depal (version 0.0.1)

index
/home/sachin/Projects/depal/depal.py

depal.py: Digital Earth Pacific (Abstration Library)

Modules

geopandas	numpy	matplotlib.pyplot	
matplotlib	planetary_computer	pystac_client	stackstac
xrspatial.multispectral	pandas	rasterio	xarray

Functions

coastal_clip(aoi, data)

colour_maps()

get_area_from_geojson(geojson_file)
```
    # AOI from GeoJson File (use geojson.io)
```

get_cloudless_mosaic(aoi, collection_name='sentinel-2-l2a', timeframe='2019-11-01/2022-11-31', cloudcover=10, resolution=100, max=100, period='yearly')
```
    # median composite
```

get_country_boundary(country, admin_type, admin)

get_data(aoi, bands=[], collection_name='sentinel-2-l2a', timeframe='2023-01-01/2023-12-31', cloudcover=10, resolution=100, max=30, period='monthly')
```
    # xarray dataset from stac
```

get_evi(aoi, collection_name='sentinel-2-l2a', timeframe='2019-11-01/2022-11-31', cloudcover=10, resolution=100, max=100, period='monthly')
```
    # evi
```

get_gci(aoi, collection_name='sentinel-2-l2a', timeframe='2019-11-01/2022-11-31', cloudcover=10, resolution=100, max=100, period='monthly')
```
    # gci
```

get_global_land_cover(name='io-lulc-9-class')

get_latest_images(aoi, collection_name='sentinel-2-l2a', timeframe='2023-01-01/2023-12-31', cloudcover=10, resolution=100, max=30, period='daily')
```
    # latest RGB
```

get_ndmi(aoi, collection_name='sentinel-2-l2a', timeframe='2019-11-01/2022-11-31', cloudcover=10, resolution=100, max=100, period='monthly')
```
    # ndmi
```

get_ndvi(aoi, collection_name='sentinel-2-l2a', timeframe='2019-11-01/2022-11-31', cloudcover=10, resolution=100, max=100, period='monthly')
```
    # ndvi
```

get_sipi(aoi, collection_name='sentinel-2-l2a', timeframe='2019-11-01/2022-11-31', cloudcover=10, resolution=100, max=100, period='monthly')
```
    # sipi
```

list_boundary_types(country)

list_countries()

list_country_boundary(country, admin_type)

list_data_assets(collection_name)

list_data_bands(collection_name='sentinel-2-l2a')

list_data_sources()

list_global_land_cover()

save(data, file_name)
```
    # save data as COG series
```

setup_dask(maxWorkers=2)
```
    # Remote Dask
```

smooth(data)
```
    # focal mean smooting
```

visualise(data, cmap=None)
```
    # needs improvement, flexibility
```

Data

__copyright__ = 'Pacific Community (SPC)'
__email__ = 'sachindras@spc.int'
__license__ = 'GPL'
__status__ = 'Development'
catalog = <Client id=microsoft-pc>
chunk_size = 4096
client = <Client: 'tcp://127.0.0.1:46577' processes=4 threads=12, memory=30.97 GiB>
cluster = LocalCluster(ea7ae1fc, 'tcp://127.0.0.1:46577', workers=4, threads=12, memory=30.97 GiB)
default_resolution = 100
padm = country ... ... -176.24805 -13.28860))) [698 rows x 12 columns]

Author

Sachindra Singh