Subject: Depth visibility with Object Graphics !!!
Posted by Antonio Santiago on Mon, 24 May 2004 18:07:42 GMT
View Forum Message <> Reply to Message

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.

Any idea?

Thanks.

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 seleccions (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])

 oImage = OBJ_NEW('IDLgrImage', img, XCOORD_CONV=xnorm, YCOORD_CONV=ynorm)
 oModel->Add, oImage


 oWindow->Draw, oView

END
```

File Attachments

1) line.pro, downloaded 78 times

Subject: Re: Depth visibility with Object Graphics !!!
Posted by btt on Wed, 26 May 2004 13:17:47 GMT
View Forum Message <> Reply to Message

Antonio Santiago wrote:

> 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.
>

I never have gotton a good feel for the DEPTH_**** keywords. Below is a simple approach that puts the places the image on a polygon as a texture map.

Rick pointed out that it is sometimes advantageous to transform the whole shebang out of pixel space into real data space. This example does just that... in fact, if the pixel size was important for your work you could get away from the normalization and work in "raw" data coordinates. For 2d objects graphics (usually images with ROIs and axes) I usually don't normalize.

```
oImage = OBJ_NEW('IDLgrImage', img)

opolygon = OBJ_NEW('IDLgrPolygon', $
 [0, 0, sizes[1]-1, sizes[1]-1], $
 [0, sizes[2]-1, sizes[2]-1, 0], $
 Color = [255,255,255], $
 XCOORD_CONV=xnorm, YCOORD_CONV=ynorm, $
 Texture_Map = oImage, $
 TEXTURE_COORD = [[0,0], [1,0], [1,1], [0,1]])

oModel->Add, opolygon


oWindow->Draw, oView
```

Subject: Re: Depth visibility with Object Graphics !!!
Posted by David Fanning on Wed, 26 May 2004 14:05:21 GMT

View Forum Message <> Reply to Message

Folks,

Fooling around with this code that Antonio and Ben
supplied a little bit this morning, I've convinced
myself this is a bug in how images are displayed.
(I've always thought images were a little strange,

anyway.) It's not just that images are always
displayed at z=0. When they are displayed there,
it is as though they were displayed by pushing
them down through Z space so that they obliterate
anything already there. (Reminds me of direct graphics
images, really.) Clearly this doesn't happen if the
image is in a polygon.

I'd really be curious to hear what Karl has to
say about this. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Depth visibility with Object Graphics !!!
Posted by Karl Schultz on Wed, 26 May 2004 14:30:31 GMT

View Forum Message <> Reply to Message

"Antonio Santiago" <d6522117@est.fib.upc.es> wrote in message
news:40B239EE.10308@est.fib.upc.es...
> 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.
>
> Any idea?

The IDLgrImage does not interact with the depth buffer as it is drawn. So,
if you draw the image after the polylines, the image will overwrite the
polylines where they overlap, independent of the depth of the polylines. If
you draw the image first, you will always see the lines. In general, it is
usually better to arrange scenes containing images so that the images get
drawn first.

The reason behind this is that IDLgrImage objects are drawn with an OpenGL pixel primitive (glDrawPixels), which is a special 2D image drawing primitive. The intent of IDLgrImage is to provide a "billboarding" style image drawing primitive, so the usage of this primitive makes sense.

The discussion in this thread about using a texture-mapped polygon gives you the flexibility of drawing the image in space whereever you want with the full interaction with the depth buffer if that is what you would expect.
So, you have both methods available to you in IDL.

The IDL (6.1) docs say:

Image objects do not take into account the Z locations of other objects that may be included in the view object. This means that objects that are drawn to the destination object (window or printer) after the image is drawn will appear to be in front of the image, even if they are located at a negative Z value (behind the image object). Objects are drawn to a destination device in the order that they are added (via the Add method) to the model, view, or scene that contains them. To rotate or position image objects in three-dimensional space, use the IDLgrPolygon object with texture mapping enabled.

Karl