Subject: Re: Rendering and Code like Points2polys
Posted by Sylvain Carette on Wed, 27 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Mark Hadfield wrote:

> "Larry Busse" <ljb@ljbdev.com> wrote in message
> news:39CB7E28.596301A1@ljbdev.com...
>> 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.
>
> I am very much a novice in the area of triangulation, computational geometry
> etc, but I'll contribute my $0.02 and see who contradicts me.
>
> Do you mean that the xyz points *define* the surface of the object and you
> want to visualise the object? In that case what you want is a 3-dimensional
> triangulation of your points. (If that is not what you want then the rest of
> this post is off the topic.)
>
> The IDL triangulation procedure, TRIANGULATE,
> won't do it because it only does planar or spherical triangulations (the
> latter referring to locations on the surface of a sphere).
>
> MESH_OBJ generates 3-D triangulated data sets, but I don't think it will do
> what you want. In one of its modes of operation it triangulates irregular
> data, but this is just a planar triangulation of the x & y components of the
> data. Its other modes of operation generate various 3D objects of simplified
> geometry (extrusions, solids of revolution)

But if you look in MESH_OBJ, you'll see that for spherical gridding, it use triangulate and trigrid to generate a planar grid. Then it convert all the vertices with CV_COORD from spherical space (x, y, radius) to cartesian (x, y, z). So to get only a triangulation (without gridding) on a sphere, I used triangulate alone to get the connectivity and convert the vertices as in MESH_OBJ (look my reply on this thread). There is still a gap that must be addressed - begin to work on it.

I still suspect there is a way to use the "sphere=s" variable in triangulate since it provide vertices and connectivity, but the ordering of the vertices is differents (xxxx, yyyy, zzzz instead of xyz, xyz, xyz, xyz - why they have done this?!?!) and also the function "reorganize" the vertices order so probably the connectivity too... Anyhow, up to then, I could only get a kind of shrinkwrapped shrinkwrap blob from it... :-(I'll have to investigate how to mess with the

connectivity to get the right result.

Now if you know of a function (in any source code) that will perform delaunay triangulation from gridded data (to remove extraneous vertices keeping vertices where it is needed -up to 80% polygon reduction can be acheived without any noticable artefact), I'll be glad to hear about it

Sylvain Carette
VRML designer-composer

>
>
> SHADE_VOLUME generates 3-D meshes, but it fits iso-surfaces to 3D gridded
> datasets, which is not what you have.
>
> So AFAIK IDL will not do what you want. Needless to say, it would be
> possible to write a routine for 3-D triangulation and if you do I'd love to
> see it!
>
> I hate to say this but there is a very good spatial & geometric analysis
> toolbox for Matlab:
>
> http://puddle.mit.edu/~glenn/kirill/saga.html
>
> It has a multi-dimensional triangulation routine. You might want to take a
> look at its FAQ for an introduction to spatial & geometric analysis
> concepts.
>
> ---
> Mark Hadfield
> m.hadfield@niwa.cri.nz http://katipo.niwa.cri.nz/~hadfield/
> National Institute for Water and Atmospheric Research
> PO Box 14-901, Wellington, New Zealand
>
> Hi! I'm a .signature virus! copy me into your .signature file to help me
> spread!