Working with the JeAn - Jet Analyzer package (v.3.1)

Alexey G. Stupishin

agstup@yandex.ru

2020-2022, St.Petersburg, Russia

Getting Started

Requirements:

- IDL (project developed and tested under IDL version 8.2 (Windows) and version 8.5.1 (Linux))
- Set of SolarSoft packages (SSW IDL), including *SSW/gen*, *SSW/sdo*, *SSW/vobs* (*SSW/vobs/gen*, *SSW/vobs/ontology*)
- AS-IDL-Library package
- JeAn – Jet Analyzer package

Be sure that all packages are available in IDL Paths.

IMPORTANT! IDL must be started by the command *sswidl.bat* from the root of the SSW package.

Call and Parameters

Main entry point is *pipeline_aia.pro* function. There are 2 required parameters:

a. *config_file*: path to the configuration file in json format (see description below). Example *config_sample.json* can be found in the root package directory.
b. *work_dir*: directory where the results of the work will be saved. Result subdirectory structure explained below.

Optional calling parameters are listed in comment to this function.

Configuration File

Configuration file fields (see also *config_sample.json* example):

- "TIME_START": beginning time of the analyzed sequence,
- "TIME_STOP": ": end time,
- "TIME_REF": reference time (not very important, it may be beginning or end time),
- "X_CENTER": the position of the center of the cutout area in longitude (arcsec),
- "Y_CENTER": the position of the center of the cutout area in latitude (arcsec),
- "WIDTH_ARC": the size of the cutout area in longitude (arcsec),
- "HEIGHT_ARC": the size of the cut-out area in latitude (arcsec),
- "WAVES": An array of SDO/AIA wavelengths (Å).

Algorithm and Presets File

Algorithm for search jet-like events is based on the analysis of the difference between neighbor images (running difference).

The algorithm selects pixels on a running difference that significantly differ from the average level, and forms clusters of such pixels (according to the principle of spatio-temporal proximity). Further, clusters are considered as jets (more precisely, jet-like), if they are
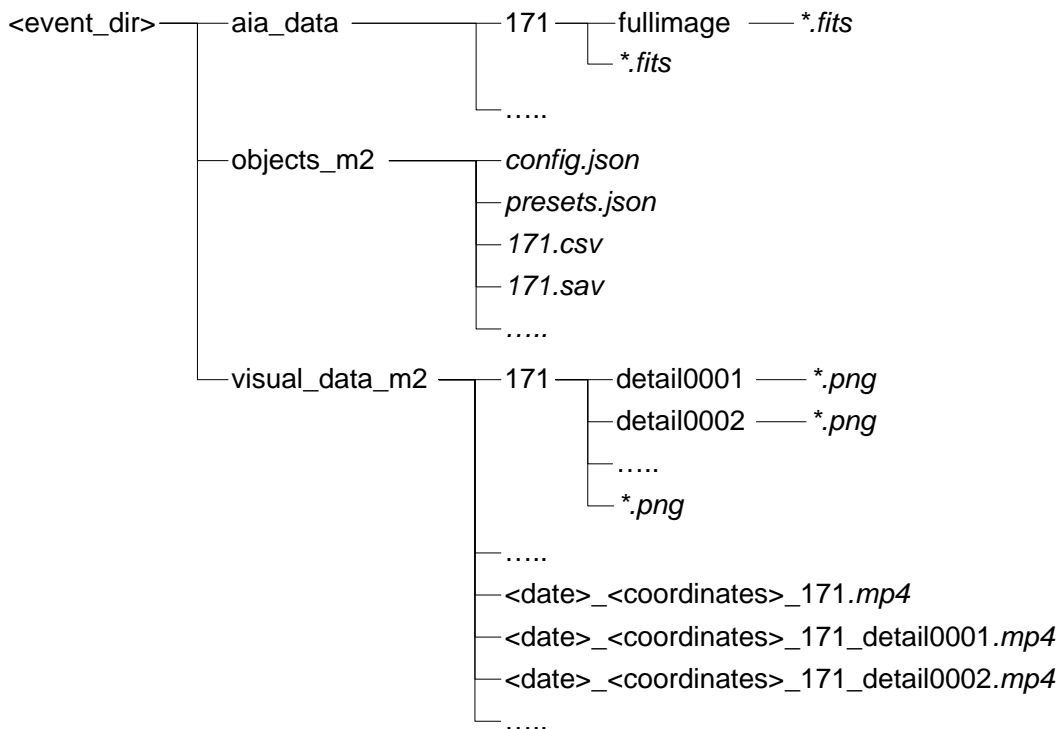
- sufficiently large (according to various estimates of the number of pixels);
- sufficiently long in time;
- sufficiently elongated (according to different estimates of length and width);
- sufficiently fast (according to various estimates of the speed of movement);
- sufficiently intense (according to additional estimates of the intensity distribution).

Parameters for such filtering listed in the presets file, sample (*presets_default.json*) can be found in the root of package (a detailed description will be provided if necessary). Default parameters is set to work with SDO/AIA 171 Å channel.

More detailed explanations[1] can be found in [1].

Directories and Files Created

After execution, the event directory (named by the beginning and end times and the position and size of the area) created in *work_dir*, and the structure of subdirectories:

```
<event_dir>──────aia_data────────────171──────fullimage ──── *.fits
                                       │        └── *.fits
                                       └── …..
              ──objects_m2 ──────────config.json
                                     ──presets.json
                                     ──171.csv
                                     ──171.sav
                                     └── …..
              ──visual_data_m2 ───────171──────detail0001 ──── *.png
                                       │       ──detail0002 ──── *.png
                                       │       ── …..
                                       │       └── *.png
                                       ── …..
                                       ──<date>_<coordinates>_171.mp4
                                       ──<date>_<coordinates>_171_detail0001.mp4
                                       ──<date>_<coordinates>_171_detail0002.mp4
                                       └── …..
```

- *aia_data*: for each channel, a subdirectory with a sequence of cutout fits (in addition *fullimage* subdirectory can be created, see key */ref_images* description in comments to the function)
- *objects_m2*:
  - copies of configuration and presets files
  - for each channel, a csv-file *<channel>.csv* with digital information about each event detail found (the list of parameters will be as needed)
  - for each channel, a sav-file *<channel>.sav* with detailed information about each found event detail (for further display and detailed analysis)
- *visual_data_m2*:
  - for each channel, a subdirectory with a sequence of generated images for each fits (the pictures are arranged as follows: the dynamics are in intensity on the left pane, in a running difference on the right pane, the found details are outlined in red). In addition, if there are found event details, in the same subdirectory subdirectories are created, named by the number of the found detail.
  - The most interesting thing: for each channel, a video is created with the name *<date>_<coordinates>_<channel>.mp4*. In addition, for each detail, a video file is created with the name *<date>_<coordinates>_<channel>_<part_number>.mp4*.

---

[1] But note that in this paper previous version of algorithm implementation was described. Actual implementation keeps the main idea, but default values of some parameters was tuned for better detail selection, and clusters are defined not frame-by-frame, but in 3d (coordinates-time) domain.

References

[1] Stupishin, A., Anfinogentov, S., Kaltman, T. Diagnostics of Parameters of Hot Jets in the Solar Corona in Time Series of Images. 2021, Geomagnetism and Aeronomy, **61**, Issue 8, p.1108. DOI, ADS