
Subject: Re: trajectory plot behind sphere
Posted by [Ralf Schaa](#) on Thu, 17 Feb 2005 16:28:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks Kenneth,

I used the Z-Buffer...what can I say, it's magic...

this is what i did and it works:

```
; set up 3d
Xcnt = N_Elements(X1)
Ycnt = N_Elements(X2)
Zcnt = N_Elements(X3)
Surface, FItArr(Xcnt,Xcnt), X1, X2, $, background=1,color=3, $
  Xrange=[-3,3], Yrange=[-3,3], ZRange=[-3,3], $
  XStyle=4, YStyle=4, ZStyle=4, TICKLEN=1, /SAVE, /NODATA

; the sphere
sphere = fltarr(40,40,40)
for x=0,39 do for y=0,39 do for z=0,39 do sphere[x,y,z]=sqrt((x-10)^2 +
(y-10)^2 + (z-10)^2)
shade_volume, sphere, 10, v, p, low=1
v=0.1d*(v-10)
image = polyshade(v, p, /t3d)
;tv,image

; draw trajectory
Plots, X1, X2, X3, color=8, thick =2, /T3D

; axis
AXIS, 0,0,0, /XAxis, /T3D, CharSize=2.5, color=5, xtickformat='(i8)'
AXIS, 0,0,0, /YAxis, /T3D, CharSize=2.5, color=6, ytickformat='(i8)'
AXIS, 0,0,0, /ZAxis, /T3D, CharSize=2.5, color=7, ztickformat='(i8)'
plots, X1[0],X2[0],X3[0], psym = 1, /T3D, color=1, symsize=3

; get z-buffer
img = tvrd()

; plot png
thisPlot = tvread(/png,/nodialog,order=1,filename='magic')
```

I do have also a routine with object graphics, which actually does the same. But I only changed the example program 'xplot3d__draw' to my needs and it worked somehow .. I think it is very time consuming doing little

things like an equivalent of 'xyouts' in object graphics , so I keep going back to direct graphics.

Cheers
-Ralf

Kenneth Bowman wrote:

```
> In article <37jpcoF4mbq4mU1@individual.net>,
> Ralf Schaa <schaa@geo.uni-koeln.de> wrote:
>
>
>> I am plotting a trajectory of a spacecraft orbit which is partially
>> going behind a sphere (the planet). This part, which is actually behind
>> the sphere is drawn right in front of the plot, but I like to have this
>> part of the trajectory invisible.
>> How can this be done?
>
>
> The problem is that the X and WIN devices are 2-D devices and do not do
> hidden line processing. SURFACE and similar built-in "3-D" functions do
> their own hidden line removal, so they work with 2-D devices, but I
> don't think that most Direct Graphic procedures do. (I'm certain I'll
> be corrected if I'm wrong!)
>
> I believe that you must either use the Z-buffer device or Object
> Graphics, both of which will remove hidden lines.
>
> With the Z-buffer device, you set the device to "Z", draw your 3-D
> graphic, read the resulting bitmap with TVRD, then TV it to the screen
> (or print it). The downside is that the result is a bitmap. On the
> other hand, the downside of Object Graphics is the learning curve. ;-)
```

> Ken Bowman