
Subject: Re: shading/outlining on surface plot
Posted by [astroboy.20000](#) on Fri, 18 Jan 2013 19:11:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Jan 18, 12:25 pm, David Fanning <n...@idlcoyote.com> wrote:

> Bob Plano writes:

>> I probably didn't pick a very good example of what I'm trying to do,
>> in that the 'pixels' on the surface plot I want to outline are not
>> necessarily contiguous. That is to say, out of a 64 by 64 surface,
>> there might be a few dozen points that need to be highlighted or
>> outlined, at (apparently) random locations. So, contours won't work,
>> at least not in a straightforward way.

>

>> I wonder if it's possible to do something like:

>

>> shade_surf,z,x,y,shades=col, where most of the col array has values
>> that are 'invisible'. I don't suppose there is such a thing...

>

> Actually, something like this works pretty well:

>

```
> peak = cgdemodata(2)
> markedData = Round(Randomu(-5L, 50) * (41*41))
> cgLoadCT, 0, /Brewer, /Reverse, NColors=254
> markedColor = 'red'
> TVLCT, cgColor('red', /Triple), 255
> markedImage = BytScl(peak, top=254)
> markedImage[markedData] = 255
> cgSurface, peak, texture_image=markedimage
> END
```

>

> Here is the result:

>

> http://www.idlcoyote.com/misc/marked_image.png

>

> You see one small problem with a very dark green pixels at the very top
> of the data range. I know what causes this and can fix it easily, if you
> think something like this works for you.

>

> Cheers,

>

> David

> --

> David Fanning, Ph.D.

> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming:<http://www.idlcoyote.com/>

> Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Thanks Dave,

This is pretty much what I want, except....

instead of having the selected pixels one particular color, I'd just like to have them outlined.

What I'm actually doing might be sort of interesting. I have an image which has colors that correspond to data values. I then create a colored histogram so to speak: I basically take each pixel and stack it in the appropriate bin of this histogram I'm making. In the end you have a mosaic, sort of, where each 'tile' is a pixel from the image. I can click on one of the tiles, or a pixel of the image, and the corresponding tile or pixel is outlined. This is a surprisingly useful tool, especially if you're trying to find out exactly how a discontinuous set of points falls within a histogram.

What I'd like to do is have the corresponding pixels on a surface plot also outlined. If all I can do is have them colored a particular color, that's OK.

I've been trying to do something along the lines of this:

```
shade_surf,data,xx,yy,/noerase,shades=whatever
xrang=!x.crange
yrang=!y.crange
zrang=!z.crange
```

```
for k=0,n_elements(selected_points)-1 do begin
```

```
;if k is the index of the point, kk is the index of the next point to
the 'upper right'
kk=some_mapping_procedure(k)
```

```
surface, [ [data(k),data(kk)],[data(k),data(kk)] ], [ [x(k),x(kk)],
[x(k),x(kk)] ], [ [y(k),y(kk)],[y(k),y(kk)] ],xrange=xrang, $
           yrange=yrang, zrange=zrang,shades=some_color
```

```
endfor
```

It's not working well, but I think that's because the mapping_procedure has a bug. It seems to me that if data, x, and y are 2x2 arrays, with the first and third data elements having the value of "lower left" and the second and fourth elements having the values of "upper right", then surface will draw the appropriate box on the pre-existing surface. I'm not sure if that's how surface works or not. I'll have to play with it for a bit.

I suppose I could do something like this with a plots command, but I was thinking surface might save me the trouble of having to make arrays with all the corner points specified. Thanks, Chris. I read your paper on wavelets about once a week because I keep forgetting what little I know about them.
