

QGIS Application - Feature request #8602

Clipper for multiple rasters

2013-09-12 02:45 AM - Luca Congedo

Status:	Closed	
Priority:	Normal	
Assignee:		
Category:		
Pull Request or Patch supplied:	No	Resolution:
Easy fix?:	No	Copied to github as #: 17344
Description		
Hello everybody, I have written a Python script for clipping multiple rasters at once using gdal_translate (see http://fromgistors.blogspot.com/2013/09/how-to-clip-multiple-rasters-at-once.html). <pre># input variables # where to save clipped rasters outputDir = "/home/user/Desktop/clip" # upper left point UX, UY = 298545 , 4628145 # lower right point LX, LY = 308715 , 4622745 # actual script import os lddLrs = qgis.utils.iface.legendInterface().layers() for lyr in lddLrs: if (lyr.type()==QgsMapLayer.RasterLayer): os.system("gdal_translate -projwin " + str(UX) + " " + str(UY) + " " + str(LX) + " " + str(LY) + " -of GTiff " + str(lyr.source()) + " " + outputDir + "/" + str(lyr.name()) + ".tif") qgis.utils.iface.addRasterLayer(str(outputDir + "/" + lyr.name() + ".tif"), str(lyr.name() + "_clip.tif"))</pre> It would be nice if it can be implemented in the main Clipper tool in the Raster menu, or maybe ported to C++. Cheers!		

Associated revisions

Revision 9ce316db - 2013-09-13 11:09 AM - Alexander Bruy

[processing] add Clip raster by extent algorithm (fix #8602)

History

#1 - 2013-09-12 10:32 AM - Paolo Cavallini

I suggest to add it to Processing instead: it should be much easier and more flexible.

#2 - 2013-09-13 01:28 AM - Luca Congedo

- File *Clip_Multiple_Rasters.py.help* added
- File *Clip_Multiple_Rasters.py* added

It sounds good to me.

I have written the script for the Processing framework in attachment. I have also added the option for No data values.

```
##Raster=group
##Input=multiple raster
# upper left point
##UX=number
##UY=number
# lower right point
##LX=number
##LY=number
##No_data=boolean
##No_data_value=number 0
##Output_directory=folder

import os
import processing

IddLrs = Input.split(';')

if No_data == 0:
    for lyr in IddLrs:
        os.system("gdal_translate -projwin " + str(UX) + " " + str(UY) + " " + str(LX) + " " + str(LY) + " -of GTiff " + str(lyr) + " " + Output_directory +
"/clip_" + str(os.path.basename(str(lyr))))
        #loading fails
        #processing.load(str(Output_directory + "/clip_" + str(os.path.basename(str(lyr)))))
else:
    for lyr in IddLrs:
        os.system("gdal_translate -a_nodata " + str(No_data_value) + " -projwin " + str(UX) + " " + str(UY) + " " + str(LX) + " " + str(LY) + " -of GTiff " +
str(lyr) + " " + Output_directory + "/clip_" + str(os.path.basename(str(lyr))))
        #loading fails
        #processing.load(str(Output_directory + "/clip_" + str(os.path.basename(str(lyr)))))
```

I have tested it in QGIS 1.8 (renaming to SEXTANTE) and QGIS 2.

Clipped rasters are loaded in QGIS 1.8, while it fails to load the clipped rasters in QGIS 2. Maybe I am missing something. However the clipping works.

Cheers!

#3 - 2013-09-13 02:09 AM - Alexander Bruy

- Status changed from Open to Closed

Fixed in changeset commit:"9ce316dbd3359e5ba0ffd468b666be35b7360318".

#4 - 2013-09-13 05:24 AM - Luca Congedo

Perfect.
Thank you!

Files

Clip_Multiple_Rasters.py	978 Bytes	2013-09-12	Luca Congedo
Clip_Multiple_Rasters.py.help	1.34 KB	2013-09-12	Luca Congedo