
Subject: Re: Image analysis and ring identification
Posted by [Dan Larson](#) on Thu, 18 Apr 2002 15:37:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <MPG.172872a05bbf04f3989895@news.frii.com>, david@dfanning.com says...

> Rachel Pepper (Rachel_Pepper@brown.edu) writes:

>

>> Actually, the rings often have a bright spot that is not in the center of
>> the ring, so I don't think the centroiding technique will work. Do you have
>> any other suggestions?

>

> I think Paul's suggestions are right on the money,
> but I wouldn't give up on the centroid theory too
> quickly. The wonderful thing about centroids is that
> they are really center of mass calculations. So if
> the values you total are actual image values,
> then the centroid will naturally zero in on
> bright spots, since they are "heavier" in the
> calculations.

>

> Cheers,

>

> David

>

I have seen this issue arise a number of times:
what is the best way to find the center of a
sub-resolution object?

A sub-resolution object will appear in the image
as a point spread function that is well
approximated by a Gaussian. However, as Paul noted,
one is usually dealing with noisy data, and the
Gaussian is rarely smooth. Centroid approaches
work well in this situation.

I use an algorithm which is intermediate
in complexity between a centroid (essentially
a binary mask) and a full non-linear least squares
fit to a Gaussian. This algorithm is a centroid-like
approach that uses an error function to obtain the mask.
It is iterative and therefore slower than a simple centroid,
but it is much more accurate and works well in noisy,
pixelated environments. This 'Gaussian mask' algorithm is
way more robust than a full non-linear least squares fit
(say Levenberg-Marquadt).

My implementation is below. There is some other junk in the

code, but I think the idea is clear.

Cheers,

Dan

```
FUNCTION gmask_fit, spot, psf_width, black_level, x_dim, y_dim

; gmask_fit calculates the center of a psf using an iterative gaussian
mask
; (Thompson, Larson, Webb, Biophysical Journal, in Press)
;
;
; Dan Larson
; 4/18/02

x0=0.0
y0=0.0
pi=3.1415926
F=1.0/(sqrt(2.0)*psf_width)
gauss_mask = dblarr(x_dim, y_dim)
error = 0.0
results=dblarr(3)

; offset correction
image=spot-black_level

; boundary condition. border is set to zero
image[0, *]= 0.0
image[x_dim-1, *]= 0.0
image[, 0]= 0.0
image[, y_dim-1] = 0.0

; easy localisation by finding the centroid of the image

center=fast_centroid(image)
x0=center[0]
y0=center[1]

; iterative centroid calculation with gaussian mask
h = 1.0e-8 ; tolerance
diff_x=0.0
diff_y=0.0
repeat_index=0

x_dim = long(x_dim)
y_dim = long(y_dim)
array=lindgen(x_dim, y_dim)
xarr=array mod x_dim
```

```
yarr=array/x_dim
```

```
REPEAT begin
```

```
x0=x0 + diff_x/2.0
```

```
y0=y0 + diff_y/2.0
```

```
a=F*(yarr - 0.5 - y0)
```

```
b=F*(yarr + 0.5 - y0)
```

```
c=F*(xarr - 0.5 - x0)
```

```
d=F*(xarr + 0.5 - x0)
```

```
gauss_mask =0.25*(errorf(a)-errorf(b))*(errorf(c)-errorf(d))
```

```
sum=total(image*gauss_mask)
```

```
trial_x0=total(xarr*image*gauss_mask)
```

```
trial_y0=total(yarr*image*gauss_mask)
```

```
diff_x = trial_x0/sum - x0
```

```
diff_y = trial_y0/sum - y0
```

```
;print, diff_x, diff_y
```

```
repeat_index=repeat_index+1
```

```
endrep until (abs(diff_x LT h)) and (abs(diff_y) LT h) or (repeat_index  
GT 300)
```

```
if (repeat_index GT 300) then begin
```

```
print, "GMASK ITERATION MAXED OUT (number of iterations=",
```

```
repeat_index, ")"
```

```
results[2]=0.0
```

```
return, results
```

```
endif
```

```
if (repeat_index LE 300) then print, "gmask_fit convergence satisfied
```

```
(number of iterations=", repeat_index, ")"
```

```
; photon number calc
```

```
sum = total(gauss_mask*gauss_mask)
```

```
N = total(image*gauss_mask)
```

```
photon_number = N/sum;
```

```
results[0]=x0
```

```
results[1]=y0
```

```
results[2]=photon_number
```

```
return, results
```

```
end
```