Subject: Re: creating a 2D mask for image filtering
Posted by David Fanning on Wed, 17 Aug 2011 17:44:58 GMT
View Forum Message <> Reply to Message

Dave Higgins writes:

>
> Thanks for the advice, very much appreciated. I followed your general method as follows, but actually it's not the filter shape I want:
>

```
>  FUNCTION CIRCLE, xcenter, ycenter, radius
>     points = (2 * !PI / 99.0) * FINDGEN(100)
>     x = xcenter + radius * COS(points )
>     y = ycenter + radius * SIN(points )
>     RETURN, TRANSPOSE([[x],[y]])
>     END
>
>  PRO apod_filter
>     image = dist(512)
>     WINDOW, 0, TITLE = 'Test data before filtering'
>     SHADE_SURF, image
>     s = Size(image, /Dimensions)
>     hf = Hanning(s[0], s[1], ALPHA=0.5)
>     maxRadius = Min(s)/2;
>     WINDOW, 1, TITLE = 'temporary window', xsize=s[0], ysize=s[1]
>     ; Use of data coords for "circle" ok since data extent may not be square
>     POLYFILL, CIRCLE(s[0]/2-1, s[1]/2-1, 0.8*maxRadius), color=1
>     circleMask = TVRD()
>     WDELETE, 1
>     indices = Where(circleMask EQ 1)
>     hf[indices] = 1
>     WINDOW, 2, TITLE = 'Filter to be applied'
>     SHADE_SURF, hf
>     WINDOW, 3, TITLE = 'Apodized image'
>     SHADE_SURF, hf*image
>     WDELETE, 0, 3
>  END
```

>
> This produces a step down at the edge of the circle, to where the Hanning window was before the circleMask was applied. But this step-down causes Gibbs ringing in a FT of the data.
> I was aiming for a smooth "S" shaped (or similar) reduction of the filter values from the edge of the circle to the edge of the data.
> I was wondering if I applied some sort of smoothing to circleMask, it would blur the edge of the circle and achieve the smooth decent at the circle edge:
>

```
>  PRO apod_filter
>     image = dist(512)
>     WINDOW, 0, TITLE = 'Test data before filtering'
>     SHADE_SURF, image
>     s = Size(image, /Dimensions)
>     maxRadius = Min(s)/2;
>     WINDOW, 1, TITLE = 'temporary window', xsize=s[0], ysize=s[1]
>     ; Use of data coords for "circle" ok since data extent may not be square
>     POLYFILL, CIRCLE(s[0]/2-1, s[1]/2-1, 0.8*maxRadius), color=1
>     circleMask = float(TVRD())
>     WDELETE, 1
>     help, circlemask
>     circleMask = SMOOTH(circleMask, 100, /EDGE_TRUNCATE, MISSING=0.0)
>     indices = Where(circleMask GT 0.01)
>     filter = fltarr(s[0], s[1])
>     filter[indices] = circleMask[indices]
>     WINDOW, 2, TITLE = 'Filter to be applied'
>     SHADE_SURF, filter
>     WINDOW, 3, TITLE = 'Apodized image'
>     SHADE_SURF, filter*image
>     WDELETE, 0, 3
>  END
>
```

> ...but this smoothing eats back into my leave-it-alone circle of data; I'd like to start the descent to zero at the edge of the originally defined circleMask.
>
> Thanks for any further help.

Aren't you smoothing the wrong thing? I think you want this:

```
PRO apod_filter
   image = dist(512)
   WINDOW, 0, TITLE = 'Test data before filtering'
   SHADE_SURF, image
   s = Size(image, /Dimensions)
   hf = Hanning(s[0], s[1], ALPHA=0.5)
   maxRadius = Min(s)/2;
   WINDOW, 1, TITLE = 'temporary window', xsize=s[0], ysize=s[1]

   POLYFILL, CIRCLE(s[0]/2-1, s[1]/2-1, 0.8*maxRadius), color=1
   circleMask = TVRD()
   WDELETE, 1
   indices = Where(circleMask EQ 1)
   hf[indices] = 1
   WINDOW, 2, TITLE = 'Filter to be applied'
   SHADE_SURF, Smooth(hf, 50, /edge_truncate)
   WDELETE, 0
END
```

Also, TVCircle (from the Astronomy Library) will produce
a real circle no matter what the aspect ratio of the window.

One of the reasons it's one of my favorite programs. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")