Subject: Re: Rendering and Code like Points2polys
Posted by Sylvain Carette on Tue, 26 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
Hi Larry,
<p>This can answer you need as it continue my thread about "spherical gridding"

Here I got finally something - it seem we try to do about the same;
building a "tin" surface over a sphere.

Special thanks to Ben who point me to some good cues in private email
about the "orb" object and the CV_COORD procedure. I ran also almost by
accident on the "mesh_obj.pro" in the lib directory which use triangulate
procedure in conjunction whith cv_coord to build spherical surface among
other types.
<p>Now in the proccess of building my own "sphere_surf.pro" which I just
have tested the basic inner working.

I will be more or less based on mesh_obj interface:

sphere_surf, type, vertex_list,[/full], [/degrees], [renderStyle=value],
[shading=value], [color=value], [etc...]

in which: type 0= spherically triangulated irregular network, 1= spherically
gridded mesh

vertex_list is the input xyz vertices for type =1 and a 2 dim array
for type 1

/full is to build the mesh or tin over a full sphere

/degrees is to input the x, y as degrees

for render_style, shading, color, etc, these are the parameters of
the inner IDLgrPolygon used to construct the result - and here I'm not
sure if I should implement as a procedure, a function or derivate a new
object - suggestion welcome.
<p>Anyhow, for the moment I badly only need just the tin over the incomplete
sphere so I'll keep the rest for later.

To complete the sphere (there is still a gap in this form as I expected),
I suppose it will be a matter of using the "B" optionnal variable in triangulate
which return a list of the indices of the boundary point in ccw order and
to triangulate those again and add the resulting connectivity to the preceeding
before feeding IDLgrPolygon.

I also tested in double precision; it is OK. (although, trying to view
in xobjview generate zclipping plane error which I dont mind at all as
long the object is ok).
<p>So here the sample code which build the tin:

 ----------8<---------- --

<tt>; Create array to hold vertices</tt>

<tt>vertexlist = dblarr(3, 200)</tt>

<tt>; Create some random longitude points:</tt>

<tt>vertexlist[0, *] = RANDOMU(seed, 200) * 360. - 180.</tt>

<tt>; Create some random latitude points:</tt>

<tt>vertexlist[1, *] = RANDOMU(seed, 200) * 180. - 90.</tt>

<tt>; Set z to uniform value</tt>

<tt>vertexlist[2, *] = 500</tt><tt></tt>
<p><tt>; Triangulate a spherical surface</tt>

<tt>Triangulate, vertexlist[0, *], vertexlist[1, *], tri, /degrees</tt>

<tt>polylist = [Replicate(3L, 1L, (N_Elements(tri) / 3L)), Temporary(tri)]</tt>

<tt>polylist = polylist[*]</tt><tt></tt>
<p><tt>vertex = CV_COORD(From_Sphere=vertexlist, /To_Rect, /degrees)</tt><tt></tt>
<p><tt>; Create model.</tt>

<tt>oGroup = OBJ_NEW('IDLgrModel')</tt><tt></tt>
<p><tt>; Create the surface.</tt>

<tt>oTin = OBJ_NEW('IDLgrPolygon', vertex, polygon=PolyList, STYLE=2,
$</tt>

<tt>

SHADING=1, COLOR=[0,20,70])</tt>

<tt>oGroup->Add, oTin</tt><tt></tt>
<p><tt>oLightGroup = OBJ_NEW('IDLgrModel')</tt>

<tt>; Create some lights.</tt>

<tt>oLight = OBJ_NEW('IDLgrLight', LOCATION=[2,2,2], TYPE=1)</tt>

<tt>oLightGroup->Add, oLight</tt>

<tt>oLight = OBJ_NEW('IDLgrLight', TYPE=0, INTENSITY=0.5)</tt>

<tt>oLightGroup->Add, oLight</tt><tt></tt>
<p><tt>xobjview, oGroup, stationary=oLightGroup</tt><tt></tt>
<p><tt>end</tt>

 ----------8<---------- -----

Just run as is. If you have other ideas about how the final object
should work, I'll be glad to hear about it.

My purpose is to have a way to construct and project and merge all
kind of dem from lat/long gridded to xyz irregular surveyed points onto
the WGS84 geoid at any kind of resolution from crude to exact. Plus the
ability to map satellite imagery over the resulting tile or "patch" and
calculating exact texture coordinate on each part. Later, I'll like also
to implement "constrained" delanoy triangulation to include VMAP and higher
resolution vectors to constrain the tin along rivers, roads, etc
<p>Hope it help
<p>Sylvain Carette

VRML designer-composer
<p>Larry Busse wrote:
<blockquote TYPE=CITE>Hello,
<p>I've been given a list of xyz points on the surface of an object and
I'd

like to be able to use the IDLgrPolygon object to view them.
This

requires trigulation....generating lists of vertices (the original

points) and a list of faces (list of indices that define each polygon,

or triangle, on the surfaces.) I found a WindowsNT program by
Parasoft

called Points2Polys that will do this but it would certainly be more

convenient if I could do it directly within IDL.
<p>Has anyone written such code thqat they would be willing to share? or

could provide pointers??
<p>David Fanning's article "Gridding XYZ Triples to form a Surface Plot"
is

a step in the right direction but I was thinking the polygon approach

would be better for closed surfaces and for datasets with a large number

of points.
<p>Any thoughts on the topic would be appreciated.</blockquote>
</html>
