Subject: Re: Routine to return indices for circle in R2
Posted by davidf on Mon, 08 Sep 1997 07:00:00 GMT
View Forum Message <> Reply to Message

David Foster writes:

> Has someone written, seen or heard of a routine that will return
> the indices of a circle within an image of given dimensions,
> given the radius and center coordinate of the circle? I vaguely
> remember seeing this in a newsgroup post some time ago.
>
> For an image of size NX x NY, with the circle at coordinate (Cx,Cy)
> and radius R, I came up with:
>
> indices = lindgen(long(NX)*NY)
> yc = indices / NY
> xc = indices - (yc * NY)
> circle = where(sqrt((xc - Cx)^2 + (yc-Cy)^2) le R)
>
> Can anyone suggest a method that is (a) faster, (b) more clever,
> (c) more elegant, or (d) uses less memory. I'd be willing to settle
> for just one of the above!

There is an article on this very topic (with code) on my web page. ;-)

  http://www.dfanning.com/tips/make_circle.html

The best program for circles I've used is Wayne Landsman's TVCircle.
You can find a link to it from the page, but here it is:

  http://idlastro.gsfc.nasa.gov/ftp/pro/tv/tvcircle.pro

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
Customizable IDL Programming Courses
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com

Subject: Re: Routine to return indices for circle in R2
Posted by John Votaw on Tue, 09 Sep 1997 07:00:00 GMT
View Forum Message <> Reply to Message

David Foster wrote:
>
> Has someone written, seen or heard of a routine that will return
> the indices of a circle within an image of given dimensions,
> given the radius and center coordinate of the circle? I vaguely
> remember seeing this in a newsgroup post some time ago.
>
> For an image of size NX x NY, with the circle at coordinate (Cx,Cy)
> and radius R, I came up with:
>
> indices = lindgen(long(NX)*NY)
> yc = indices / NY
> xc = indices - (yc * NY)
> circle = where(sqrt((xc - Cx)^2 + (yc-Cy)^2) le R)
>
> Can anyone suggest a method that is (a) faster, (b) more clever,
> (c) more elegant, or (d) uses less memory. I'd be willing to settle
> for just one of the above!
>
> Thanks!
>

Here is one idea: define the edge of the circle and then use polyfillv
to return all indicies inside the circle. It definately uses less
memory than the above and depending on the image size, could run much
faster.

```
Nang=20  ;number of verticies
ang=findgen(nang)*!pi*2./nang
x=r*cos(ang)+cx+.5  ;.5 needed for rounding
y=r*sin(ang)+cy+.5
circle=polyfillv(x,y,Sx,Sy)
```

where Sx and Sy are defined in the polyfillv manual entry.
Nang determines how smooth the circle is. Larger circles require larger
Nang.

Good Luck,

John R. Votaw, Ph.D.
Associate Professor of Radiology
Emory Center for Positron Emission Tomography
votaw@commander.eushc.org
voice: (404)712-7954 FAX: (404)712-7962