Subject: Point within country boundary
Posted by Matt[3] on Fri, 24 Jan 2014 18:48:10 GMT
View Forum Message <> Reply to Message

Hi All,

I'm trying to find out which countries a set of points fall within. I thought I could do this using the supplied map shapefiles and IDLanROI. However, when I try this:

```
country_path = "/usr/local/exelis/idl/resource/maps/shape/country.shp"

oSHP = OBJ_NEW('IDLffShape', country_path )
ent1 = oSHP -> GetEntity( 34, /ATTRIBUTES ) ;USA
OBJ_DESTROY, oSHP

oROI = OBJ_NEW('IDLanROI', (*ent1.vertices))
test = oROI -> ContainsPoints([-100.], [35.])
OBJ_DESTROY, oROI

print, test
```

... I get 0, whereas I'm pretty sure -100E, 35W should be within the USA.

Am I misunderstanding how these objects or shape files should be used?

Thanks in advance for any pointers,

Matt

Subject: Re: Point within country boundary
Posted by David Fanning on Fri, 24 Jan 2014 19:08:20 GMT
View Forum Message <> Reply to Message

Matt writes:

> I'm trying to find out which countries a set of points fall within. I thought I could do this using the supplied map shapefiles and IDLanROI. However, when I try this:
>
```
>    country_path = "/usr/local/exelis/idl/resource/maps/shape/country.shp"
>
>    oSHP = OBJ_NEW('IDLffShape', country_path )
>    ent1 = oSHP -> GetEntity( 34, /ATTRIBUTES ) ;USA
>    OBJ_DESTROY, oSHP
>
>    oROI = OBJ_NEW('IDLanROI', (*ent1.vertices))
>    test = oROI -> ContainsPoints([-100.], [35.])
>    OBJ_DESTROY, oROI
```

>
> print, test
>
> ... I get 0, whereas I'm pretty sure -100E, 35W should be within the USA.
>
> Am I misunderstanding how these objects or shape files should be used?
>
> Thanks in advance for any pointers,

It is a little more complicated than that. :-)

When I extract the USA shape from that file with cgExtractShape, I get
back an IDLanROIGroup object that contains 59 separate IDLanROI objects.
You are probably not looking in the right one. ;-)

  http://www.idlcoyote.com/code_tips/extractpoly.php

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Point within country boundary
Posted by Phillip Bitzer on Fri, 24 Jan 2014 19:43:31 GMT
View Forum Message <> Reply to Message

I am curious - is the IDLanROIGroup method ContainsPoints more efficient than using the routine discussed below?

http://www.idlcoyote.com/tips/point_in_polygon.html

Subject: Re: Point within country boundary
Posted by David Fanning on Fri, 24 Jan 2014 20:06:00 GMT
View Forum Message <> Reply to Message

Phillip Bitzer writes:

> I am curious - is the IDLanROIGroup method ContainsPoints more efficient than using the routine discussed below?
>
> http://www.idlcoyote.com/tips/point_in_polygon.html

Don't know the answer to that. I was, frankly, surprised to see the
method listed for IDLanROIGroup. If it is optimized, I would be even
more surprised. :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Point within country boundary
Posted by Phillip Bitzer on Fri, 24 Jan 2014 21:22:09 GMT
View Forum Message <> Reply to Message

On Friday, January 24, 2014 2:06:00 PM UTC-6, David Fanning wrote:
> Phillip Bitzer writes:
>
>
>
>> I am curious - is the IDLanROIGroup method ContainsPoints more efficient than using the routine discussed below?
>
>>
>
>> http://www.idlcoyote.com/tips/point_in_polygon.html
>
>
>
> Don't know the answer to that. I was, frankly, surprised to see the
>
> method listed for IDLanROIGroup. If it is optimized, I would be even
>
> more surprised. :-)
>
>
>
> Cheers,
>
>
>
> David
>
> --

>
> David Fanning, Ph.D.
>
> Fanning Software Consulting, Inc.
>
> Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
>
> Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Well, I guess the question was more or less directed to Matt - since he has the data readily available for testing. I've tried to use ContainsPoints and found it was, um, slow. Inside was much faster, but I wonder if that's me or the routine. /me shrugs

---

Subject: Re: Point within country boundary
Posted by David Fanning on Fri, 24 Jan 2014 21:36:12 GMT
View Forum Message <> Reply to Message

Phillip Bitzer writes:

> Well, I guess the question was more or less directed to Matt - since he has the data readily available for testing. I've tried to use ContainsPoints and found it was, um, slow. Inside was much faster, but I wonder if that's me or the routine. /me shrugs

Well, believe it or not, this is nearly instantaneous!

```
IDL> Print, file
       C:\Program Files\Exelis\IDL82\resource\maps\shape\country.shp
IDL> canada = cgExtractShape(file, 'CNTRY_NAME', 'CANADA')
IDL> tic & print, canada -> ContainsPoints(-100, 35) & toc
      0
 Elapsed Time:  0.001000
IDL> usa = cgExtractShape(file, 'CNTRY_NAME', 'UNITED STATES')
IDL> tic & print, usa -> ContainsPoints(-100, 35) & toc
      1
 Elapsed Time:  0.000000
```

Hard to believe, huh!?

