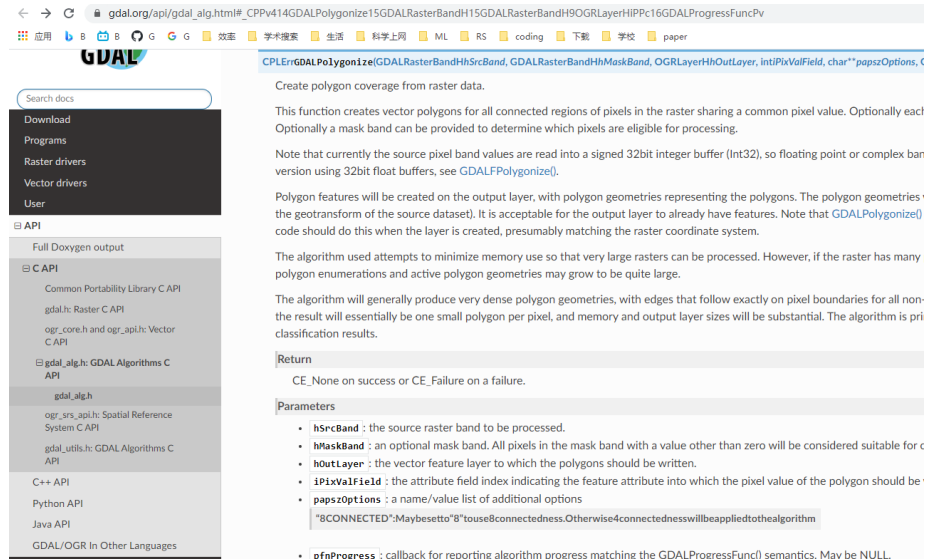


Gdal 文档使用小结

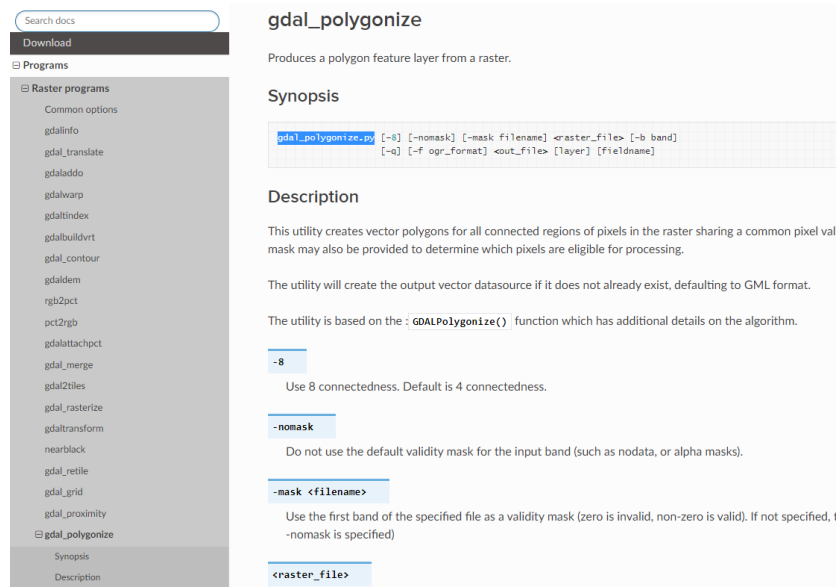
王爵 2021 年 3 月 25 日

- i. 任务：需要在 python 环境下，借助 gdal 完成遥感应用的基础性开发，如栅格转矢量。
- ii. 困难：以栅格转矢量为例，我们需要采用 python 接口 `gdal.polygonize`。然而，Gdal 的官方文档为 C++接口 `CPLerr GDALPolygonize`，与 Python 接口不完全一致，且参数解释语焉不详。



Gdal C++接口文档

- iii. 解决方案：
Step1、在文档的 programs 中，找到调用该 C++函数 `CPLerr GDALPolygonize` 的 python 文件 `gdal_polygonize.py`。该 python 文件对 C++函数从顶层进行封装，并介绍如何顶层调用。



