**Extracting geographical data from images**

**Author: Dave K Brown,** *david.brown4@communities.gov.uk*

**Date: 30th July 2024**

This document explains how to extract a bounded area in a graphic image (e.g. PNG, JPG) so that geographical data can be created.

**Tools used:**

Python3 with libraries of: OpenCV, matplotlib, numby

JuypterLab

Gimp

QGis and the QGis python console and QuickMapServices

**Setup Steps:**

1. Download and install python3

C:\> python3 --version

Python 3.10.11

1. Download and install JuypterLab. Version 3.6.3
2. Download QGIS, Version 3.34.8-Prizren
3. Create a new QGIS project;

Add \\Web\QuickMapServices\OSM OSM Standard

Set EPSG:3857 – WGS 84 / Pseudo-Mercator (???)

1. If needed, install additional python libraries into QGIS

To check for which libraries are already installed use:

import pkg\_resources

installed\_packages = pkg\_resources.working\_set

installed\_packages\_list = sorted(["%s==%s" % (i.key, i.version) for i in installed\_packages])

for p in installed\_packages\_list:

print(p)

To install libraires into QGIS use:

pip.main(['install', 'shapely'])

1. Create a file structure as shown below:

Phase1\BaseImages

\FinalPolygon

\bin

1. Add the following files to the bin directory;

Phase1\bin\makeWhiteTransparent.bat

Phase1\bin\pullLineFromPDF\_2.ipynb

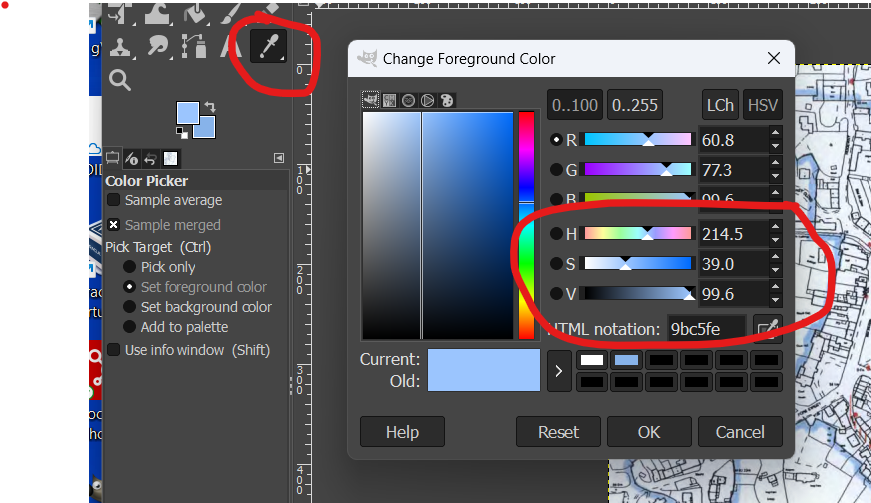
Phase1\bin\ReadCSVtoPolygon\_1.py

**Production Steps:**

* 1. Create a JPG or PNG of the best map of the conservation area, using a screen shot,
  2. Using Gimp find colour of highlighted area,
  3. Alter python3 script to look for the file and the colour found above,
  4. Using JupyterLab, run the image through OpenCV to extract a trace of the area,
  5. Tidy image in Gimp if needed, remove spurs and spurious features,
  6. Run trace image through gimp to make background transparent,
  7. Georeference in Qgis, at least four points,
  8. In QGis convert line to vectors,
  9. Run QGis python on vectors to thicken and close gaps,
  10. Import Polygon into QGIS

These production steps are detailed below:

1. **Create a JPG or PNG of the best map of the conservation area, using a screen shot,**
   * 1. Source the best online image we can find, Column D of the “**Conservation Area List**” in the Areas GoogleSheet, <https://docs.google.com/spreadsheets/d/1r96ruDklYhgm0m0mASR2Bsxc7KE59FQ1>
     2. Take a screenshot and save the base Image as “Phase1\BaseImages\ConservationArea\_NNNN.png”
2. **Using Gimp find colour of highlighted area**

****

Record the figures, Hue (H), Saturation (S) and Value (V).

