Subject: Re: Search and Replace in 2D Array (Image)
Posted by David Fanning on Thu, 28 May 2009 19:28:08 GMT
View Forum Message <> Reply to Message

vikramivatury@gmail.com writes:

```
>  I have an image "m" which is [1229 x 1229] pixels that contains 1's
>  and 0's. I also have an image "img" which is [1229 x 1229] that
>  contains pixel IDs. For every 1 in image "m" it corresponds to a
>  certain pixel ID in image "img". I am trying to write a loop in IDL
>  that scans through the pixels of image "m" and when its get to a 1,
>  spit out its information from "img". I used the 'where' function (w =
>  where m gt 1.........ids = img(w)) and that worked fine in outputting
>  a 1D array of 1510441 pixels, but I am trying to do it for a 2D array
>  using loops.
>
>  img = read_tiff('NP.tif')
>  ids = fltarr(1229,1229)
>
>  for i = 0,1228 do begin
>    for j = 0,1228 do begin
>     if (m(i,j) eq 1.) then begin
>      ids(i,j) = img(m(i,j))
>     endif
>    endfor
>   endfor
>
>  Any suggestions would be great.
```

Well, don't do it like this. :-)

I suggest you investigate LABEL_REGION. *Much* faster!

Cheers,

David

--
David Fanning, Ph.D.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Search and Replace in 2D Array (Image)
Posted by vikramivatury on Thu, 28 May 2009 19:39:43 GMT
View Forum Message <> Reply to Message

On May 28, 3:28 pm, David Fanning <n...@dfanning.com> wrote:
> vikramivat...@gmail.com writes:
>> I have an image "m" which is [1229 x 1229] pixels that contains 1's
>> and 0's. I also have an image "img" which is [1229 x 1229] that
>> contains pixel IDs. For every 1 in image "m" it corresponds to a
>> certain pixel ID in image "img". I am trying to write a loop in IDL
>> that scans through the pixels of image "m" and when its get to a 1,
>> spit out its information from "img". I used the 'where' function (w =
>> where m gt 1.........ids = img(w)) and that worked fine in outputting
>> a 1D array of 1510441 pixels, but I am trying to do it for a 2D array
>> using loops.

```
>
>>  img = read_tiff('NP.tif')
>>  ids = fltarr(1229,1229)
>
>>  for i = 0,1228 do begin
>>            for j = 0,1228 do begin
>>                   if (m(i,j) eq 1.) then begin
>>                          ids(i,j) = img(m(i,j))
>>                   endif
>>            endfor
>>     endfor
>
```

>> Any suggestions would be great.
>
> Well, don't do it like this. :-)
>
> I suggest you investigate LABEL_REGION. *Much* faster!
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Coyote's Guide to IDL Programming (www.dfanning.com)
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Ok, I will take a look at Label_Regions.

I was just thinking....Is it possible to do something like this
without an if-statment:

```
img = read_tiff('NP.tif')
ids = fltarr(1229,1229)
w = fltarr(1229,1229)

for i = 0,1228 do begin
```

```
        for j = 0,1228 do begin
                w(i,j) = where m eq 1
                ids(i,j) = img(w(i,j))

        endfor
endfor
```

Subject: Re: Search and Replace in 2D Array (Image)
Posted by Jean H. on Thu, 28 May 2009 20:19:41 GMT
View Forum Message <> Reply to Message

vikramivatury@gmail.com wrote:
> Hello,
>
> I have an image "m" which is [1229 x 1229] pixels that contains 1's
> and 0's. I also have an image "img" which is [1229 x 1229] that
> contains pixel IDs. For every 1 in image "m" it corresponds to a
> certain pixel ID in image "img". I am trying to write a loop in IDL
> that scans through the pixels of image "m" and when its get to a 1,
> spit out its information from "img". I used the 'where' function (w =
> where m gt 1.........ids = img(w))

should be GE or EQ, not GT, or you will never have the ones selected

and that worked fine in outputting
> a 1D array of 1510441 pixels, but I am trying to do it for a 2D array
> using loops.
>
> img = read_tiff('NP.tif')
> ids = fltarr(1229,1229)
>
> for i = 0,1228 do begin
>   for j = 0,1228 do begin
>   if (m(i,j) eq 1.) then begin

Be careful here. Don't use a float if your data is byte/integer. If it
is of type float, you must do "where X-1.0 lt epsilon"... read David
Fanning's "the sky is falling" article

>   ids(i,j) = img(m(i,j))

This will always be equal to 1. Are you sure you want to subset m here?

>   endif
>   endfor
> endfor

>
> Any suggestions would be great.
>
> Thanks,
> Vikram

"where" remains your friend here.

```
img = read_tiff('NP.tif')
ids = fltarr(1229,1229)
OneIDX = where(m eq 1)
ids[oneIDX] = img[oneIDX]
```

If you really want, you can convert back the 1D index to 2D with
ARRAY_INDICES, but it changes nothing to the above.

Jean

Subject: Re: Search and Replace in 2D Array (Image)
Posted by vikramivatury on Thu, 28 May 2009 20:34:27 GMT

View Forum Message <> Reply to Message

On May 28, 4:19 pm, "Jean H." <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
wrote:
> vikramivat...@gmail.com wrote:
>> Hello,
>
>> I have an image "m" which is [1229 x 1229] pixels that contains 1's
>> and 0's. I also have an image "img" which is [1229 x 1229] that
>> contains pixel IDs. For every 1 in image "m" it corresponds to a
>> certain pixel ID in image "img". I am trying to write a loop in IDL
>> that scans through the pixels of image "m" and when its get to a 1,
>> spit out its information from "img". I used the 'where' function (w =
>> where m gt 1.........ids = img(w))
>
> should be GE or EQ, not GT, or you will never have the ones selected
>
> and that worked fine in outputting
>
>> a 1D array of 1510441 pixels, but I am trying to do it for a 2D array
>> using loops.
>

```
>>  img = read_tiff('NP.tif')
>>  ids = fltarr(1229,1229)
>
>>  for i = 0,1228 do begin
>>            for j = 0,1228 do begin
>>                    if (m(i,j) eq 1.) then begin
>
> Be careful here. Don't use a float if your data is byte/integer. If it
> is of type float, you must do "where X-1.0 lt epsilon"... read David
> Fanning's "the sky is falling" article
>
>>                            ids(i,j) = img(m(i,j))
>
> This will always be equal to 1. Are you sure you want to subset m here?
>
>>                    endif
>>            endfor
>>    endfor
>
>> Any suggestions would be great.
>
>> Thanks,
>> Vikram
>
> "where" remains your friend here.
>
> img = read_tiff('NP.tif')
> ids = fltarr(1229,1229)
> OneIDX = where(m eq 1)
> ids[oneIDX] = img[oneIDX]
>
> If you really want, you can convert back the 1D index to 2D with
> ARRAY_INDICES, but it changes nothing to the above.
>
> Jean
```

Great! Thats exactly what I wanted. Thanks Jean.

-Vikram