Subject: Re: point inside/outside of 3D object.
Posted by Junum on Tue, 21 Jun 2011 17:22:14 GMT
View Forum Message <> Reply to Message

On Jun 21, 10:18 am, Karl <karl.w.schu...@gmail.com> wrote:
> On Jun 21, 3:24 am, Wox <s...@nomail.com> wrote:
>
>
>
>
>
>> On Sat, 18 Jun 2011 11:34:42 -0700 (PDT), Junum <junshi...@gmail.com>
>> wrote:
>
>>> Thanks Karl.
>>> I wanted know whether IDLanROI::ContainsPoints can be applied to 3D
>>> case.
>
>> I'd guess the answer is no. You should implement this yourself (as
>> Karl suggested) or you could do something like below. I'm not sure
>> whether this is the best way, but it seems to work.
>
>> ; Generate vertices
>> v=TetrahedronVertices(r=10,phideg=-20)
>
>> ; Connectivity list: [n,i[0],...,i[n-1],n,j[0],...,j[n-1],...]
>> ; n: number of vertices for each face
>> ; i[0],...,i[n-1]: vertices for face 1, ordered so that the normal
>> ; points outwards (right-hand rule)
>> ; j[0],...,j[n-1]: vertices for face 2, ordered so that the normal
>> ; points outwards (right-hand rule)
>> conn=[3,0,3,1, 3,0,1,2, 3,0,2,3, 3,1,3,2]
>
>> ; Remark: if the number of vertices > 4 then you could generate
>> ; the list like this:
>> ;Qhull, v, tr, /delaunay
>> ;conn=tetra_surface(v, tr)
>
>> ; Point
>> p=[0,0,0.]
>
>> ; Volume of the polyhedron
>> volume=tetra_volume(v,conn)
>
>> ; Expanded polyhedron (including your point)
>> ; vertices and connectivity list
>> v2=[[v],[p]]
>> Qhull, v2, tr, /delaunay

>> conn2=tetra_surface(v2, tr)
>
>> ; Volume of the expanded polyhedron
>> volumeexp=tetra_volume(v2,conn2)
>
>> ; If the "expanded volume" is larger, the point lies outside
>> if volumeexp gt volume then print,'Exterior' else print,'Interior'
>>> I wanted know whether IDLanROI::ContainsPoints can be applied to 3D
>>> case.
>
>> I'd guess the answer is no.
>
> Right. Although not explicitly stated, I think that most of the anROI
> support is for 2D ROI's even though you can specify points in 3D
> space. The ROI's were intended for image analysis, I think. I
> understand that ROI's can be 1D and 3D as well, but I think that the
> algorithms in anROI are intended for 2D. Note that there are methods
> to compute such things as area and perimeter, but not volume and
> surface area. I also recall that some of the anROI algorithms will go
> ahead and try to do things like compute areas for non-planar ROI's
> even though the result may be questionable. Other algorithms might
> actually go ahead and do a planarity check and throw an error if the
> data wasn't planar.
>
> The expanding polyhedron approach is clever! I like it.

Thank you very much, Wox and Karl.

Jun
