
Subject: Re: IDLgrSurface with transparent lines?
Posted by [Rick Towler](#) on Thu, 29 Apr 2004 17:22:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Antonio Santiago" wrote in message...

> But really i want something like the next code, but instead with a
> texture map image i want use the vert_colors options. In the next code i
> draw a texture map image with lines in the IDLgrSurface. If you change
> the ";" to use vert_colors and ommit texture_map then the IDLgrSurface
> is drawn without transparence.
> i would like can specifie a level of transparence in the colors of
> IDLgrSurface (if possible).

You can do this by creating a texture containing your palette and then selecting your texture coordinates appropriately. It is a bit more work, but you'll have control of the color at every vertex.

-Rick

```
oPalette = obj_new('IDLgrPalette')  
oPalette -> LoadCT, 13
```

```
; Create a 256x256 texture map modulating color and opacity  
imagedata = BYTARR(4, 256, 256, /NOZERO)  
for alpha = 0, 255 do begin  
  for color=0,255 do begin  
    imagedata[0:2,alpha,color] = oPalette -> GetRGB(color)  
    imagedata[3,alpha,color] = alpha  
  endfor  
endfor  
oTexture = OBJ_NEW('IDLgrImage', imagedata, BLEND_FUNCTION=[3,4])
```

```
; Create the surface object  
z = DIST(45)  
oSurface = OBJ_NEW('IDLgrSurface', DIST(45), COLOR=[255,255,255], $  
  STYLE=1, THICK=2, TEXTURE_MAP=oTexture)  
oModel = OBJ_NEW('IDLgrModel')  
oModel -> Add, oSurface
```

```
; Create the texture coordinates  
; Get the surface vertices  
oSurface -> GetProperty, DATA=surfVerts  
nVerts = SIZE(surfVerts, /DIMENSIONS)  
nVerts = nVerts[1] * nVerts[2]
```

```

; Reform so it is a bit easier to work with
surfVerts = REFORM(surfVerts, 3, nVerts)

; Find the max
maxZ = MAX(surfVerts[2,*])

; Create the texture coordinate array x&y for each vertex
texCoords = FLTARR(2,nVerts, /NOZERO)

; set Color (1) and Alpha (0) according to height
; Fudge the texcoord max otherwise you'll have a hole
; in the top.
texCoords[0,*] = 0.0 > (surfVerts[2,*] / maxZ) < 0.99999
texCoords[1,*] = 0.0 > (surfVerts[2,*] / maxZ) < 0.99999

oSurface -> SetProperty, TEXTURE_COORD=texCoords


; Compare to VERT_COLORS

; Create the vert colors array
vertColors = BYTARR(3, nVerts, /NOZERO)
for v=0, nVerts-1 do $
    vertColors[* ,v] = oPalette -> GetRGB(texCoords[0,v] * 255)

; Create the vert colors surface
oSurfaceVC = OBJ_NEW('IDLgrSurface', DIST(45), COLOR=[255,255,255], $
    STYLE=1, THICK=2, VERT_COLORS=vertColors)
oModelVC = OBJ_NEW('IDLgrModel')
oModelVC -> Add, oSurfaceVC

XOBJVIEW, oModelVC
XOBJVIEW, oModel, /BLOCK

OBJ_DESTROY, [oModel, oModelVC, oTexture, oPalette]

end

```
