Subject: Re: arbitrary rotation of 3-d arrays
Posted by Michael Asten on Fri, 11 Jun 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Not sure how sophisticated you want to be here.
The !P.T structure makes it easy to start in idl. See online documentation under "Three-dimensional graphics" for starters - including the demo routine HOUSE.PRO
The demo shows how to rotate coordinates for the purpose of making a 2D projection, but the same tools work for rotating 3D arrays as abstract entities.

When I want to rotate a set of coordinates given by vectors Xin, Yin, Zin, in 3D, I generate a !P.T transformation using the following

```
;  3D coordinates before rotation are in vectors Xex,Yex,Zex
t3d,/reset
 t3d,rotate=[0.,0.,plu] & t3d,rotate=[-dip,0.,0.] & t3d,rotate=[0.,0.,str]
 t3d,translate=[xsh,ysh,zsh]
;  we have set up t3d to rotate a body thru a strike(-azimuthal)  angle str,
;     a dip angle dip, and a plunge angle plu,
; and we have added a translation of position of the refernce point of the body
;   to (xsh,ysh,zsh).
;
;  we now execute the rotation and translation
do_rotation,xex,yex,zex,xrot,yrot,zrot
;  and can plot or otherwise operate on the new rotated coordinates howsoever we ;  please
end  ;  of demo
```

The routine to do the rotation and translation is simply:

```
;  routine to perform rotation of n points in x[0:n-1],y[  ] and z[  ]
;     using the existing !P.T transformation
;  input:  xin,yin zin being arrays of reals
;  output: xout,yout,zout  being arrays of reals,  for transformed points
;  Author: Michael Asten, Monash University, Melbourne Australia.  June 1999.

 pro do_rotation,xin,yin,zin,xout,yout,zout
   P=fltarr(4,n_elements(xin))
   P[0,*]=xin & P[1,*]=yin & P[2,*]=zin & P[3,*]=1.
   P=transpose(P)
   Prot=P#!P.T  ;  do rotation and shift
   Prot=transpose(Prot)
   xout=Prot[0,*]/Prot[3,*]
   yout=Prot[1,*]/Prot[3,*]
   zout=Prot[2,*]/Prot[3,*]
 end
```

"D. Mattes" wrote:

> hello idl users:
> has anybody out there in idl-land written or seen code to apply arbitrary
> rotations to 3-d arrays???
>
> thanks in advance!
>
> david mattes
