Subject: Re: making a circle of certain values
Posted by Loren Anderson on Tue, 02 Oct 2007 15:24:07 GMT
View Forum Message <> Reply to Message

On Oct 1, 4:21 pm, Jean H <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
wrote:
> rpert...@gmail.com wrote:
>> Hi,
>> I have a set of data points (x,y coordinates) that can be plotted on
>> my 1240,1024 array. However, they are sparse x,y spots across the
>> image, and I want to 'thicken' it by drawing a circle with my x,y as
>> centers. I want to give it a certain radius and a certain value too,
>> so that the pixels in the circle (the filling) have values and can
>> thus contribute to my rgb image...
>> anyone know how to draw a circle, and assign a value to pixels within
>> that circle?
>
>> Thanks!
>> RP
>
> Hi,
>
> I am not sure if you want to display the circle or not. If not, you can
> compute the distances from your pixel to every other pixels, then select
> only the cells that are close enough. Here is a code I wrote a long time
> ago (might not be the most optimized one, but it works well!)
>
> So, you have an image of 100*200:
> distances = distanceInMatrix(image, PointX,PointY, 100)
> circle = where(distances le radius)
> image[circle] = newValue
>
> Jean.
>
> ;This function compute for every point in the array the distance to the
> origine point.
> ;INPUT: indexCells: a 1D or 2D array of coordinate, for which the
> ;distances will be computed.
> ;   xPos and yPos: the position of the origine point
> ;   x_size: the size of the matrix (number of columns)
> ;OUTPUT: a float array of distances to the origine point
> ;
> ;Author: Jean-Gabriel Hasbani
> ;   jghas...@DELETETHIS.ucalgary.ANDTHIS.ca
> ;   September 2005
>
> function distanceInMatrix, indexCells, xPos,yPos, x_size
>   ;print, "the index", indexCells

```
>         numberOfDistances = N_elements(indexCells)
>
>         ;get the X;Y coordinate of the points.
>         coordCells = ulonarr(2,numberOfDistances)
>         coordCells[1,*] = indexCells[*] / x_size                        ;Y
>         coordCells[0,*] = indexCells[*] - x_size * coordCells[1,*]      ;X
>
>         distances = fltarr(numberOfDistances)
>         distances[*] = sqrt((xPos*1.0 - coordCells[0,*]*1.0)^2+(yPos*1.0-
>  coordCells[1,*]*1.0)^2)
>
>         ;print, coordCells
>         ;print, "In the distance Matrix:", distances
>         return, distances
>  end
```

I'm not sure if this is what you need, but here are the guts of the tvcircle routine that David suggested:

```
FUNCTION Circle, radius, xcenter, ycenter, NPoints=NPoints
; Returns the x and y values of a circle

IF N_Elements(NPoints) EQ 0 THEN NPoints=100

seeds = Findgen(npoints)/(npoints-1)*2*!pi
xvals = sin(seeds)*radius+xcenter
yvals = cos(seeds)*radius+ycenter

RETURN, Transpose([[xvals], [yvals]])
END
```

Unlike, tvcircle, this doesn't plot the values, is just returns them in an array that you can plot yourself. If you need particular pixel locations instead of floats, just change to longs.

```
circlevals = round(circle(50, 50, 20))
```

-Loren