Gdal 提供的顶层功能文件 gdal_polygonize.py

Step2、在 github 中，找到 gdal 项目中该 gdal 功能的顶层 python 文件。

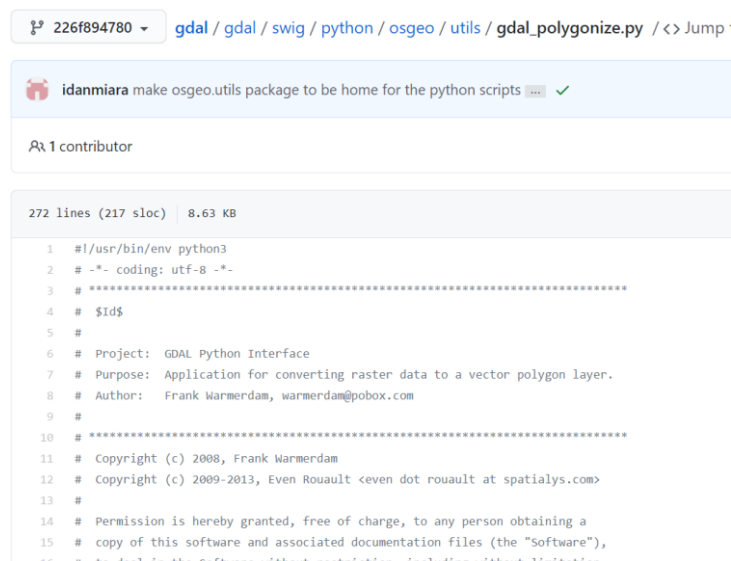


The screenshot shows the GitHub interface for the file `gdal / gdal / swig / python / scripts / gdal_polygonize.py`. The repository is owned by `rouault` and has 5 contributors. The file is an Executable File, 11 lines (8 sloc), and 315 Bytes. The code content is as follows:

```
1  #!/usr/bin/env python3
2
3  import sys
4  # import osgeo.utils.gdal_polygonize as a convenience to use as a script
5  from osgeo.utils.gdal_polygonize import * # noqa
6  from osgeo.utils.gdal_polygonize import main
7  from osgeo.gdal import deprecation_warn
8
9
10 deprecation_warn('gdal_polygonize', 'utils')
11 sys.exit(main(sys.argv))
```

Github gdal 项目中 `gdal_polygonize.py` 文件

Step3、根据上图红框中的位置，就可以找到 `gdal_polygonize.py` 的底层实现。



The screenshot shows the GitHub interface for the file `gdal / gdal / swig / python / osgeo / utils / gdal_polygonize.py`. The repository is owned by `idanmiara` and has 1 contributor. The file is 272 lines (217 sloc) and 8.63 KB. The code content is as follows:

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  # *****
4  # $Id$
5  #
6  # Project:  GDAL Python Interface
7  # Purpose:  Application for converting raster data to a vector polygon layer.
8  # Author:   Frank Warmerdam, warmerdam@pobox.com
9  #
10 # *****
11 # Copyright (c) 2008, Frank Warmerdam
12 # Copyright (c) 2009-2013, Even Rouault <even dot rouault at spatialys.com>
13 #
14 # Permission is hereby granted, free of charge, to any person obtaining a
15 # copy of this software and associated documentation files (the "Software"),
16 # to deal in the Software without restriction, including without limitation
```

```
mask = None\n    arg = argv[i]\n\n    if arg == '-f' or arg == '-of':\n        i = i + 1\n        frmt = argv[i]\n\n    elif arg == '-q' or arg == '-quiet':\n        quiet_flag = 1\n\n    elif arg == '-8':\n        options.append('8CONNECTED=8')\n\n    elif arg == '-nomask':\n        mask = 'none'
```

Step4、研读该文件，并与 `gdal_polygonize.py` 文件的网页文档比对，即可得知 `gdal` 文档没说清楚的功能，究竟是如何使用的。