Subject: Re: voxel_proj and seg fault
Posted by David Fanning on Wed, 27 Feb 2002 14:03:32 GMT
View Forum Message <> Reply to Message

Jacques Basson (jfb37@removeme.cam.ac.uk) writes:

> 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 :)

The code you provided runs on my Windows machine,
but I get a floating underflow warning and the
result contains all zeros. Are you sure the code
is doing what you *think* it is doing?

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: voxel_proj and seg fault
Posted by Jacques Basson on Wed, 27 Feb 2002 15:22:17 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
> Jacques Basson (jfb37@removeme.cam.ac.uk) writes:
>
>
>> 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 :)
>>
>
> The code you provided runs on my Windows machine,
> but I get a floating underflow warning and the
> result contains all zeros. Are you sure the code
> is doing what you *think* it is doing?
>
> Cheers,
>
> David

When I run it without the /interpolate, I get the proper result (using
my data rather than the data generated in the provided code - so I know
more or less what it is supposed to look like). Running the code
provided without interpolate and doing "tvscl, result, true=3" provides
a cube rotated appropriately. And result is definitely not all zeros.

.....
IDL> result = voxel_proj(vol, rgbo)
IDL> help, result
RESULT   BYTE   = Array[640, 512, 3]
IDL> help, max(result)
<Expression>   BYTE   = 255
IDL> help, min(result)
<Expression>   BYTE   =   0
IDL> print, moment(result)
% Compiled module: MOMENT.
      246.125   2173.24   -5.12102   24.0455

Cheers,
Jacques

Subject: Re: voxel_proj and seg fault
Posted by David Fanning on Wed, 27 Feb 2002 15:37:37 GMT
View Forum Message <> Reply to Message

Jacques Basson (jfb37@removeme.cam.ac.uk) writes:

> When I run it without the /interpolate, I get the proper result (using
> my data rather than the data generated in the provided code - so I know
> more or less what it is supposed to look like). Running the code
> provided without interpolate and doing "tvscl, result, true=3" provides
> a cube rotated appropriately. And result is definitely not all zeros.

Well, with the code you provided, I get zeros with or without

the Interpolate. So, I guess I can't help. :-(

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: voxel_proj and seg fault
Posted by Rick Towler on Wed, 27 Feb 2002 18:13:23 GMT
View Forum Message <> Reply to Message

I had the same level of success that David had:

On Solaris:

```
IDL> print, !version
{ sparc sunos unix 5.4.1 Jan 16 2001      32      64}
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)
% Program caused arithmetic error: Floating underflow
IDL> print,min(result)
   0
IDL> print,max(result)
   0
```

On Win32:

```
IDL> print,!version
{ x86 Win32 Windows 5.4 Sep 25 2000      32      64}
```

```
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)
% Program caused arithmetic error: Floating underflow
IDL> tv,result
IDL> print,max(result)
   0
IDL> print,min(result)
   0
```

-Rick

"Jacques Basson" <jfb37@removeme.cam.ac.uk> wrote in message news:3C7CB205.8070100@removeme.cam.ac.uk...

> 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
> --
>

Subject: Re: voxel_proj and seg fault
Posted by Jeff Guerber on Wed, 27 Feb 2002 20:11:08 GMT
View Forum Message <> Reply to Message

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
> ----------

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 wad 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
>> --
>>
>