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

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 musch slowier 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

Subject: Re: Point within country boundary
Posted by David Fanning on Sat, 25 Jan 2014 13:36:55 GMT
View Forum Message <> Reply to Message

Fabien writes:

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

I'm not sure that you can even *think* about going for coffee in the
time you save, let alone going to get a cup. ;-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Point within country boundary
Posted by Fabzi on Sat, 25 Jan 2014 14:14:23 GMT
View Forum Message <> Reply to Message

On 25.01.2014 14:36, David Fanning wrote:
> I'm not sure that you can even*think* about going for coffee in the
> time you save, let alone going to get a cup.;-)

well, there is a factor of 1300 between ContainPoints (brute force) and ComputeMask, and a factor of 150 betwen ContainPoints (brute force) and the "smart" ContainPoints (when only the points that passed the mask test are checked).

I do this kind of operation quite often with large images and complex ROI groups (see for example the size of the shapes of the last Randolph Glacier Inventory). Be sure I am glad to have found this trick ;-)

Cheers

Fabien

Subject: Re: Point within country boundary
Posted by Matt[3] on Tue, 28 Jan 2014 11:35:50 GMT
View Forum Message <> Reply to Message

Hi All,

Thanks for the help. This has been very useful. David's cgExtractShape works a treat.

As for speed, I needed to allocate points to specific countries on a relatively fine grid (~1e6 points), so ContainsPoints was far too slow. CalculateMask was much more appropriate.

Cheers,

Matt

As for speed, whilst ContainPoints

On Saturday, 25 January 2014 14:14:23 UTC, Fabien wrote:
> On 25.01.2014 14:36, David Fanning wrote:
>
>> I'm not sure that you can even*think* about going for coffee in the
>
>> time you save, let alone going to get a cup.;-)
>
>
>
> well, there is a factor of 1300 between ContainPoints (brute force) and
>
> ComputeMask, and a factor of 150 betwen ContainPoints (brute force) and

>
> the "smart" ContainPoints (when only the points that passed the mask
>
> test are checked).
>
>
>
> I do this kind of operation quite often with large images and complex
>
> ROI groups (see for example the size of the shapes of the last Randolph
>
> Glacier Inventory). Be sure I am glad to have found this trick ;-)
>
>
>
> Cheers
>
>
>
> Fabien

Subject: Re: Point within country boundary
Posted by Fabzi on Tue, 28 Jan 2014 13:01:49 GMT
View Forum Message <> Reply to Message

Matt,

I think you are aware that they are not doing the same thing.
ComputeMask things "pixels", containPoints thinks "(grid) points".

Cheers

Subject: Re: Point within country boundary
Posted by Fabzi on Tue, 28 Jan 2014 15:05:21 GMT
View Forum Message <> Reply to Message

On 28.01.2014 14:01, Fabien wrote:
> Matt,
>
> I think you are aware that they are not doing the same thing.
> ComputeMask things "pixels", containPoints thinks "(grid) points".
>
> Cheers
>
>

