

IDL Programming with ENVITASK

ENVI and IDL are trademarks of L3HARRIS, Inc.
http://www.harrisgeospatial.com



Intro

본 문서에서는 ENVITASK 함수를 활용한 IDL 프로그래밍 실습을 진행하고자 합니다.

ENVITASK

ENVI 내 대부분의 기능은 ENVITASK 함수를 통해 IDL에서 제공되며, 사용자는 ENVITASK 함수를 활용하여 일련의 자동화된 처리를 수행할 수 있습니다. ENVITASK 함수에서 제공되는 각 기능에 대한 설명은 설치경로 내 HTML 링크를 통해 확인할 수 있습니다.

⇒ C:\W\Program Files\Harris\ENVI55\IDL
87\help\online_help\Subsystems\envi\Content
WExtendCustomize\ENVITasks

ENVITASK는 ENVI와 동일한 형태의 Workflow 형식으로, [ENVI 기능 선택]-[입력 파일 설정]-[파라미터 설정]-[출력 경로 및 파일명 설정] 순으로 이루어집니다.

Tutorial

본 실습에서는 ENVITASK 함수를 활용하여 Landsat-8 Level-1 데이터의 전처리 (방사 및 대기보정)를 수행하고자 하며, 1) Landsat-8 데이터 입력 2) Radiometric Calibration 3) QUAC 순으로 이루어집니다. 구현 과정은 다음과 같습니다.

Step 1. Setting

첫번째 과정에서는 ENVI 실행 및 입출력 경로 지정을 수행합니다. 또한, 입출력 경로에서 Landat-8 데이터를 찾는 과정까지 이루어집니다.

```
COMPILE_OPT IDL2, HIDDEN
ON_ERROR, 2

; Execute ENVI
e = ENVI()

; Path
in_DIR = 'C:\Users\USER\Downloads'
files = FILE_SEARCH(in_DIR+'*LC08_*'+*WLC08*_*.MTL.txt', $
count = ct, /FOLD_CASE)
L8_list = STRARR(ct)
FOR j = 0, ct-1 DO L8_list[j] = files[j]

out_DIR = 'C:\Users\USER\Downloads\Result'
```

코드 내 각 과정에 대한 설명은 다음과 같습니다.

1) COMPILE_OPT IDL2, HIDDEN

COMPILE_OPT는 프로시저나 함수를 컴파일하는데 있어 가정하는 기본 설정을 인위적으로 변경하는 역할을 합니다. 이때, IDL2는 DEFINT32, STRICTARR를 나타내며, 각각 모든 정수는 4 byte의 Long Integer로 인식 그리고 배열의 인덱스를 명시할 때 []만 허용 (: () 사용 불가)을 의미합니다. 자세한 설명은 아래의 링크에서 확인할 수 있습니다.

⇒ <http://blog.daum.net/swrush/146>

2) ON_ERROR, 2

에러 발생시, 해당 줄(LINE)로 이동합니다.

2) e = ENVI(/headless)

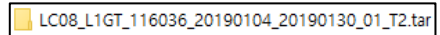
ENVI를 실행시키는 과정으로, e 변수를 선언하여 ENVI를 제어합니다. 이때, /headless 키워드를 사용할 경우, ENVI 인터페이스는 표출되지 않고, 백그라운드를 통해 실행됩니다. 만약 /headless 키워드를 사용하지 않으면 (e = ENVI()), ENVI 인터페이스가 표출됩니다.

3) 입출력 경로 지정

입출력 경로를 각각 in_DIR와 out_DIR로 설정하였습니다.

4) Landsat-8 데이터 검색

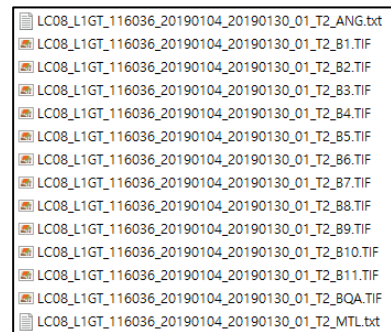
Landsat-8 데이터는 USGS 홈페이지에서 다운로드 가능하며, LC08_*.tar 로 제공됩니다.



폴더명은 LXSS_LLLL_PPPRRR_YYYYMMDD_yyyymmdd_CC_TX로 구성됩니다.

- ① L : Landsat / X : 센서 (C → OLI/TIRS) / SS : 위성 (08 → Landat-8)
- ② LLLL : 보정 단계 (GT : Corrected systematically for ground and terrain)
- ③ PPP : Path / RRR : Row
- ④ YYYYMMDD : 촬영 날짜 / yyyymmdd : 처리 날짜
- ⑤ CC : Collection number / TX : Collection category

또한, 폴더 하부에는 ① *_MTL.txt ② *.TIF ③ *_ANG.txt 의 파일로 이루어져 있습니다. 아래의 그림은 본 실습에서 사용한 Landsat-8 데이터입니다.



각각에 대한 설명은 다음과 같습니다.

- ① 메타데이터 파일로, 데이터 입력에 필요합니다.
- ② 각 분광밴드가 TIFF 형태로 제공됩니다.
- ③ Angle-coefficient 파일로, sensor-viewing angle에 대한 model coefficients가 제공됩니다.

Landsat-8 데이터 입력을 위해서는 입력 경로 내 *_MTL.txt 파일이 요구되며, 이는 FILE_SEARCH 함수를 통해 가능합니다. 이때, in_DIR+'*LC08_*'+*WLC08*_*.MTL.txt' 를 통해 입력 경로 내 모든 Landsat-8 폴더에서 메타데이터 파일을 검색할 수 있으며, 이를 L8_list에 저장하였습니다.

