
Subject: Re: Point within country boundary
Posted by [Fabzi](#) on Sat, 25 Jan 2014 12:54:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

On 24.01.2014 22:36, David Fanning wrote:
> Well, believe it or not, this is nearly instantaneous!

Contain points is much slower than computeMask, see following test:

pro test_contain_points

```
poly_x = [10, 90, 90, 10, 10] + 0.1  
poly_y = [10, 10, 90, 90, 10] + 0.1
```

```
n = 1800
```

```
o = IDLanROI(poly_x, poly_y)
```

```
print, 'Compute mask'
```

```
tic
```

```
result = o->ComputeMask(DIMENSION=[n,n])
```

```
toc
```

```
i = Image(result)
```

```
xx = INDGEN(n) # (LONARR(n) + 1)
```

```
yy = INDGEN(n) ## (LONARR(n) + 1)
```

```
mask = BYTARR(n,n)
```

```
print, 'Contain point'
```

```
tic
```

```
result = o->ContainsPoints(xx,yy)
```

```
toc
```

```
mask[where(result)] = 255
```

```
i = Image(mask)
```

```
mask = BYTARR(n,n)
```

```
print, 'This does the trick'
```

```
tic
```

```
totest = where(o->ComputeMask(DIMENSION=[n,n]))
```

```
result = o->ContainsPoints(xx[totest],yy[totest])
```

```
toc
```

```
mask[totest[where(result)]] = 255
```

```
i = Image(mask)
```

```
end
```

on my machine:

```
IDL> test_contain_points
```

```
Compute mask
```

```
% Time elapsed: 0.00040602684 seconds.
```

```
Contain point
```

```
% Time elapsed: 0.53887701 seconds.
```

```
This does the trick
```

```
% Time elapsed: 0.0035710335 seconds.
```

Since contains points can do what compute mask can't, I use the trick above to spare computing time...

Cheers,

Fabien
