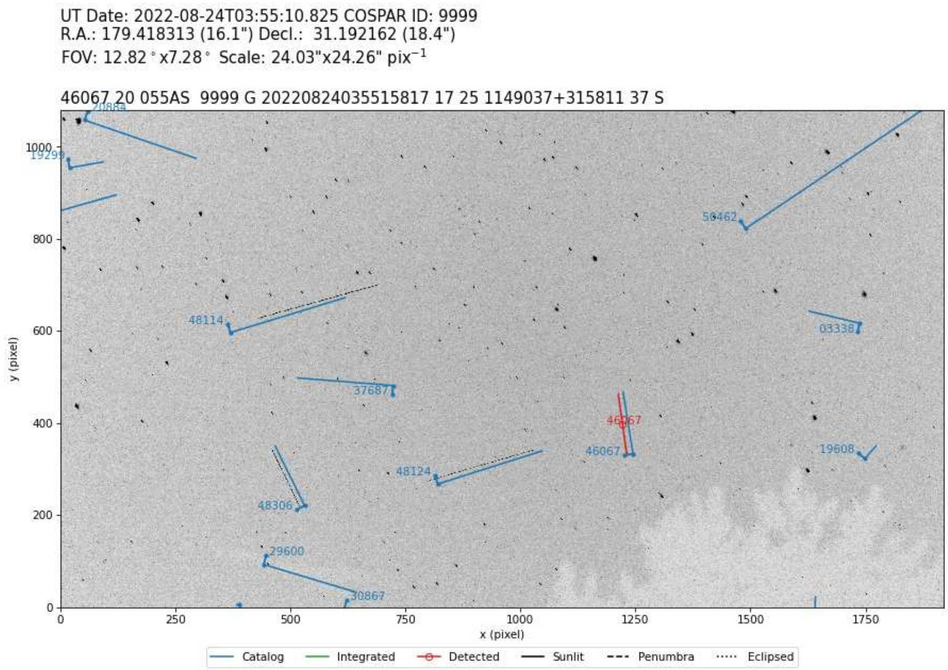


Figure 5.2: Satellite images processed by stvid.

5.4 Upload Data

Push. The SatNOGS network isn't ready for optical data yet.

Analysis

More Than Just Starlink and Old Rocket Bodies

6.1 Overview of Analysis

Analyze data with applications, such as AstroImageJ. AstroImageJ can read FITS format files. See 6.1, page 32, for a screenshot displaying a FITS file generated by **`stvid`**.

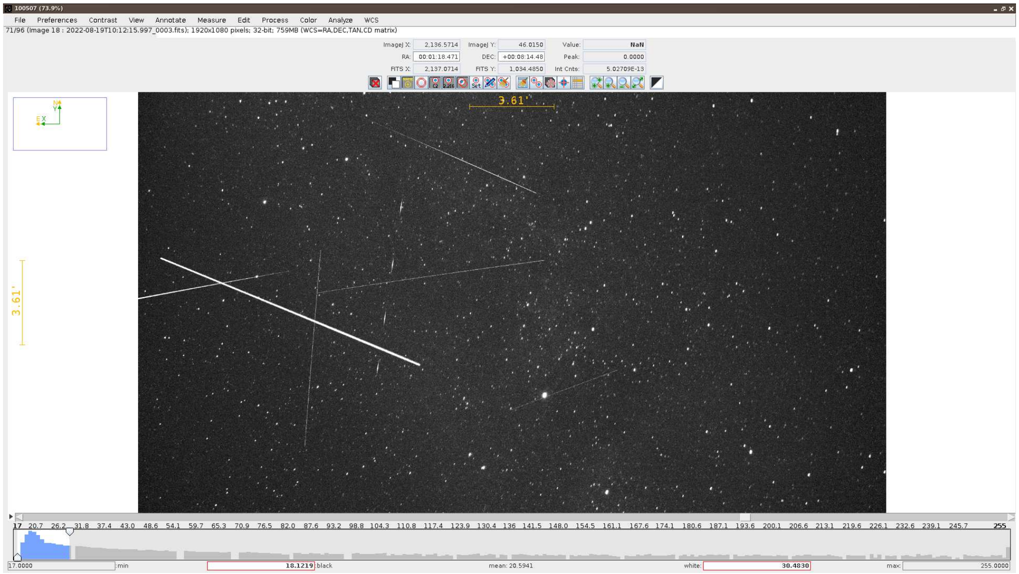


Figure 6.1: AstroImageJ screenshot viewing FITS file.

Support

Help!

7.1 Support

How to get help.

- **`#satnogs-optical`** channel in Matrix — `#satnogs-optical:matrix.org`
- Libre Space Foundation Community Forum — `https://community.libre.space`

Contact

Email, Chat, Forum

8.1 Contact

Jeff Moe

Username: `jebba`
Email: moe@spacecruft.org

Index

Glossary

Artificial satellite Satellite launched by teh hoomans.

Colophon

Created with 100% Free Software

Debian GNU/Linux

LaTeX