
Subject: Re: Transparent texture mapped polygons
Posted by [Rick Towler](#) on Mon, 25 Mar 2002 04:59:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yes. In the simpler cases this is as easy as adding a 4th channel to your texture map called an alpha channel. There are examples of this in the "What's new in 5.5" docs and at David Fanning's website. If your polygon is a billboard (plane) this all you'll probably need.

-Rick

"lyubo" <lzagorch@cs.wright.edu> wrote in message
news:a7ljs1\$fr\$1@mercury.wright.edu...

> Is it possible to make a texture mapped polygon object partially
transparent
> so that it could be displayed over another texture mapped polygon?
> If anyone knows how to do that please let me know.
>
> Thank you.
>
> Lyubo
>
>
>

Subject: Re: Transparent texture mapped polygons
Posted by [lyubo](#) on Mon, 25 Mar 2002 18:15:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Rick,

First, thanks again for your response.

I have studied David's example about transparent images and also the example in "What's new in IDL 5.5", but I still couldn't get it to work with texture mapped polygons. It works fine if I try to blend two images, but it doesn't work with polygons.

I've been trying to overlay two images and draw them in 3D. To do that I create two texture mapped planes (with the images as texture) but I have problems making one of the planes transparent. I create an "alpha" image, as in David's example, for the foreground plane hoping that it will be blended with the background image (the other plane) but it doesn't work because when I add the background plane to the

model and then
the foreground plane (the one with the alpha channel), the foreground plane
will
not show up on the screen. It is there, but is covered by the background
plane.

I was trying different things and I noticed that if I switch the order of
adding the planes to the model (first the "alpha" plane and then the
background plane),
both planes will be displayed, but the foreground plane isn't transparent,
rather it is
drawn over the background plane.

Below is the code that I used to test this, a part of it comes from David's
example.

If you see what's wrong I would really appreciate your help.

Thanks,

Lyubo

```
.*****  
,  
*****  
  
oImage=OBJ_NEW('IDLgrImage',(*volume.image)[*,*,depth/2])  
oPlane = OBJ_NEW('IDLgrPolygon', verts,  
    COLOR=[255,255,255],TEXTURE_COORD=[[0,0],[1,0],[1,1],[0,1]], $  
        TEXTURE_MAP=oImage, XCOORD_CONV=xs,  
YCOORD_CONV=ys, ZCOORD_CONV=zs)  
  
foregroundImage=(*volume.image)[*,*,20]  
s = Size(foregroundImage, /Dimensions)  
alpha_image = BytArr(4, s[0], s[1])  
  
(*pTlbState).oPalette[2]->LoadCT, 3  
(*pTlbState).oPalette[2]->GetProperty, red_values=r, green_values=g,  
blue_values=b  
alpha_image[0,*,*]=r[foregroundImage]  
alpha_image[1,*,*]=g[foregroundImage]  
alpha_image[2,*,*]=b[foregroundImage]  
    ; Pixels with value 0 with be totally transparent.  
    ; Other pixels will start out half transparent.  
blendMask = BytArr(s[0], s[1])  
blendMask[Where(foregroundImage GT 10)] = 1B  
alpha_image[3, *, *] = blendMask * 128B  
  
alphaImage = OBJ_NEW('IDLgrImage',alpha_image, INTERLEAVE=0,  
BLEND_FUNCTION=[3,4], PALETTE=(*pTlbState).oPalette[2])
```

```
oBlendPlane = OBJ_NEW('IDLgrPolygon', verts, COLOR=[255,255,255],  
TEXTURE_COORD=[[0,0],[1,0],[1,1],[0,1]], $  
  TEXTURE_MAP=alphaImage, TEXTURE_INTERP=1, XCOORD_CONV=xs,  
YCOORD_CONV=ys,  
ZCOORD_CONV=zs)
```

```
(*pDisplayState).oModel[1,2]->Add, oPlane  
(*pDisplayState).oModel[1,2]->Add, oBlendPlane
```

Subject: Re: Transparent texture mapped polygons
Posted by [Rick Towler](#) on Mon, 25 Mar 2002 20:24:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

"lyubo" <lzagorch@cs.wright.edu> wrote

> I've been trying to overlay two images and draw them in 3D. To do that I
> create two texture mapped planes (with the images as texture) but I have
> problems
> making one of the planes transparent. I create an "alpha" image, as in
> David's example,
> for the foreground plane hoping that it will be blended with the
background
> image (the other
> plane) but it doesn't work because when I add the background plane to the
> model and then
> the foreground plane (the one with the alpha channel), the foreground
plane
> will
> not show up on the screen. It is there, but is covered by the background
> plane.

Yes, objects in a view are rendered according to their order in the view container. When texturing with an alpha channel the order of rendering becomes important. Add items from back to front like you first tried. I see that you are sharing the same vertices between your billboards. I am not sure how IDL will handle rendering this since your planes occupy the same z space but I think this is your problem. Push your background plane back a little in z.

You can also remove the palette keyword for "alphaImage". It should be ignored since you have passed (and will always pass) a 4 channel image but maybe it is having an effect.

-Rick

Subject: Re: Transparent texture mapped polygons
Posted by [David Fanning](#) on Tue, 26 Mar 2002 01:16:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Rick Towler (rtowler@u.washington.edu) writes:

> Yes, objects in a view are rendered according to their order in the view
> container. When texturing with an alpha channel the order of rendering
> becomes important. Add items from back to front like you first tried. I
> see that you are sharing the same verticies between your billboards. I am
> not sure how IDL will handle rendering this since your planes occupy the
> same z space but I think this is your problem. Push your background plane
> back a little in z.

I think having coincident polygons is the problem, too.
I've modified my Image_Blend program so that the images
rotate on polygons. You can have a look if you are
interested.

ftp://ftp.dfanning.com/pub/dfanning/outgoing/misc/image_blend_poly.pro

Cheers,

David

--

David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Transparent texture mapped polygons
Posted by [Karl Schultz](#) on Tue, 26 Mar 2002 16:09:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.170979cae136002198984b@news.frii.com...

> Rick Towler (rtowler@u.washington.edu) writes:

>

>> Yes, objects in a view are rendered according to their order in the view
>> container. When texturing with an alpha channel the order of rendering
>> becomes important. Add items from back to front like you first tried.

I

>> see that you are sharing the same verticies between your billboards. I
am

>> not sure how IDL will handle rendering this since your planes occupy the

```
>> same z space but I think this is your problem. Push your background
plane
>> back a little in z.
>
> I think having coincident polygons is the problem, too.
> I've modified my Image_Blend program so that the images
> rotate on polygons. You can have a look if you are
> interested.
>
> ftp://ftp.dfanning.com/pub/dfanning/outgoing/misc/image_blend_poly.pro
>
```

Yes, I agree about the coincident polygons.

Another approach that may be interesting to try is to use the DEPTH_OFFSET keyword that was added in 5.5. This feature is intended to help solve Z-buffer "stitching" and interference problems in situations like these. It has the advantage of applying the depth offset in view coordinates, so you don't have to tweak the objects' locations as you spin them around with a trackball in order to keep the "back" polygon in the back. Just add DEPTH_OFFSET=1 to your "back" polygon.

Karl