1. **Alter python3 script to look for file and the colour found above**

Within the JuypterLab script look for the two sections below:

14 DIR = "C:\\Users\\DavidBrown\\Documents\\GIS\\ Phase1\\BaseImages"

15 fileName = "\\ConservationArea\_NNNN.png"

31 gimpH = 217

32 gimpS = 29

33 gimpV = 95

The basic tolerance is preset but can easily be altered:

30 tolerance\_percent = 2

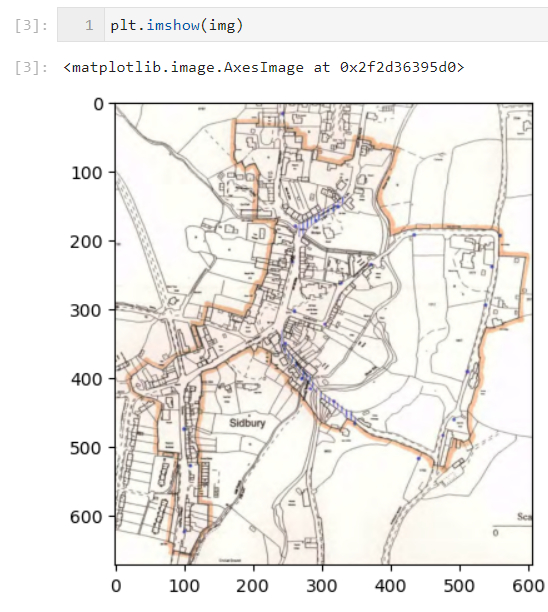
1. **Using JupyterLab, run the image through OpenCV to extract a trace of the area**

The script can be amended in many ways, certainly the tolerance can be altered.

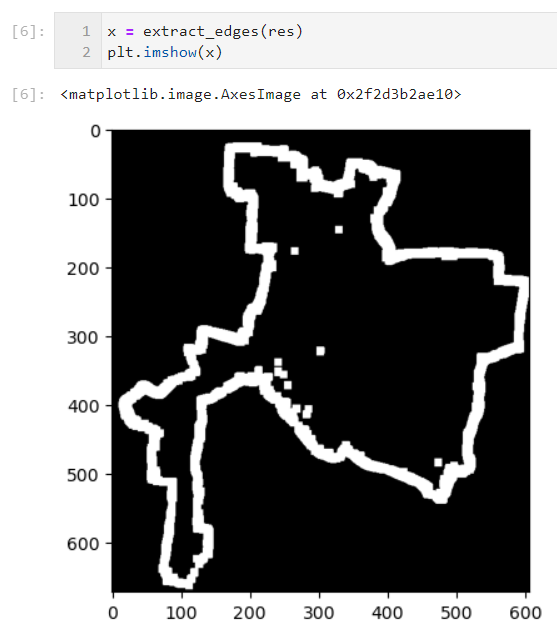
But also many parameters within the extract\_lines function.

Stop when you have an acceptable line trace.

