

Анализатор корональных струй (бета-версия)

Т. И. Кальтман, А. Г. Ступишин, С. А. Анфиногентов

13 октября 2020 г.

1 Шаги конвейера

Текущая версия написана на языке IDL и реализует 5 шагов:

- Скачивание данных AIA
- Вырезка анализируемого фрагмента диска
- Поиск кандидатов в струи
- Визуализация (статические картинки)
- Визуализация (видео)

Каждый этап может быть выполнен отдельно, **интерфейсы будут описаны позже**.
Модуль *pipeline_aia_all.pro* реализует все этапы одновременно.

2 IDL-пакет

Должна быть установлена версия IDL не ниже 8.2, и основной пакет SSW (см. [здесь](#) и [здесь](#)), с установленной миссией SDO и, **возможно, еще какими-то дополнениями, если появятся проблемы, надо будет что-то доставить**. Из известного авторам на данный момент: надо убедиться, что присутствует каталог *SSW/gen/idl/sunglobe* (и при отсутствии его загрузить из SSW, например, путем **обновления пакета SSW**). Для установки пакета конвейера поиска струй необходимо создать подкаталог (с любым именем, например, *jets*), в доступном для IDL месте (например, в *SSW/packages*). В этот подкаталог следует положить:

- подкаталог, содержащий собственно модули анализатора (*jet_analyzer*), может быть скачан [здесь](#)
- подкаталог, содержащий вспомогательные файлы (*common*), может быть скачан [здесь](#)

Файлы, представляющие интерес для конечного пользователя:

<i>jet_analyzer/config20160428.json</i>	пример конфигурационного файла
<i>jet_analyzer/pipeline_aia_all.pro</i>	все 5 шагов конвейера
<i>jet_analyzer/jettests.pro</i>	пример вызова <i>pipeline_aia_all.pro</i> в некоей конкретной среде
<i>jet_analyzer/jet_analyzer.pdf</i>	настоящая документация

Файлы, представляющие интерес, будут описаны позже:

<i>pipeline_aia_download_aia_full.pro</i>	загрузка данных AIA
<i>pipeline_aia_cutout.pro</i>	вырезание данных AIA
<i>pipeline_aia_find_candidates.pro</i>	поиск кандидатов в струи
<i>pipeline_aia_movie_prep_pict.pro</i>	подготовка статических картинок
<i>pipeline_aia_make_movie.pro</i>	изготовление видео из статических картинок

3 Конфигурационный файл

Конфигурационный файл должен быть представлен в формате *json* и содержать те же поля, что и приложенном примере:

```
{
  "TIME_START": "2017-08-24 09:00:00",    время начала поиска
  "TIME_STOP": "2017-08-24 09:10:00",     время конца поиска
  "TIME_REF": "2017-08-24 09:06:00",      время для преобразования координат
  "X_CENTER": -413.8,                     x-координата центра изучаемой области, arcsec
  "Y_CENTER": 3.5,                         у-координата центра изучаемой области, arcsec
  "WIDTH_PIX": 300,                       ширина (по x) центра изучаемой области, arcsec
  "HEIGHT_PIX": 300,                      ширина (по y) центра изучаемой области, arcsec
  "WAVES": [171, 304]                     список анализируемых длин волн
}
```

За время для преобразования координат можно взять предположительное время наблюдения струи или середину заданного временного интервала. Пример конфигурационного файла есть в корне каталога `jet_analyzer`. Элементы массива "WAVES" должны быть из набора длин волн, наблюдаемых SDO/AIA (КУФ: 94, 131, 171, 193, 211, 304, 335; УФ: 1600, 1700) (в Å). Для нас, по-видимому, в первую очередь представляют интерес волны 171 и 193 Å.

4 Структура каталогов

Функции, реализующие каждый этап, не зависят от расположения файлов на диске (рабочий каталог (`<work_dir>`) задается параметром вызова функции *pipeline_aia_all.pro*). Функция *pipeline_aia_all.pro* строит следующее дерево каталогов:

<code><work_dir></code>	параметр
<code>global_cache</code>	общий кэш
<code><date_time></code>	
<code><wavelength1></code>	скачанные aia fits
<code><wavelength2></code>	скачанные aia fits
<code>.....</code>	<code>.....</code>
<code><event_ID></code>	каталог для события
<code>aia_data</code>	
<code><wavelength1></code>	вырезанные aia fits
<code><wavelength2></code>	вырезанные aia fits
<code>.....</code>	<code>.....</code>
<code>objects</code>	кандидаты в струи
<code>visual_data</code>	визуализация
<code><wavelength1></code>	png
<code><wavelength2></code>	png
<code>.....</code>	<code>.....</code>

`<event_ID>` — каталог для каждого вырезанного и обработанного периода (соответствует конфигурационному файлу), и название строится из параметров конфигурационного файла:

`<дата_время начала>_<дата_время конца>_<координаты>_<размеры>`

Важное замечание: каталог *global_cache* будет занимать много дискового пространства (типичное событие — 2 – 5 GB на каждую длину волны), об этом следует помнить при выборе места. Для разделения исходных данных и результатов обработки можно при вызове указать локацию каталога общего кэша (см. ниже, описание параметров вызова, *cache_dir*). В случае, если кэш-каталог указан явно, *work_dir/global_cache* не создается.

Результаты обработки уже не так страшны, обычно не более 250 MB на каждую длину волны в событии, их можно уже класть в общий каталог на сервере.

