Subject: Re: Depth visibility with Object Graphics !!! Posted by Antonio Santiago on Wed, 26 May 2004 06:48:28 GMT View Forum Message <> Reply to Message

Sorry, but I can't obtain any good result with your ideas. I have been playin with DEPTH\_TEST\_xxx atributes but nothing, perhaps i am a squar-head:)

I attach a little file with my problem. First i add to the model the "line" an then the "image". I change the values of DEPTH TEST FUNCTION, DEPTH\_TEST\_DISABLE and DEPTH\_TEST\_WRTIE but i can't do the line overdrawn the image.

Thanks.

```
Rick Towler wrote:
> "Antonio Santiago" wrote...
>> Well, thanks for your help, but that's not really i mean (perhaps it is
>> due to my bad english, sorry).
>
>
> You haven't convinced me that what Ben is talking about isn't what you mean.
> You can do it without texture mapping your image but everything is going to
> be in image coordinates. Make sure your line coordinates are appropriate.
>
>
>
>> I have a model inside which i put an IDLgrImage as a map of a portion of
>> Spain. Also i put an IDLgrPolyline to make selections (as a line) over
>> the map.
>
>
> Where are you setting the DEPTH TEST DISABLE or DEPTH TEST FUNCTION
> keywords? If both of these atoms are in a single model setting these
> keywords for the model will have no effect. You will need to set the
> DEPTH TEST FUNCTION keyword of your polyline to "always pass" (I think it is
> 8 but don't have IDL in front of me).
>
>> My problem is that the line is never seen because i put the map
   int the model after put the line, it is the map overlaps the line.
>
```

```
> Assuming you aren't using alpha blending the order in which you add your
> atoms to the model shouldn't matter as long as they have different Z values.
> Remember, images that aren't texture mapped are rendered at Z=0. Your line
> must have +Z values.
If you hide the image, can you see the line?
>
>
> -Rick
>
>
>> Ben Tupper wrote:
>>> Antonio Santiago wrote:
>>>
>>>
>>>> Hi,
>>>>
>>>> i am working with Object Grpahics and i hace a problem :)
>>>>
>>>> I have put on a model two objects. Firts, i must put an IDLgrPolyline
>>>> betwen two points (a line) that i can modify to select one trajectory.
>>> Second i put an IDLgrImage.
>>>> My problem is that image is drawing over the polyline. I play with
>>>> DEPTH_TEST_DISABLE and DEPT_TEST_FUNCTION, but always the image is
>>>> drawing over the polyline.
>>>>
>>>
>>> Hi,
>>>
>>> IDL renders images at Z=0 always. To get around this you have to map
>>> the image as a texture map onto a polygon; then set the Z values for the
>>> polygon to something 'further' away than your trajectory. There is an
>>> example here...
>>>
>>> http://tinyurl.com/3h5aw
>>>
>>> Ben
>>
>
>
; This is a utility routine to calculate the scaling vector
; required to position a vector of specified range at a
; specific position given in normalized coordinates. The
```

```
; scaling vector is given as a two-element array like this:
  scalingVector = [translationFactor, scalingFactor]
 The scaling vector should be used with the [XYZ]COORD_CONV
 keywords of a graphics object or model. For example, if you
 wanted to scale an X axis into the data range of -0.5 to 0.5.
 you might type something like this:
 xAxis->GetProperty, Range=xRange
  xScale = Normalize(xRange, Position=[-0.5, 0.5])
  xAxis. XCoord Conv=xScale
FUNCTION Normalize, range, Position=position
On Error, 1
IF N_Params() EQ 0 THEN Message, 'Please pass range vector as argument.'
IF (N Elements(position) EQ 0) THEN position = [0.0, 1.0] ELSE $
 position=Float(position)
range = Float(range)
scale = [((position[0]*range[1])-(position[1]*range[0])) / $
 (range[1]-range[0]), (position[1]-position[0])/(range[1]-range[0])]
RETURN, scale
END
;---- MAIN PROGRAM -----
PRO line
:Create widgets and objects
wBaseTop = WIDGET BASE(TITLE='Line')
wDraw = WIDGET_DRAW(wBaseTop, XSIZE=300, YSIZE=300, GRAPHICS_LEVEL=2)
WIDGET CONTROL, wBaseTop, /REALIZE
WIDGET_CONTROL, wDraw, GET_VALUE=oWindow
oView = OBJ_NEW('IDLgrView', VIEWPLANE_RECT=[-1,-1,2,2])
oModel = OBJ_NEW('IDLgrModel')
oView->Add, oModel
;First i put a line.
coord = [[-1,-1, 1], [1,1,1]]
polyline = [2, 0,1]
oLine = OBJ_NEW('IDLgrPolyline', coord, POLYLINES=polyline, $
 DEPTH TEST FUNCTION=8)
oModel->Add, oLine
```

## ;Second i put an image

;Read image

READ\_JPEG, FILEPATH('rose.jpg', SUBDIR=['examples','data']), img, /TRUE

;Normalize dimensions (3 x X x Y)

sizes = SIZE(img, /DIMENSIONS)

xnorm = NORMALIZE([0, sizes[1]-1], Position=[-0.9,0.9])

ynorm = NORMALIZE([0, sizes[2]-1], Position=[-0.9,0.9])

olmage = OBJ\_NEW('IDLgrImage', img, XCOORD\_CONV=xnorm, YCOORD\_CONV=ynorm) oModel->Add, oImage

oWindow->Draw, oView

**END** 

## File Attachments

1) line.pro, downloaded 83 times