Subject: Re: volume() command
Posted by Mark Piper on Mon, 25 Mar 2013 20:36:40 GMT
View Forum Message <> Reply to Message

On Saturday, March 23, 2013 5:42:27 PM UTC-6, David Grier wrote:
> Volumetric rendering being one of the things that IDL does better than Matlab or python's
>
> matplotlib, it might seem surprising that IDL does not have a volume() function analogous to the other
>
> function graphics routines. In fact, IDL's function graphics system actually can create
>
> a volumetric rendering analogous to the iTools ivolume routine, but with better programmability.
>
> To create a volume,
>
>
>
> IDL> graphic, 'volume', data, _extra = ex, graphic = vol
>
>
>
> DATA should be a [nx, ny, nz] volumetric data set.
>
> VOL is the resulting object.
>
> Keywords are passed through the extra mechanism.
>
>
>
> To learn what properties are available,
>
> IDL> itpropertyreport, vol.gettool(), igetid('volume')
>
>
>
> Although this procedure yields a full-fledged function graphics object, it has some compatibility
>
> quirks when used in combination with image() and plot() commands. It would be nice if IDL's
>
> developers were to box up and document a well-integrated implementation of volume().
>
>
>
> Just my 2 cents,
>
>
>

> David

Hi David,

VOLUME may make it into 8.2.3. If not, it'll be in 8.3.

mp

Subject: Re: volume() command
Posted by Dick Jackson on Thu, 28 Mar 2013 01:59:04 GMT
View Forum Message <> Reply to Message

Mark Piper wrote:
> Hi David,
>
> VOLUME may make it into 8.2.3. If not, it'll be in 8.3.
>
> mp

Hi Mark,

Are there plans for similar handy functions for a polygon mesh or polyline in its own window?

Right now we need to do this:

```
mesh_obj,3,verts,conn,replicate(2,[9,9]) ; Make data for rough cylindrical tube
verts[0:1,*] += 2                        ; Move so X, Y ranges are [0..4]
tube =  POLYGON((verts[0,*])[*],(verts[1,*])[*],(verts[2,*])[*],conn =conn, /Data)

; % POLYGON: Graphics window does not exist.
; % Execution halted at: $MAIN$
; Oops, need a graphics window first...

surf = surface(randomu(seed, [5, 5]))
tube =  POLYGON((verts[0,*])[*],(verts[1,*])[*],(verts[2,*])[*],conn =conn, /Data)
; That works, with awkward vertex data handling that requires vectors.

; Easy data handling would be like this:
tube2 = Obj_New('IDLgrPolygon', Data=verts, Polygons=conn, Color=[128,128,128])
XObjView, tube2
; It's there, you just have to rotate the view to see it :-)


; The POLYGON() function provides for annotating an existing view.
; So, it would be nice to have something like...
tube3 = POLYGONMESH(verts, conn=conn)
```

; ... to act like SURFACE(), etc.

Is that in the works, too? :-)

--

Cheers,
-Dick

Dick Jackson Software Consulting
Victoria, BC, Canada
www.d-jackson.com

> On Saturday, March 23, 2013 5:42:27 PM UTC-6, David Grier wrote:
>> Volumetric rendering being one of the things that IDL does better than Matlab or python's
>>
>> matplotlib, it might seem surprising that IDL does not have a volume() function analogous to the other
>>
>> function graphics routines. In fact, IDL's function graphics system actually can create
>>
>> a volumetric rendering analogous to the iTools ivolume routine, but with better programmability.
>>
>> To create a volume,
>>
>>
>>
>> IDL> graphic, 'volume', data, _extra = ex, graphic = vol
>>
>>
>>
>> DATA should be a [nx, ny, nz] volumetric data set.
>>
>> VOL is the resulting object.
>>
>> Keywords are passed through the extra mechanism.
>>
>>
>>
>> To learn what properties are available,
>>
>> IDL> itpropertyreport, vol.gettool(), igetid('volume')
>>
>>
>>
>> Although this procedure yields a full-fledged function graphics object, it has some compatibility

>>
>> quirks when used in combination with image() and plot() commands. It would be nice if IDL's
>>
>> developers were to box up and document a well-integrated implementation of volume().
>>
>>
>>
>> Just my 2 cents,
>>
>>
>>
>> David
