Subject: Re: 3-d viewing
Posted by davidf on Tue, 21 Mar 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Steven Chetelat (CS) (chetelat@csee.usf.edu) writes:

> Hello all. I've got another problem to ask for advice with. I could
> probably muddle through entirely on my own, but I've learned I know just
> the wrong amount of idl to try to do that...I'd end up with unmaintainable
> code for sure. So here goes. My question has to do with viewing a
> surface. It's an isosurface of a binary 3d image. Right now I generate
> it with:
>
> shade_volume, new, .5,vert,poly, /low
>
> and display it with :
>
> scale3, xrange=[0,xr], yrange=[0,yr], zrange=[0,zr],ax=xa,az=za
> tv,bytscl(polyshade(vert,poly,/t3d))
>
> The angles and everything work fine, but the problem is that my x-axis
> is about twice as long as the other 2, and I'm displaying it in an
> 800x400 draw widget. The problem I run into is that as I change the
> angles, so that one of the other axes is oriented along the long edge of
> the display window, the view is scaled into the window and the result
> looks frighteningly comical. Is there a *really* simple way to fix this?
> If not, which approach should I use. There's a couple of hints I've
> gotten looking through the manuals and David's book, but I figured I'd see
> if anyone could tell me where to dig before I start digging...

I've just got a minute, so I can't write much
of a helpful response. But if I really wanted a
place to start digging, I'd be digging in the Object
Graphics manuals. :-)

I'd start with a program like Simple_Surface to help you set up
the Viewport coordinate system, add a trackball, etc. You will
want to substitute a Polygon object for the Surface object in
that program, but most of the infrastructure is already
built for you, and you can certainly see from the surface
object how to scale your polygon object into the view.

ftp://www.dfanning.com/pub/dfanning/outgoing/idl_course/simp le_surface.pro

The reason you are having trouble is the very same reason
that drove RSI to develop the object graphics system in the
first place. :-)

Cheers,

David

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

Subject: Re: 3-d viewing
Posted by davidf on Wed, 22 Mar 2000 08:00:00 GMT
View Forum Message <> Reply to Message

I wrote yesterday:

> I'd start with a program like Simple_Surface to help you set up
> the Viewport coordinate system, add a trackball, etc. You will
> want to substitute a Polygon object for the Surface object in
> that program, but most of the infrastructure is already
> built for you, and you can certainly see from the surface
> object how to scale your polygon object into the view.

Since this took just a couple of minutes to do, I decided to
make it available to everyone. The program is named
Poly_Surface. You call it with the vertex and polygon
lists that you can obtain from a program like Shade_Volume:

```
IDL> Poly_Surface, vertices, polygons
```

If you call the program with no parameters I'll build you
a little spherical polygon object. :-)

```
IDL> Poly_Surface
```

You can find the program here:

ftp://ftp.dfanning.com/pub/dfanning/outgoing/idl_course/poly _surface.pro

Be aware that this program is NOT extensively tested.
I only spent about 5 minutes on it. :-)

Cheers,

David

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

Subject: 3-d viewing again
Posted by Steven Chetelat (CS) on Mon, 27 Mar 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Last week, I wrote:

> Hello all. I've got another problem to ask for advice with. I could
> probably muddle through entirely on my own, but I've learned I know just
> the wrong amount of idl to try to do that...I'd end up with unmaintainable
> code for sure. So here goes. My question has to do with viewing a
> surface. It's an isosurface of a binary 3d image. Right now I generate
> it with:
>
> shade_volume, new, .5,vert,poly, /low
>
> and display it with :
>
> scale3, xrange=[0,xr], yrange=[0,yr], zrange=[0,zr],ax=xa,az=za
> tv,bytscl(polyshade(vert,poly,/t3d))
>
> The angles and everything work fine, but the problem is that my x-axis
> is about twice as long as the other 2, and I'm displaying it in an
> 800x400 draw widget. The problem I run into is that as I change the
> angles, so that one of the other axes is oriented along the long edge of
> the display window, the view is scaled into the window and the result
> looks frighteningly comical. Is there a *really* simple way to fix this?
> If not, which approach should I use. There's a couple of hints I've
> gotten looking through the manuals and David's book, but I figured I'd see
> if anyone could tell me where to dig before I start digging...

To which David graciously replied that I needed to dig into object graphics (Thank you, David, but that's not what I wanted to hear :-P ;-). So far I've gotten it to display, but not very well. It's just a couple of blobs on my Window object. I can't seem to recreate the lighting conditions and orientation that I had originally. Can anybody point me to something which would allow me to get started? I remember when I originally did it in direct graphics last year I had some of the same type of problems, but I overcame them by playing with it. Unfortunately, I don't know where to start playing with this...

K-Bye,
STEVE!