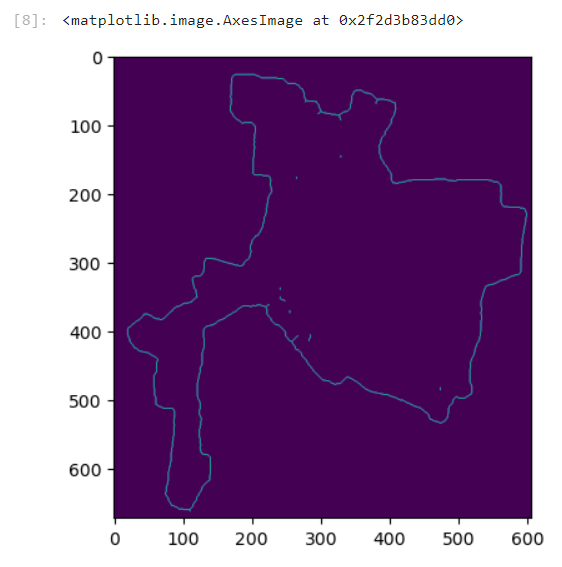
Initial Image



After “extract\_edges function” Image



Final Image

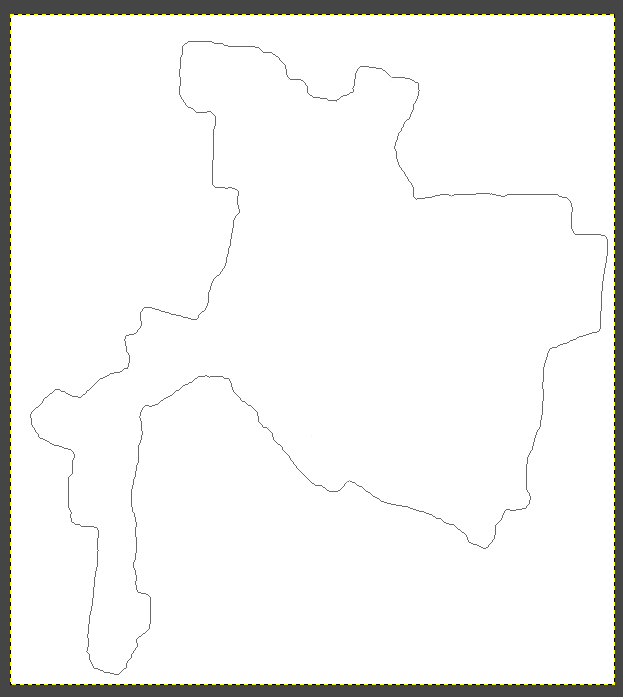


The output file will be named:

. . . \Phase1\BaseImages\img\_ConservationArea\_NNNN.png

1. **Tidy image in Gimp if needed, remove spurs and spurious features.**

Open image in Gimp and tidy it.



Save it as “Phase1\BaseImages\cleaned\_img\_ConservationArea\_NNNN.png”

1. **Run trace image through gimp to make background transparent**

PS C:\Users\DavidBrown\Documents\GIS\beer> .\makeWhiteTransparent.bat > mWT.log 2>&1

"C:\Users\DavidBrown\AppData\Local\Programs\GIMP 2\bin\gimp-2.10.exe" -i -b "( let\* ( ( image ( car (file-png-load 1 \"C:/Users/DavidBrown/Documents/GIS/beer/beer\_line\_only.png\" \"C:/Users/DavidBrown/Documents/GIS/beer/beer\_line\_only.png\" ) ) ) ( drawable ( car (gimp-image-active-drawable image ) ) ) ) ( plug-in-colortoalpha 1 image drawable '(255 255 255) ) ( gimp-file-save RUN-NONINTERACTIVE image drawable \"C:/Users/DavidBrown/Documents/GIS/beer/\_beer\_line\_only.png\" \"C:/Users/DavidBrown/Documents/GIS/beer/\_beer\_line\_only.png\") )" -b "(gimp-quit 0)"

The following is **suppressed** by using the 2>&1 at the end of the command.

using gegl copy

batch command executed successfully

C:\Users\DavidBrown\AppData\Local\Programs\GIMP 2\bin\gimp-2.10.exe: GEGL-WARNING: (../gegl-0.4.48/gegl/buffer/gegl-tile-handler-cache.c:1076):gegl\_tile\_cache\_destroy: runtime check failed: (g\_queue\_is\_empty (&cache\_queue))

1. EEEEeEeek! 3 GeglBuffers leaked

(Type any character to close this window)

1. Georeference in Qgis, at least four points,
2. Open QGIS project
3. Navigate to [\\Layer\georeferencer](file:///\\Layer\georeferencer)
4. Click on the Open Raster top menu button and navigate to the file created above: Phase1\BaseImages\\_cleaned\_img\_ConservationArea\_NNNN.png
5. Using the original image georeference at least four points
   1. Add Point,
   2. Click on image, then click on same place on base map in QGIS,
   3. Repeat,
   4. Click the green play button,
   5. Accept the default output file name,
   6. Press green button again
   7. Save the points file to . . .
6. **In QGis convert line to vectors,**
   1. \\Raster\Conversion\Polygonize (Raster to Vector),

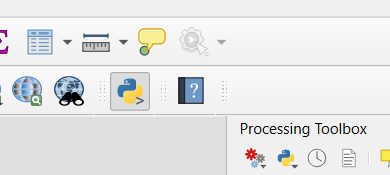
A screenshot of a computer

Description automatically generated

* 1. Right-hand click make permanent

1. **Run QGis python on vectors to thicken and close gaps**

Open the python console:

****

Open Phase1\bin\ReadCSVtoPolygon\_1.py

1. **Import Polygon into QGIS**

\\Layer, Data Source Manager => As below

A screenshot of a computer

Description automatically generated

Accept defaults!