Subject: Re: texture_coord
Posted by Karl Schultz on Thu, 01 Nov 2001 16:11:32 GMT
View Forum Message <> Reply to Message

"Harald von der Osten-Woldenburg" <hvdosten@lb.netic.de> wrote in message news:3BE13E28.B452973E@lb.netic.de...
> Hi,
>
> maybe it's easier than I feel, but I don't came along with it:

It isn't that bad, but it is sort of hard to explain. Code helps - see the bottom of this post.

> I would like to map a jpeg-file onto a small part of a 3D-surface. It
> works fine if I consider the entire surface. But this is not what I want
> to have. The problem seems to be the array texture_coord.

Yes, using TEXTURE_COORD is how you do this.

> If the surface is of an arry of - lets say - 1000 x 1200, how could I
> map a jpeg-file onto this surface with the subsurface-grid-coordinates
> [100, 50], [200,50], [200,300], [100,300]? And: I hope that the
> jpeg-file can have a higher resolution than [100 x 250] pixels for this
> example?

The image (jpeg file) can be any resolution you want. That is part of the utility of texture mapping.

> Whatever I try - I get the error-message "number of vertices, normals,
> and texture coordinates do not match". Concerning to the online-help
> "TEXTURE_COORD property defines how individual data points within the
> image data are mapped...". I don't hope that each pixel in the jpeg-file
> must be referenced by corresponding coordinates....

Yes, you do need to specify a texture coordinate for each vertex in your polygon or surface. Otherwise, IDL won't know how to map your image to the geometry. I know that it "seems" obvious in a lot of cases, especially when dealing with a specialzation of a polygon such as the surface. It seems like you'd just want to linearly interpolate the image across the surface, and that is what we do for the default case. But you need to be fully explict with anything other than that.

Here is a real simple program you can try that maps an IDL logo onto a subrect in the middle of a surface. Loop-haters can probably figure out how to get rid of the loop, but I hope that what I have below is clear enough to understand easily.

Hope that this helps,

Karl

```
pro texture

filename = FILEPATH('examples.tif', SUBDIRECTORY=['examples','data'])
imageData = READ_TIFF(filename, R, G, B)
imageData = REVERSE(imageData,2)
oPalette = OBJ_NEW('IDLgrPalette', R, G, B)
oImage = OBJ_NEW('IDLgrImage', imageData, PALETTE=oPalette)
texCoords = FLTARR(2,100,120)
; subrect is [20,30] to [80,90]
FOR y=30, 90 DO BEGIN
   texCoords[0, 20:80, y] = FINDGEN(61)/ 60
   texCoords[1, 20:80, y] = (y-30) / 60.0
ENDFOR
oSurface = OBJ_NEW('IDLgrSurface', dist(100,120), $
   COLOR=[255,255,255], STYLE=2, TEXTURE_MAP=oImage,
TEXTURE_COORD=texCoords)

xobjview, oSurface
end
```