Subject: 3D projection rotation
Posted by Dave Brennan on Mon, 08 Feb 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Hi,

I am currently trying to produce rotating maximum intensity projections
using the voxel_proj command.
To do this I am using scale3 to produce the !P.T transform matrix, i.e.

```
for j=0,35 do begin
   xrot = (j-9)*10

 scale3,xrange=[0,sizematx*scalex],yrange=[0,sizematy*scaley]
,zrange=[0,slicemat*scalez],ax=xrot

   img=voxel_proj(imagebyt,/maximum_intensity)
   array(*,*,j) = img
   print,j
endfor
```

(scalex etc are the sizes of the voxels in the corresponding directions)

This produces data which can be viewed in xinteranimate producing a
rotating MIP.

However, I wish to increase the size of the rotating MIP, to say twice
it's original size, is there a simple way of accomplishing this, with
the minimum of computing time? I am sure I am missing a simple solution.

Thanks for your help

Dave Brennan

Subject: Re: 3D projection rotation
Posted by Dave Brennan on Tue, 09 Feb 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Thanks for the solution.

However, I found that I had to use REBIN before the second XINTERANIMATE command, otherwise the
window was resized, but the data was not.

I have one further question though,

I use the Voxel_Proj command with a data set of 256x256x128. This produces projection data in a

window of 640x512. However, the window is much larger than the data produced so I would like to reduce the data output from Voxel_Proj to a smaller matrix to reduce the 'border' around the data. I
have tried the following,

 img=voxel_proj(imagebyt,/maximum_intensity,xsize=512,ysize=4 10)

This does reduce the size of the output data , but the projection data, which was central within the output window is now offset from the centre. How do I reduce the output size whilst keeping the data
central within the window.

thanks for you help

Dave Brennan

David Fanning wrote:

>
> Some of the relevant code is missing here, but I think
> the simplest solution is to just increase the size of
> the XInteranimate window by a factor of two. :-)
>
>    XInterAnimate, Set=[currentX*2, currentY*2, frames], /Showload
>
> Be sure this is done *before* you calculate the Scale3
> values, since it will use the size of the current display
> window in its calculations.
>
> 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
>
> [Note: This follow-up was e-mailed to the cited author.]