Subject: Re: area threshold
Posted by gunvicsin11 on Mon, 08 May 2017 10:52:21 GMT
View Forum Message <> Reply to Message

On Wednesday, May 3, 2017 at 9:34:07 PM UTC+5:30, Dick Jackson wrote:
> On Wednesday, 3 May 2017 05:55:12 UTC-7, Helder wrote:
>> On Wednesday, May 3, 2017 at 7:55:14 AM UTC+2, sid wrote:
>>> On Tuesday, May 2, 2017 at 2:53:16 PM UTC+5:30, Nikola Vitas wrote:
>>>> On Monday, May 1, 2017 at 11:37:55 AM UTC+1, gunvi...@gmail.com wrote:
>>>> > On Monday, May 1, 2017 at 3:29:49 PM UTC+5:30, Nikola Vitas wrote:
>>>> > > On Monday, May 1, 2017 at 9:00:01 AM UTC+1, sid wrote:
>>>> > > > hi all,
>>>> > > > can anyone give some ideas for giving area threshold. Right now I am using
>>>> > > > label_region, histogram and array_indices. Inorder to discard a region which is of n x n pixel area. But in this method some pixels are left without discarding and if I go for higher pixel area then the useful data also gets discarded.
>>>> > > > Therefore anyone has any better idea for area thresholding.
>>>> > > > thanks
>>>> > > >
>>>> > > How is your n x x patch defined? By the values it contain or by position in a bigger image?
>>>> > > >
>>>> > > Anyway, check for the logical operators, EQ, NE, GT, LT and similar and for the function WHERE. It's all in the IDL help files including useful examples.
>>>> > >
>>>> > My n x n patch are defined by a position in a bigger image.
>>>> >
>>>> You should specify your problem more specifically. What do you want to get as a result? Some statistics of the rest of the image or you just need to display the image with this patch masked out.
>>>
>>> I have got a binary image in that the region which is less than 10 x 10 pixels has to be discarded or shud be made zero from the patch. I want to estimate the total area of the rest of the region of the patch.
>>
>> I think Nikola is right. You should better explain your problem. From what you wrote, here is what I understood. You want to exclude from your analysis a patch of nxn pixels. Depending on the analysis, there are different solutions. Since you mentioned the total area of the rest of the region (without the patch) then I would:
>> 1) calculate area of patch: area_patch = n*n
>> 2) calculate area of image: area_image = nCols*nRows
>> 3) rest area is then: area_rest = area_image - area_patch
>>
>> My guess though, is that this is not what you asked, so you should really explain better. Eventually post an image online somewhere to show the problem in greater detail.
>>
>> Regards,
>> Helder

>
> Hi all,
>
> I'm willing to make a few guesses as to what sid is looking for (sorry, your use of "region" and "patch" wasn't clear to me). This code shows how to identify only the blobs in an image which can contain at least a 10 x 10 pixel region.
>
> I hope this is helpful, sid!
>
> Cheers,
> -Dick
>
> Dick Jackson Software Consulting Inc.
> Victoria, BC, Canada --- http://www.d-jackson.com
>
>
> PRO FindBigBlobs
>
> ; Read a greyscale image with blobs of various intensities
> img = Read_Image(Filepath('mineral.png', SUBDIRECTORY=['examples', 'data']))
> imgDims = Size(img, /DIMENSIONS)
>
> ; Make a binary image with several blobs and speckles
> bImg = img GT 190B
>
> ; Make a binary image of only blobs that include a 10 x 10 region
> bigBlobsImg = Morph_Open(bImg, Replicate(1B, [10, 10]))
>
> ; Starting from those blob pixels, expand to all connected pixels in bImg
> ; (result is vector of indices of pixels where this result expanded to)
> whGrownRegion = Region_Grow(bImg, Where(bigBlobsImg))
>
> ; Make a new binary image with these pixels set to 1B
> newBImg = BytArr(imgDims)
> newBImg[whGrownRegion] = 1B
>
> ; Display images to show processing
>
> Window, /FREE, XSIZE=imgDims[0]*3, YSIZE=imgDims[1]*2
> TV, img, 0 ; Original greyscale image
> TVScl, bImg, 1 ; Binary image
> TVScl, bigBlobsImg, 2 ; Detected big blobs
> TVScl, bImg+bigBlobsImg, 3 ; Show detected big blobs overlaid on bImg
> TVScl, newBImg, 4 ; Extended blobs from binary image
>
> ; Compute pixel statistics
>
> Print, 'Pixels from big blobs: ', Total(newBImg, /INTEGER)

> Print, 'Total image pixels: ', Product(imgDims, /INTEGER)
>
> END

Thanks a lot for the code. This is what i wanted, but my blob area is changing, it is not a constant 10 x 10 region always.
Is there a way to find the best possible size of blob region by some means.