Just for the record ;-), a small programm that illustrates the difference:

```
pro illustrate_contain_points

  grid = cgScaleVector(FINDGEN(5,4),0,254)

  points_X = (FINDGEN(5)+0.5) # (LONARR(4) + 1)
  points_Y = (FINDGEN(4)+0.5) # (LONARR(5) + 1)

  shape_x = [1.2, 3.9, 2.3, 1.2]
  shape_y = [1.2, 2.3, 3.7, 1.2]
  roi = IDLanROI(shape_x, shape_y)

  cgLoadCT, 33 & TVLCT, 130, 130, 130, 255 & cgWindow
  cgImage, grid, /KEEP_ASPECT_RATIO, /AXES, /ADDCMD, TITLE=''
  cgPolygon, shape_x, shape_y, /ADDCMD, COLOR='black', THICK=2
  cgPlotS, points_X, points_Y, PSYM=16, /ADDCMD

  mask = roi->ComputeMask(DIMENSIONS=[5,4], $
    PIXEL_CENTER=[0.5,0.5], MASK_RULE=2)
  grid_ = grid
  grid_[where(mask ne 0)] = 255

  cgLoadCT, 33 & TVLCT, 130, 130, 130, 255 & cgWindow
  cgImage, grid_, /KEEP_ASPECT_RATIO, /AXES, /ADDCMD, $
    TITLE='Compute Mask, RULE=2'
  cgPolygon, shape_x, shape_y, /ADDCMD, COLOR='black', THICK=2
  cgPlotS, points_X, points_Y, PSYM=16, /ADDCMD

  mask = roi->ComputeMask(DIMENSIONS=[5,4], $
       PIXEL_CENTER=[0.5,0.5], MASK_RULE=1)
  grid_ = grid
  grid_[where(mask ne 0)] = 255

  cgLoadCT, 33 & TVLCT, 130, 130, 130, 255 & cgWindow
  cgImage, grid_, /KEEP_ASPECT_RATIO, /AXES, /ADDCMD, $
    TITLE='Compute Mask, RULE=1'
  cgPolygon, shape_x, shape_y, /ADDCMD, COLOR='black', THICK=2
  cgPlotS, points_X, points_Y, PSYM=16, /ADDCMD

  mask = mask * 0
  cp = roi->ContainsPoints(points_X, points_Y)

  cgLoadCT, 33 & TVLCT, 130, 130, 130, 255 & cgWindow
  cgImage, grid, /KEEP_ASPECT_RATIO, /AXES, /ADDCMD, $
    TITLE='Contain Points'
  cgPolygon, shape_x, shape_y, /ADDCMD, COLOR='black', THICK=2
```

```
  cgPlotS, points_X, points_Y, PSYM=16, /ADDCMD
  cgPlotS, points_X[where(cp)], points_Y[where(cp)], $
    PSYM=16, COLOR='dark grey', /ADDCMD

end
```

Subject: Re: Point within country boundary
Posted by David Fanning on Tue, 28 Jan 2014 15:15:54 GMT
View Forum Message <> Reply to Message

Fabien writes:

> Just for the record ;-), a small programm that illustrates the difference:

Oh, that's helpful! (Although some additional explanation might be
needed for the casually motivated reader.) Maybe I'll write an article
about this. Great illustrations. Thanks! :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Point within country boundary
Posted by David Fanning on Tue, 28 Jan 2014 15:20:30 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Fabien writes:
>
>> Just for the record ;-), a small programm that illustrates the difference:
>
> Oh, that's helpful! (Although some additional explanation might be
> needed for the casually motivated reader.) Maybe I'll write an article
> about this. Great illustrations. Thanks! :-)

Oddly, I just came across an Albert Einstein quote that seems to apply
here:

  "The secret to creativity is knowing how to hide your sources."

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Point within country boundary
Posted by David Fanning on Tue, 28 Jan 2014 16:08:36 GMT
View Forum Message <> Reply to Message

Fabien writes:

> I think you are aware that they are not doing the same thing.
> ComputeMask things "pixels", containPoints thinks "(grid) points".

Now that I think about it, another way of making a pixel mask, with just
the country boundaries, might be something like this.

```
cgDisplay, 720, 360
cgErase, 'black'
cgMap_Set, /Cylindrical, Position=[0,0,1,1], /NoBorder, /NoErase
file = Filepath(SubDir=['resource','maps','shape'], "country.shp")
usa = cgExtractShape(file, 'CNTRY_NAME', 'UNITED STATES')
cgDraw_ROI, usa, Color='white'
mask = TVRD() NE 0
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Point within country boundary
Posted by iqbalhabibie0684 on Wed, 22 Feb 2017 09:18:27 GMT
View Forum Message <> Reply to Message

I try to use cgEXTRACTSHAPE from this website

http://www.idlcoyote.com/documents/cg_maps.php#cgEXTRACTSHAP E, but I cant find *.pro. Please send me for extract shapefile. Thanks

---

Subject: Re: Point within country boundary
Posted by Helder Marchetto on Wed, 22 Feb 2017 09:41:57 GMT
View Forum Message <> Reply to Message

On Wednesday, February 22, 2017 at 10:18:28 AM UTC+1, iqbalhab...@gmail.com wrote:
> I try to use cgEXTRACTSHAPE from this website
http://www.idlcoyote.com/documents/cg_maps.php#cgEXTRACTSHAP E, but I cant find *.pro. Please send me for extract shapefile. Thanks

I don't have it, but if you click on the link http://www.idlcoyote.com/idldoc/forsale/cgextractshape.html you notice that it says on the top "Coyote Graphics Routines For Sale". So my guess, is that you need to pay for it. I think that the prices are quite fair, so you might just email David and find out how much it costs.

cheers
