Subject: Re: "Color vectors" & shading
Posted by David.Chevrier on Wed, 21 Jan 2004 16:45:28 GMT
View Forum Message <> Reply to Message

Hi,
  I'm not sure if this will help or not, but I had a similar (less complex) problem. I had three arrays: long/lat/depth and wanted to create one semi-transparent polygon that was green for all points above 0 (land) and blue for all points less than 0 (water). This created a bathymetric map. Here is how I did it. I know its only two colors and uses a lot of memory (my arrays had more than 9000 numbers each.) (I also used the 'temporary' command a lot to get rid of unneeded variables, but have taken most of them out here to make it easier to read.)

```
 MESH_OBJ,0,verts,conn,TRANSPOSE([[contourlng],[contourlat],[ contourdepth]])
verts=MESH_SMOOTH(TEMPORARY(verts), conn, /FIXED_EDGE_VERTICES)
depsize=(SIZE(verts))[2]
mapimage=REPLICATE(255b,4,3,depsize)
mapimage[3,*,*]=105b  ;transparent level

oContourImage=OBJ_NEW('IDLgrImage', mapimage, BLEND_FUNCTION=[3,4],
INTERLEAVE=0, /NO_COPY)

r=MAKE_ARRAY(256,/BYTE,VALUE=0b)
g=MAKE_ARRAY(256,/BYTE,VALUE=0b)
b=MAKE_ARRAY(256,/BYTE,VALUE=255b)

depths=REFORM(verts[2,*])
lands=WHERE(depths GE 0.0, anyland)
depths=BYTSCL(TEMPORARY(depths))
IF (anyland GT 0L) THEN BEGIN
  depths[TEMPORARY(lands)]=255b
  g[255]=255b
  b[255]=0b
ENDIF

oPalette1=OBJ_NEW('IDLgrPalette',r,g,b)
oWaterMap=OBJ_NEW('IDLgrPolygon',verts, POLYGONS=conn, $
  SHADING=1, TEXTURE_MAP=oContourImage, /TEXTURE_INTERP,
HIDDEN_LINES=1, $
  XCOORD_CONV=(*stateptr).xs, VERT_COLORS=depths, PALETTE=oPalette1, $
  YCOORD_CONV=(*stateptr).ys, ZCOORD_CONV=(*stateptr).zs)
```