Subject: Re: Is there an automated way to estimated FWHM on 2-D image
Posted by Jeremy Bailin on Mon, 21 Sep 2009 02:41:53 GMT
View Forum Message <> Reply to Message

On Sep 20, 9:12 am, wlandsman <wlands...@gmail.com> wrote:
> On Sep 20, 8:10 am, Jeremy Bailin <astroco...@gmail.com> wrote:
>
>> On Sep 19, 9:02 pm, John Shaw <jds...@udel.edu> wrote:
>
>>> I was wondering if anyone had a routine for estimating the full-width-
>>> at-half-maximum (FWHM) of possible point sources in a 2-D array. Most
>>> of the routines I have found and examined request the FWHM for a
>>> guassian to be convolved to find the sources.
>
>> Would PKFIT in the IDL astronomy library give you what you need? You
>> can get the Gaussian dispersion out, which is easy to convert into a
>> FWHM.
>
> I would just use any Gaussian-2d fitting routine, such as
> gauss2dfit.pro in the ITTVIS library, or (preferably) the Gaussian
> option of the mp2dfitfun.pro function in Craig Markwardt's fitting
> library (http://www.physics.wisc.edu/~craigm/idl/fitting.html).
>
> One thing to be careful of is the choice of the fitting region
> size. We don't observe Gaussians in real life, and for example,
> star images have very extended wings. If your fitting region
> includes the far wings, then your derived FWHM will be strongly biased
> (especially since there are many more pixels in the wings). A
> general rule is that the fitting region should be the size of the
> FWHM. Since the FWHM is what you are trying to determine, you
> might have to iterate. (So if using a 5x5 box gives you a FWHM =
> 1.8, then you might want to recompute it using a 3x3 box.
>
> The pkfit.pro procedure does have some advantages: (1) it fits a
> Gaussian convolved with the pixel size rather than just a Gaussian,
> and (2) it gives lower weight to pixels far from the centroid,and (3)
> it iterates to choose the best (circular) fitting radius, among 3, 5,
> and 7 pixels. But it is very old and ugly code (circa 1988), and
> does a lot of extraneous calculations since it is part of a larger
> fitting package.
>
> Finally, note that if you are only using the FWHM as input to a source
> detection algorithm, then it does not need to be very accurate. --
> Wayne

Of course, another issue is what if your PSF doesn't even remotely
look Gaussian, even within the FWHM. The nice thing about the FWHM is
that it's pretty well-defined for any declining profile... so you

could do something like this (UNTESTED):

```
; we want to find FWHM of image within a box of size "width" around
"x0", "y0":
pixelvalues = image[x0-0.5*width:x0+0.5*width,
y0-0.5*width,y0+0.5*width]
nbox = n_elements(pixelvalues)
pixelcoords = array_indices(pixelvalues, lindgen(nbox)) + rebin
([x0,y0],2,nbox)-0.5*width
pixelradii2 = total(pixelcoords^2, 1)

; get a smoothed version of pixelvalues to get the average profile.
use a boxcar
; of width 5 as a wild guess. this part of the code could be a lot
smarter.
sortedradii = sort(pixelradii2)
smoothed_pixelvalues = smooth(pixelvalues[sortedradii], 5)

; find the half-max point
maxvalue = max(pixelvalues)
halfmaxpoint = sqrt(interpol(pixelradii2, smoothed_pixelvalues,
0.5*maxvalue))

fwhm = 2. * halfmaxpoint
```

-Jeremy.
