Nuno Oliveira wrote:
> I'm almost shooting myself! Every time I think I'm reaching a solution I
> have a new problem! And this one is really something.
>
>
>
> The issue is that if find a boundary of ROI using Label Region before I don'
> t get the pixels that are on the borders of the image. This is really
> strange, I don't what Label Region does and even I can't find the code (is
> it in the IDL library? I don't find it!)
>

Hello,

Yes, it is a bit aggravating. The Label_Region function treats the
boundary of the image as background. Here's the blurb on the input
argument from the documentation....

" A n-dimensional image to be labeled. Data is converted to integer type
if necessary. Pixels at the edges of Data are considered to be zero."

The function below will work for 2d images.

FUNCTION MY_LABEL_REGION, data, _Extra = extra

Sz = Size(data)
dim = Sz[1:Sz[0]]
d = Make_Array(dim+2, value = 0, type = Sz[Sz[0]+1])

d[1,1] = data
mask = LABEL_REGION(d, _Extra = extra)

Return, mask[1:dim[0], 1:dim[1]]
END

If you have a lot of blobs within an image, you might prefer to use the
HISTOGRAM function with REVERSE_INDICES to navigate around the blobs.
It will save repeated calls to WHERE which has to search the entire array.

Ben

Thanks for the solution, Ben. Well, it's little bit embarrassing to me. I've read the description of the Label_Region function, but obviously not that careful.

I'm not sure if I understood your suggestion of using histogram function. I read the description of the function and perhaps I still don't get it.

If I make

hist = HISTOGRAM(data, REVERSE_INDICES=R)

To get de indices for first region defined by label region (the indices that are equal to 1) I should do... (according to the reference guide.)

i = 1

blob = data(R[R(i):R(i+1)-1])

Is that it? If I haven't made any mistake I'm trying this and I get nothing back (or I get strange array with two indices when should have much more)

Cheers,

Nuno.

"Ben Tupper" <btupper@bigelow.org> wrote in message
news:c0b9sg$15875l$1@ID-189398.news.uni-berlin.de...
>
> Hello,
>

> Yes, it is a bit aggravating. The Label_Region function treats the
> boundary of the image as background. Here's the blurb on the input
> argument from the documentation....
>
> " A n-dimensional image to be labeled. Data is converted to integer type
> if necessary. Pixels at the edges of Data are considered to be zero."
>
> If you have a lot of blobs within an image, you might prefer to use the
> HISTOGRAM function with REVERSE_INDICES to navigate around the blobs.
> It will save repeated calls to WHERE which has to search the entire array.
>
> Ben
>

Subject: Re: Label Region trouble
Posted by Nuno Oliveira on Wed, 11 Feb 2004 15:09:37 GMT

Seems that I need to reboot...

Guess what. I tried

blob = R[R(i):R(i+1)-1] instead of

blob = data(R[R(i):R(i+1)-1])

and. it worked!

Cheers,

Nuno.

P.S. Only realize that it was this way when I printed out the variable and
saw that it was an array full of one. then suspected I was doing something
wrong, L.

"Nuno Oliveira" <nmoliveira@fc.ul.pt> wrote in message
news:c0de3o$1lsp$1@pegasus.fccn.pt...
> I'm not sure if I understood your suggestion of using histogram function.
I
> read the description of the function and perhaps I still don't get it.
>
> If I make
>
> hist = HISTOGRAM(data, REVERSE_INDICES=R)
>
> To get de indices for first region defined by label region (the indices
that
> are equal to 1) I should do... (according to the reference guide.)
>
> i = 1
>
> blob = data(R[R(i):R(i+1)-1])
>
> Is that it? If I haven't made any mistake I'm trying this and I get
nothing
> back (or I get strange array with two indices when should have much more)
>
> Cheers,
>
> Nuno.

Nuno Oliveira wrote:
> Thanks for the solution, Ben. Well, it's little bit embarrassing to me. I've
> read the description of the Label_Region function, but obviously not that
> careful.
>
>
Hi,

You shouldn't be embarassed. I have been there, too! I think if you
searched the Google database for this ("label_Region" and "Edge") you
find that I *really* have been there. There maybe other, faster, ways
of doing this.

The only thing you should be embarassed about is using CW_DEFROI (just
kidding!)

>
> I'm not sure if I understood your suggestion of using histogram function. I
> read the description of the function and perhaps I still don't get it.
>

Yeah, well, only JD really knows how histogram works - I just ape the
same steps over and over.   You should look at JD's tutorial on
HISTOGRAM. David F hosts this for the benefit of all at
http://www.dfanning.com/documents/tips.html#Tutorials

> hist = HISTOGRAM(data, REVERSE_INDICES=R)
>
>
>
> To get de indices for first region defined by label region (the indices that
> are equal to 1) I should do... (according to the reference guide.)
>
>
>
> i = 1
>
> blob = data(R[R(i):R(i+1)-1])
>
> Is that it? If I haven't made any mistake I'm trying this and I get nothing
> back (or I get strange array with two indices when should have much more)

Well, I'm not sure what is happening. I suggest that you make sure that
data is the labeled image (not the original). Also, note the following...

i = 1

hist[i] = number of pixels in bin number i
 (in this case i is also the pixel values)

indices = R[R[i]:R[i+1]-1] = data indices for all of the pixels in bin i

blob = data[indices] = all of the pixels in bin i
 (in this case they should all have a value of i)

Ben