
Subject: Re: voxel_proj and seg fault

Posted by [Jacques Basson](#) on Thu, 28 Feb 2002 09:08:38 GMT

[View Forum Message](#) <> [Reply to Message](#)

That is the correct result for the test data (taken from the online help for project_vol btw). Sorry, I didn't think that it was important that the final result was just a skewed block. Reverse the opacities
rgbo[:,3] = reverse(bindgen(256))
just before the call to voxel_proj to give a more funky looking skewed block (make the darker colours opaque).

OK, so I found a version 5.4 and loaded that up. Here is a summary of the results:

5.4:

test data - all zeros

test data without /interpolate - all zeros

real data - all zeros

real data without /interpolate - all zeros

5.5a (5.5 patched with 24-bit colour bug patch):

test data - segfault

test data without /interpolate - correct projection (skewed block)

real data - segfault

real data without /interpolate - correct projection

This is consistent with Rick getting all zeros on 5.4 (don't know which version David was running when he tried).

So it looks like it is not working properly in 5.4, and is half fixed in 5.5 (and then maybe broken in 5.5a). Anyway, I am well underway to writing my own projection routine (should be quicker than trying to resolve this one).

Thanks for all the help.

Cheers,
Jacques

Jeff Guerber wrote:

> Seems to run OK on { hp_pa hp-ux unix hp-ux 5.5 Aug 28 2001 32 64}.
> "result" is all 0 or 255, with all 3 planes identical. When displayed
> with tv they appear to be slightly skewed blocks, all white except for
> black in the upper left and lower right corners. No underflow messages.

>

> Jeff Guerber

>

> On Wed, 27 Feb 2002, Jacques Basson wrote:

```
>
>
>> Hi all
>>
>> I seem to be getting segmentation faults when using the interpolate
>> keyword to voxel_proj (strangely enough, removing the rotations gets rid
>> of the segfault, but then that's not exactly useful). It is repeatable
>> on several machines running linux or solaris. xvolume works fine, but is
>> not configurable enough (automating translations / getting a contour
>> plot instead of an image out of the thing...).
>>
>> Does anyone know of a suitable workaround, apart from not using the
>> interpolate keyword - it's nice to have smooth-looking final plots :)
>>
>> Cheers,
>> Jacques
>>
>> -----
>> IDL> print, !version
>> { x86 linux unix linux 5.5a Feb 7 2002    32    32}
>> IDL> n = 192
>> IDL> vol = randomu(1, n+6, n+6, n+6)
>> IDL> for i=0,10 do vol = smooth(vol, 3)
>> IDL> vol = bytscl(vol(3:n+2, 3:n+2, 3:n+2))
>> IDL> !x.s = [0.,1.] / (n-1)
>> IDL> !y.s = [0.,1.] / (n-1)
>> IDL> !z.s = [0.,1.] / (n-1)
>> IDL> t3d, /reset, translate=[-0.5,-0.5,-0.5]
>> % Compiled module: T3D.
>> IDL> t3d, rotate=[60,0,0]
>> IDL> t3d, rotate=[0,0,-80]
>> IDL> t3d, translate=[0.5,0.5,0.5]
>> IDL> rgbo = bindgen(256)#[1,1,1,1]
>> IDL> result = voxel_proj(vol, rgbo, /interpolate)
>> Segmentation fault
>> -----
>>
>
```
