

---

Subject: Re: Objects in ENVI ROI files

Posted by [Josh Sixsmith](#) on Fri, 24 May 2013 04:22:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Using a toy example, I have a single ROI file containing 3 groups; Group A has 5 polygons, Group B has 4 polygons and Group C has 3 polygons.

`roi_ids = envi_get_roi_ids(fid=fid)` ; should give me an array of length 3, indicating 3 distinct groups.

`roi_addr_A = envi_get_roi(roi_ids[0])` ; will give me the pixels contained within Group A

`roi_addr_B = envi_get_roi(roi_ids[1])` ; will give me the pixels contained within Group B

`roi_addr_C = envi_get_roi(roi_ids[2])` ; will give me the pixels contained within Group C

I make an assumption here that each polygon within a single group shares no common border with another polygon. If two polygons within a group share a border then the `label_region()` function will not distinguish between the two and lump them together. It makes no sense (to me) for polygons of the same group to share a border, so they should be merged. An alternative would be to put one of those polygons into a different group.

Anyway, we now have the addresses (indices) of all the pixels contained within Group A. If there are no borders shared between the polygons contained within Group A, then the `label_region()` function should label the image with unique ids for each polygon, eg 1,2,3. A value of 0 will be the background.

```
labels = label_arr[uniq(label_arr, sort(label_arr))]
```

In this example, Group A has 5 polygons, the length of labels should be 6, taking into account the background value of 0.

```
num_polygons = n_elements(labels) -1
```

```
print, num_polygons
```

```
5
```

The rest of the code I posted previously will then visit each polygon and retrieve the data contained within.

In your code example:

```
for i=0, n_elements(roi_ids)-1 do begin
```

```
    roi_addr = envi_get_roi(roi_ids[i], roi_name=name)
```

```
    ENVI_GET_ROI_INFORMATION,roi_ids, ns=ns, nl=nl, npts=npts
```

```
    dummy = bytarr(ns,nl)
```

```
    dummy[roi_addr] = 1
```

```
    label_arr = label_region(dummy)
```

```
    labels = label_arr[uniq(label_arr, sort(label_arr))]
```

```
    num_polygons = n_elements(labels) -1 ;ignoring background value of 0
```

```
    print, format=('%"Polygon %s contains %i polygons "'), name, num_polygons  
endfor
```

I hope that clarifies things. If it still doesn't work, then you might have polygons that share a common border. If so, then consider putting one of those polygons into a different group.

Cheers  
Josh

---