Subject: Re: spots on image
Posted by Brian Larsen on Thu, 25 Jan 2007 21:22:23 GMT
View Forum Message <> Reply to Message

I have done a bit of similar work. Just in quick pseudocode

dat = fltarr(256,256)
;; fill with an image
;; find one of the pinhole brightnesses using your method
;; call that center[2] [0]->x [1]->y
;; take one of the points and form a sub-array around that larger than the pinhole, call the pinhole 4 pixels max
sub_arr = dat[center[0]-5:center[0]+5, center[1]-5:center[1]+5] ; make sure you have grabbed a large enough chunk but small enough not to grad two pinholes
;; use gauss2dfit on the chunk Result = GAUSS2DFIT(Z, A [, X, Y] [, /NEGATIVE] [, /TILT])
res = gauss2dfit(sub_arr, a)
;; then get rid off all the indices in subarr from the indices you found in your original where statement with another where

Poof done. Of course it will be more work as you iron out the details but that should get you started.

Make and sense?

Brian

---------- ----------
Brian A. Larsen
Dept. of Physics
Space Science and Engineering Lab (SSEL)
Montana State University - Bozeman
Bozeman, MT 59717

On Jan 25, 2:03 pm, "rpert...@gmail.com" <rpert...@gmail.com> wrote:
> Hello,
> I am new to IDL so bear with me! I am analyzing an image with some
> spots on them (from a few pinholes). I need to know where these spots
> are and mark them and make a Gaussian fit. I have used the where
> function to find where the spots are and have obtained the pixels where
> the intensity is higher than a threshold. However, the spots are made
> up of more than 1 pixel, therefore i need to group these pixels
> together to make up one spot, and then fit a Gaussian...
>
> Any ideas how to do that?

>
> Thanks!
>
> RP

Subject: Re: spots on image
Posted by Michael Galloy on Thu, 25 Jan 2007 21:35:35 GMT
View Forum Message <> Reply to Message

On Jan 25, 2:22 pm, "Brian Larsen" <balar...@gmail.com> wrote:
> I have done a bit of similar work. Just in quick pseudocode
>
> dat = fltarr(256,256)
> ;; fill with an image
> ;; find one of the pinhole brightnesses using your method
> ;; call that center[2] [0]->x [1]->y

For this part, I would do something like:

```
nColumns = 256   ; the number of columns in your image
mask = dat ge threshhold
regions = label_region(mask)
for r = 1L, max(regions) do begin
  ind = where(regions eq r, count)
  center = [mean(ind mod nColumns), mean(ind / nColumns)]
endfor
```

then continue on with Brian's fitting code.

Mike
--
www.michaelgalloy.com