IDL Main Body

다음 과정에서는, 입력된 Landsat-8 데이터에 대하여 Radiometric Calibration 및 QUAC을 통해 전처리를 수행합니다.

```
FOR k = 0, ct-1 DO BEGIN
  L8 = L8_list[k]
  file = e.OpenRaster(L8)
  MS = file[0]
  date = STRMID(L8, 30, 8, /REVERSE_OFFSET)

  task_1 = ENVITask('RadiometricCalibration')
  task_1.input_raster = MS
  task_1.calibration_type = 'Top-of-Atmosphere Reflectance'
  task_1.Execute

  task_2 = ENVITask('QUAC')
  task_2.input_raster = task_1.output_raster
  task_2.output_raster_uri = out_DIR + PATH_SEP() + date + '.BOA'
  task_2.Execute
ENDFOR
```

각 과정에 대하여 살펴보면 다음과 같습니다.

1) Landsat-8 데이터 입력

FOR문을 통해, L8_list에 저장된 각각의 Landsat-8 메타데이터 파일을 ENVI에 입력합니다 (OpenRaster 함수 이용). 이때, 5개의 오브젝트를 가진 배열이 생성됩니다.

```
ENVI> print, L8
<ObjHeapVar565928(ENVIRASTER)>
<ObjHeapVar565929(ENVIRASTER)>
<ObjHeapVar565930(ENVIRASTER)>
<ObjHeapVar565931(ENVIRASTER)>
<ObjHeapVar565932(ENVIRASTER)>
```

위의 요소들은 순서대로 ① Multispectral ② Panchromatic ③ Cirrus ④ Thermal ⑤ Quality를 나타냅니다. 더 나아가, Multispectral의 경우, Coastal aerosol, Blue, Green, Red, NIR, SWIR1, SWIR2를, Thermal의 경우, TIR1, TIR2 밴드를 포함합니다.

2) 다중분광 밴드 추출

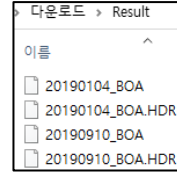
본 실습에서는 Landsat-8 데이터 내 Multispectral 영상만을 활용하여 전처리를 수행할 예정입니다. 따라서, L8[0]을 통해 L8로 정의된 배열의 첫번째 오브젝트인 Multispectral 밴드만을 추출하였습니다. 만약, Thermal 밴드만을 추출하고자 한다면, L8[3]을 통해 가능합니다.

3) 전처리

전처리의 각 과정은 ENVITASK 함수에서 필요 기능 선택 및 파라미터 설정을 통해 이루어지며, 제공되는 기능 및 파라미터는 앞서 언급한 HTML 링크에서 확인할 수 있습니다. 만약 각 task의 결과를 저장하고자 할 경우, output_raster_uri 파라미터에서 출력 경로 및 파일명 설정을 수행해야 합니다. 본 실습에서는 task_1 결과에 대한 저장 경로를 따로 설정하지 않았기 때문에, task_2의 입력자료를 task_1.output_raster로 설정하였습니다.

또한, 파일명 설정을 위해 STRMID 함수를 통하여 Landsat-8 데이터 폴더에서 촬영 날짜 (YYYYMMDD)의 문자열을 추출하였습니다. 이는 L8_list의 메타데이터 경로에서 수행하였으며, /REVERSE_OFFSET은 뒤에서부터의 순서를 세겠다는 의미입니다.

Task_2를 통해 획득한 최종 결과물은 출력 경로 내에 "촬영날짜_BOA"로 저장하였습니다.



만약, ENVI 실행시, /headless 대신 e = ENVI()를 통해 ENVI를 호출하였다면, 인터페이스가 실행됩니다. 이때, 최종적인 결과물을 인터페이스에 표출할 수 있습니다.

```
PRO Batch_TEST
  COMPILE_OPT IDL2, HIDDEN
  On_ERROR, 2

  ; Execute ENVI
  e = ENVI()

  ; Path
  in_DIR = 'C:\Users\USER\Downloads'
  files = FILE_SEARCH(in_DIR + '\LC08_*_*\*LC08_*_*.MTL.txt', $
    , count = ct, /FOLD_CASE)
  L8_list = STRARR(ct)
  FOR j = 0, ct-1 DO L8_list[j] = files[j]

  out_DIR = 'C:\Users\USER\Downloads\Result'

  ; Get Data in the Data Manager
  DataColl = e.Data

  ; Pre-processing
  FOR k = 0, ct-1 DO BEGIN
    L8 = L8_list[k]
    file = e.OpenRaster(L8)
    MS = file[0]
    date = STRMID(L8, 30, 8, /REVERSE_OFFSET)

    task_1 = ENVITask('RadiometricCalibration')
    task_1.input_raster = MS
    task_1.calibration_type = 'Top-of-Atmosphere Reflectance'
    task_1.Execute

    task_2 = ENVITask('QUAC')
    task_2.input_raster = task_1.output_raster
    task_2.output_raster_uri = out_DIR + PATH_SEP() + date $
      + '.BOA'
    task_2.Execute

    ; Add output to the Data Manager
    DataColl.Add, task_2.output_raster

    ; Display the result
    View = e.GetView()
    Layer = View.CreateLayer(task_2.output_raster)
  ENDFOR
END
```

인터페이스 표출을 위한 추가 명령은 다음과 같습니다.

1) DataColl = e.Data

e.Data를 통해 Data Manager를 불러온 후, Add 함수를 통해 task_2의 결과물을 Data Manager에 추가합니다.

2) View = e.GetView()

e.GetView()를 통해 View 생성 후, CreateLayer를 통해 task_2의 결과물을 표출합니다. 만약 View를 추가하고 싶다면, view2 = e.CreateView()를 통해 수행할 수 있습니다.

