Subject: Re: optimization; which point falls into a polygon
Posted by Maarten[1] on Wed, 23 Sep 2009 11:38:22 GMT
View Forum Message <> Reply to Message

On Sep 22, 11:57 pm, Klemen <klemen.zak...@gmail.com> wrote:

> I am working with geostationary satellite data and some GIS rasters.
> The satellite data do not have a regular sampling (pixel has a form of
> a parallelogram) but I know corner coordinates of each pixels.
>
> I would like to do some kind of spatial join - I would like to know
> which of my pixels in GIS rasters (regularly sampled) fall into each
> satellite pixel – I would like to create a raster where each GIS pixel
> contains an index of the corresponding satellite index.

[snip]

> Does anybody have a suggestion?

Approach from the other direction.

The IDLanROI class can do this for you. Loop over all Satellite pixels, and create an IDLanROI object for it, using the corner coordinates. Use the ContainsPoints method to figure out which GIS raster point fall in this Satellite grid cell. ContainsPoints is array aware. Use where on the result, and set the result in your destination array to the intended value.

Caveats: Since you are geostationary, you won't have to deal with the poles, but the dateline might be an issue. The correct way is to do everything in Cartesian coordinates, but I never bothered.

Maarten

Subject: Re: optimization; which point falls into a polygon
Posted by Jeremy Bailin on Wed, 23 Sep 2009 11:46:06 GMT
View Forum Message <> Reply to Message

Something akin to JD's match_2d might work... it would at least tell you the closest corner of a satellite pixel for each GIS point, and is probably faster than building a triangulation.

Is there an analytic expression for the satellite pixel corners, or do you really have to go from the data? I'm trying to figure out if you can directly compute it...

-Jeremy.

On Sep 22, 5:57 pm, Klemen <klemen.zak...@gmail.com> wrote:
> Hi,
>
> I am working with geostationary satellite data and some GIS rasters.
> The satellite data do not have a regular sampling (pixel has a form of
> a parallelogram) but I know corner coordinates of each pixels.
>
> I would like to do some kind of spatial join - I would like to know
> which of my pixels in GIS rasters (regularly sampled) fall into each
> satellite pixel – I would like to create a raster where each GIS pixel
> contains an index of the corresponding satellite index. This can be
> easily done using 4 for loops when the GIS raster is small. However, I
> would like to do this on a 1000 * 1000 large GIS layer. Satellite data
> have factor 3 less pixels in one direction and factor 6 less pixels in
> the other direction. Such a way is then really time consuming.
>
> How to do it faster? I was reading of triangulation, but this would
> probably mean that I would have to triangulate each GIS raster point
> with satellite centre points and then check in which of four closest
> satellite pixels falls the raster pixel. But I can imagine, that
> building a triangulation for a million times is also not really fast.
>
> Does anybody have a suggestion?
>
> Thank you in advance!
> Klemen

Subject: Re: optimization; which point falls into a polygon
Posted by Klemen on Wed, 23 Sep 2009 12:11:39 GMT
View Forum Message <> Reply to Message

Thank you both for suggestions, I will try both.

> Is there an analytic expression for the satellite pixel corners, or do
> you really have to go from the data? I'm trying to figure out if you
> can directly compute it...

Well, I have a layer containing latitude and longitude of each center
point - from these coordinates I estimate the corners positions.

Subject: Re: optimization; which point falls into a polygon
Posted by hethomas on Wed, 23 Sep 2009 15:08:58 GMT
View Forum Message <> Reply to Message

I did something similar a while ago for non-geostationary satellite data..... I found the corner coordinates and used these to draw lines around each pixel. I could then find out which points landed within each pixel by using gt or lt expressions for the lines as compared to the points.

Subject: Re: optimization; which point falls into a polygon
Posted by penteado on Wed, 23 Sep 2009 15:56:51 GMT
View Forum Message <> Reply to Message

On Sep 23, 9:11 am, Klemen <klemen.zak...@gmail.com> wrote:
> Thank you both for suggestions, I will try both.
>
>> Is there an analytic expression for the satellite pixel corners, or do
>> you really have to go from the data? I'm trying to figure out if you
>> can directly compute it...
>
> Well, I have a layer containing latitude and longitude of each center
> point - from these coordinates I estimate the corners positions.

See the help on map_proj_forward and map_proj_inverse. They may be useful to convert between latitude/longitude and pixel values, depending on how your GIS rasters are sampled. ROIs as suggested above are a more general way to find if points are in polygons, and are useful in more situations, so I do not know which way will be more appropriate for your case.

Subject: Re: optimization; which point falls into a polygon
Posted by Klemen on Fri, 25 Sep 2009 09:13:04 GMT
View Forum Message <> Reply to Message

> See the help on map_proj_forward and map_proj_inverse. They may be
> useful to convert between latitude/longitude and pixel values,

I process already projected data (Lambert conformal projection). I am also trying to write a shape file at the same time, so I can check each pixel to be sure.

I have tried the suggested solutions and I think the optimal for me is ROI way. It took 27s to get the results on a layer with 200 by 200 pixels; geostatioary image has in this case 74 * 45 pixels.

I have tried also JD's match_2d to get the closest point and then I wanted to proceed by approach suggested by hethomas, however, it took me 45s just to compute the closest satellite centre point (without

checking if GIS point really fits into the polygon)...

But I would have gone for such a similar approach if I hadn't asked you for an advice before. So thank you all.