Subject: Re: Question About Scatter Surface Pro (www.dfanning.com)
Posted by Antonio Santiago on Wed, 17 May 2006 15:34:50 GMT
View Forum Message <> Reply to Message

I think you can use the XRANGE, YRANGE, ZRANGE keywords or YCOORD_CONV
and so on to define how to show data.

Bye.

(I have my IDL very oxy-dated :( ).

j.vanknippenberg escribiï¿½:
> Hi everyone :)
>
> I want to use this program to view roughly 100 regularly spaced points,
> and their properties. The problem I seem to have is that the ORBS are
> not spherical, but instead kind of ellipsoid. This is because the data
> ranges for X, Y and Z are not comparable? What I mean is that the
> xrange goes from 0-100, while the zrange from 0-0.01 for example. Any
> idea on how to fix this? (Im relatively new to IDL ;))
>
> The program can be found here:
> http://www.dfanning.com/tip_examples/scatter_surface.pro
>
> And I think the orbs are created in this section:
>
> ; Create the symbols for each point. This is almost certainly
> ; not the most efficient way if you have lots of points, but
> ; it works well for a reasonable number.
>
> npts = N_Elements(x)
> orbs=ObjArr(npts)
> line=ObjArr(npts)
> FOR j=0,npts-1 DO BEGIN
> orbs[j] = Obj_New('ORB', Color=[r[zcolors[j]], g[zcolors[j]],
> b[zcolors[j]]], $
> Style=2, Radius=0.015, Pos=[x[j],y[j],z[j]])
> line[j] = Obj_New('IDLgrPolyLine', [x[j], x[j]], [y[j], y[j]],
> [min(z), z[j]], $
> Color=[r[zcolors[j]], g[zcolors[j]], b[zcolors[j]]])
> thisModel->Add, orbs[j]
> thisModel->Add, line[j]
> ENDFOR
>
> Also, my data consists of 3 different sets. I would like to have 1
> color for each set, but ofcourse different when compared to the other
> 2. Any hints on how to proceed on this?

>
> Thanks in advance! :)
>
> J.vanknippenberg
>

Subject: Re: Question About Scatter Surface Pro (www.dfanning.com)
Posted by David Fanning on Wed, 17 May 2006 16:30:34 GMT
View Forum Message <> Reply to Message

j.vanknippenberg writes:

> I want to use this program to view roughly 100 regularly spaced points,
> and their properties. The problem I seem to have is that the ORBS are
> not spherical, but instead kind of ellipsoid. This is because the data
> ranges for X, Y and Z are not comparable? What I mean is that the
> xrange goes from 0-100, while the zrange from 0-0.01 for example. Any
> idea on how to fix this? (Im relatively new to IDL ;))

Alas, the problem is in the RSI (Whoops! Excuse me, ITTVIS)
supplied orb object. It uses a single number for the radius
of the orb, and you want to construct the orb using three
different radii in the proper relation to one another.

I haven't looked at the ORB code in a long while, so I
don't know how difficult this would be to do, but remembering
the equation for an ellipsoid, I wouldn't think too difficult.
You are not the only person to wish for this, however, so
if you figure it out and publish it here (with decent examples
of how to get a sphere under difficult conditions), there is
a good chance you can become famous. :-)

Cheers,

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

Subject: Re: Question About Scatter Surface Pro (www.dfanning.com)
Posted by David Fanning on Wed, 17 May 2006 19:09:08 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Alas, the problem is in the RSI (Whoops! Excuse me, ITTVIS)
> supplied orb object. It uses a single number for the radius
> of the orb, and you want to construct the orb using three
> different radii in the proper relation to one another.
>
> I haven't looked at the ORB code in a long while, so I
> don't know how difficult this would be to do, but remembering
> the equation for an ellipsoid, I wouldn't think too difficult.
> You are not the only person to wish for this, however, so
> if you figure it out and publish it here (with decent examples
> of how to get a sphere under difficult conditions), there is
> a good chance you can become famous. :-)

I've heard from someone who prefers NOT to be famous.
He writes:

> The ORB object is a subclass of IDLgrModel, so all you
> really have to do is call -> Scale on the object reference
> (and/or Translate to get it in the range 0-Size rather
> than -radius to +radius).