Не менее важное замечание: первый этап конвейера — скачивание данных AIA (полнодисковых) — процесс не быстрый, зависит, разумеется, от коннекта, но для грубой оценки можно считать, что данные по каждой волне скачиваются «в масштабе реального времени» (т.е. наблюдения длительностью 3 часа будут скачиваться по 3 часа на каждую длину волны). Скачивание сразу вырезанного фрагмента анонсировано командой SDO, но работает ненадежно, и пока в конвейере не используется.

5 Вызов *pipeline_aia_all.pro*

Функция имеет следующие опциональные параметры:

- *config_file* — путь к конфигурационному файлу (по умолчанию файл *config.json* в текущем каталоге IDL)
- *work_dir* — каталог, в котором будет создаваться структура каталогов (по умолчанию текущий каталог IDL)
- *cache_dir* — каталог, в котором будет храниться данные, скачанные с сервера (по умолчанию подкаталог *work_dir/global_cache*)
- *fps* — частота кадров видео (по умолчанию 5)
- */no_load* — можно указать, например, если уже скачаны все нужные файлы, и хочется изменить, скажем, размер и/или положение вырезаемой области
- */no_cut* — можно указать, например, если уже скачаны все нужные файлы и вырезана нужная область, но не найдены кандидаты или не построены картинки/видео (например, из-за глюков или перезагрузки компьютера) **или хочется поменять настройки эвристик поиска струй, это будет описано позднее.**

Замечание: параметры */no_load*, */no_cut* следует использовать с осторожностью, только если есть уверенность, что **все** данные на **всех** волнах скачаны и вырезаны. Если в процессе выполнения произошел сбой и все слетело, правильнее всего сделать следующее:

- Проверить, какие длины волн уже обработаны полностью и, если таковые есть, удалить их из *config_file*.
- Повторить вызов (без параметров */no_load*, */no_cut*!). Уже скачанные файлы скачиваться не будут, но на проверку наличия файлов в *global_cache* все-таки небольшое время уйдет. Уже вырезанные будут тоже проверяться на наличие, это совсем быстрая операция.

- **другие параметры, будут описаны позднее, список дополняется**

Пример вызова *pipeline_aia_all.pro* в некоей конкретной среде: *jetttests.pro*.

Все этапы обработки осуществляются для каждой заданной длины волны последовательно.

6 Результат работы

Основные файлы:

- Файл *<work_dir>/<event_ID>/objects/<wavelength>.csv* — текстовый файл со списком кандидатов в струи. Создается только в том случае, если найден хоть один кандидат. Описание столбцов:

T start	время начала струи
T max	время максимума струи
T end	время конца струи
Duration	продолжительность
Max cardinality	макс. количество пикселей
Max aspect ratio	максимальная вытянутость
Min aspect ratio	минимальная вытянутость
X from	Координаты ограничивающего прямоугольника
X to	
Y from	
Y to	

Вытянутость оценивается как отношение характерной длины к характерной ширине. Координаты ограничивающего прямоугольника даны за всю историю струи. Если вы пользуетесь Excel'ем под Windows, открытие *.csv* Excel'ем может не разбить на столбцы, простейший способ исправить это — выделить первый столбец, перейти на ленту «данные» (Data), выбрать

«Текст по столбцам» (Text to Columns), и пройти по шагам Мастера, выбрав «с разделителем» (Delimited) и «запятая» (Comma).

- каталог `<work_dir>/<event_ID>/visual_data/<wavelength>` — картинки (*.png) для каждого момента времени. На левой панели показана интенсивность в УФ/КУФ (заголовок — момент времени), в цветовой гамме, принятой в миссии SDO/AIA, на правой — бегущая разность (в серой гамме), найденные кандидаты подсвечены справа красным цветом (заголовок — количество найденных пикселей в случае наличия кандидата).

Для каждого найденного кандидата строится дополнительный подкаталог (с именем *detailNN*, *NN* — номер кандидата) с картинками, вырезанными по времени существования кандидата и по полю зрения (соответственно, в более крупном масштабе).

- Файл `<work_dir>/<event_ID>/visual_data/<wavelength>.mp4` — видео, полученные из картинок из предыдущего пункта. Для каждого кандидата там же появляются дополнительные видео (с именами вида `<wavelength>_detailNN.mp4`).

7 Текущие цели

Конвейер работает как таковой, и это уже хороший результат, которым можно отчитываться. Однако эвристики поиска струй пока предположительные, худо-бедно работают в нескольких случаях, их надо настраивать/изменять/доделывать...

Мы полагаем, что на ближайшее время цели будут таковы:

- Каждый Участник проекта берет событие из нашего каталога **алгоритм распределения событий по участникам выработаем позднее**.
- Событие пропускается Участником через конвейер.
- Участник просматривает сначала получившиеся видео (полные и для кандидатов) для общего понимания, загружает в общий каталог на сервере **куда именно — скоро определим**, затем формирует свое мнение, список вопросов (типа «почему не найдена струя в такое-то время, хотя она там наверняка есть» или «почему найдена струя в такое-то время, хотя она и не струя вовсе») и замечаний (типа «хотелось бы видеть на картинках и в файлах такие-то и такие-то параметры»), и посылает все это Авторам данного документа.
- Авторы (и, скорее всего, АС) предпринимают меры по усовершенствованию кода, настройке параметров эвристик, и вообще делают все возможное и, при возможности, невозможное для того, чтобы получить приемлемый результат.
- И все это итерируется для получения приемлемого механизма поиска струй.

Проблема в том, что не все участники проекта представляют себе, что нужно искать (наверное, только ВН хорошо представляет себе, что именно). Возникает проблема обучения Участника. Как ее решать? Авторы пока не имеют четкого понимания (любые идеи будут приветствоваться) и в ближайшее время надеются выработать стратегию дальнейших шагов по достижению приемлемых результатов.