Seems a lot easier than my idea. :-)

Cheers,

David

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

Subject: Re: Question About Scatter Surface Pro (www.dfanning.com)
Posted by David Fanning on Wed, 17 May 2006 21:03:07 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> I've heard from someone who prefers NOT to be famous.
> He writes:
>
> The ORB object is a subclass of IDLgrModel, so all you
> really have to do is call -> Scale on the object reference
> (and/or Translate to get it in the range 0-Size rather
> than -radius to +radius).
>

> Seems a lot easier than my idea. :-)

Humm. It seems easier, but can't get it to work. :-(

Does the person who made this suggestion have an example? :-)

Cheers,

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

Subject: Re: Question About Scatter Surface Pro (www.dfanning.com)
Posted by j.vanknippenberg on Thu, 18 May 2006 08:45:24 GMT
View Forum Message <> Reply to Message

Thanks for the input so far. :)

I'll have a look at it, but if the person who prefers not to be famous
:p can provide us with an example, it would be greatly appreciated ;)

Subject: Re: Question About Scatter Surface Pro (www.dfanning.com)
Posted by David Fanning on Thu, 18 May 2006 16:18:25 GMT
View Forum Message <> Reply to Message

j.vanknippenberg writes:

> Thanks for the input so far. :)
>
> I'll have a look at it, but if the person who prefers not to be famous
> :p can provide us with an example, it would be greatly appreciated ;)

OK, after much to-ing and fro-ing, I think we have
a solution in search of an explanation.

Because I grew up in a direct graphics world (in my day
the most exciting computer graphics was done with monospaced
fonts on huge line printers) I like to use a viewplane
coordinate system that goes from 0 to 1, or something
close to it. And I position my axes and data and whatnot
into that coordinate system with my NORMALIZE function,
which produces a two-element array for scaling and translating
data into this coordinate system.

In the SCATTER_SURFACE code we are talking about, I do this
to position the three axes:

```
  xs = Normalize(xrange, Position=[-0.5,0.5])
  ys = Normalize(yrange, Position=[-0.5,0.5])
  zs = Normalize(zrange, Position=[-0.5,0.5])

   ; Scale the axes and place them in the coordinate space.
   ; Note that not all values in the Location keyword are
   ; used. (I've put really large values into the positions
   ; that are not being used to demonstate this.) For
   ; example, with the X axis only the Y and Z locations are used.

  xAxis->SetProperty, Location=[9999.0, -0.5, -0.5], XCoord_Conv=xs
  yAxis->SetProperty, Location=[-0.5, 9999.0, -0.5], YCoord_Conv=ys
  zAxis->SetProperty, Location=[-0.5,  0.5, 9999.0], ZCoord_Conv=zs
```

Note the use of the [XYZ]COORD_CONV keywords. These are meant to
scale and translate (if I read the documentation correctly) the
object from it's native data coordinate system into MY coordinate
system.

So far, so good. The axes end up where they are suppose to be, etc.
But in the very next line, I also use the [XYZ]COORD_CONV keywords
to position the orbs (heads of the pins) and lines (shank of the pins)
in my data coordinate system:

```
  FOR j=0,npts-1 DO BEGIN
    (line[j]) -> SetProperty, XCoord_Conv=xs, $
              YCoord_Conv=ys, ZCoord_Conv=zs
    (orbs[j]) -> SetProperty, XCoord_Conv=xs, $
              YCoord_Conv=ys, ZCoord_Conv=zs
 ENDFOR
```

This appears to be the problem. When the range of the X, Y, and Z
axes are similar, the orb appears as I expect it to appear, as a
nice ball-like structure. But if, say, the X range is 100 times
larger than the Y and Z range, the orb turns into a flat disk.

I have learned in discussions this morning that I can avoid
this problem if I *don't* use the [XYZ]COORD_CONV keywords with
the orbs, but simply position them with the DATA keyword *while*
scaling and translating them. Huh!?

Here is the corrected code:

```
  FOR j=0,npts-1 DO BEGIN
```

```
    (line[j])-> SetProperty, XCoord_Conv=xs, $
            YCoord_Conv=ys, ZCoord_Conv=zs
    (line[j])->GetProperty, Data = Data
    orbs[j]->SetProperty, Pos = [data[0, 1]*xs[1] + xs[0], $
     data[1, 1]*ys[1] + ys[0], data[2, 1]*zs[1] + zs[0]]
  ENDFOR
```

This appears to work for any axis data range. But *WHY* it works is a complete mystery to me. I'd be interested in hearing any good theories. :-)

Cheers,

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