Subject: Re: OG polygon to EPS problem
Posted by David Fanning on Thu, 29 Aug 2002 18:58:00 GMT
View Forum Message <> Reply to Message

Mirko Vukovic (mvukovic@taz.telusa.com) writes:

> In the following program, if I have hidden_lines on, the post-script
> is shaded black. If I have hidden_lines off, I can see the grid.
> This is in IDL5.4.on WinNT. The behavior with hidden_lines on does
> not make sense to me. Anyone else?
>
> My real application is a 3D polygon, with hidden lines on. However,
> all I currently get is a black blob. If I use surface elevation
> shading, hidden_lines have no effect, and I can trick IDL somewhat by
> using black-white color map with the gamma correction at 10.

I think it is a mistake to think "vector" output is going
to work with polygons. I'd leave that keyword off, and then
see if the rest of it makes sense. I can't really tell here
what I am suppose to be seeing. :-)

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
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: OG polygon to EPS problem
Posted by Mark Hadfield on Thu, 29 Aug 2002 21:37:24 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.17d81ca4bab3db5098998a@news.frii.com...

> Mirko Vukovic (mvukovic@taz.telusa.com) writes:
>
>> In the following program, if I have hidden_lines on, the
>> post-script is shaded black. If I have hidden_lines off, I can see
>> the grid. This is in IDL5.4.on WinNT. The behavior with
>> hidden_lines on does not make sense to me. Anyone else?
>

> I think it is a mistake to think "vector" output is going to work
> with polygons. I'd leave that keyword off, and then see if the rest
> of it makes sense. I can't really tell here what I am suppose to be
> seeing. :-)

I concur with David. IDL Object Graphics is based on OpenGL, which provides for rendering to bitmap destinations, but not to vector graphics formats. When first introduced in version 5.0, OG had the same limitation. RSI then reluctantly added vector output in version 5.1 or 5.2. (I think they got sick of hearing users say "My OG line plot generates a 6 MB printer file, it takes 10 minutes to print, and it doesn't look that hot either.") They warned that some features of OG scenes would not be rendered properly in vector formats, but what may surprise some people is *how many* features this warning applies to. For example, fill patterns (as specified by the FILL_PATTERN keyword) seem like pretty vector-ish entities, but they are not rendered in vector formats. And I guess the same applies to HIDDEN_LINES.

--
Mark Hadfield "Ka puwaha te tai nei, Hoea tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)

---

Subject: Re: OG polygon to EPS problem
Posted by Karl Schultz on Thu, 29 Aug 2002 22:43:52 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.17d81ca4bab3db5098998a@news.frii.com...
> Mirko Vukovic (mvukovic@taz.telusa.com) writes:
>
>> In the following program, if I have hidden_lines on, the post-script
>> is shaded black. If I have hidden_lines off, I can see the grid.
>> This is in IDL5.4.on WinNT. The behavior with hidden_lines on does
>> not make sense to me. Anyone else?
>>
>> My real application is a 3D polygon, with hidden lines on. However,
>> all I currently get is a black blob. If I use surface elevation
>> shading, hidden_lines have no effect, and I can trick IDL somewhat by
>> using black-white color map with the gamma correction at 10.
>
> I think it is a mistake to think "vector" output is going
> to work with polygons. I'd leave that keyword off, and then
> see if the rest of it makes sense. I can't really tell here
> what I am suppose to be seeing. :-)

I know what the problem is.

The short answer is to not try to do hidden line removal in vector mode. Hidden line/hidden surface removal (HLHSR) is often thought of as a "raster-space" operation, rather than a "vector" operation. That is, in vector mode, vectors and shapes are drawn without considering what other vectors or shapes have already been drawn. In vector mode, there is no frame buffer or "raster space". Thus, we don't do HLHSR on a vector device. The IDL docs do say that vector output is unable to reproduce certain effects, such as shading and depth buffering operations. We sort things in Z to make the best picture we can, however.

Here's why you see solid polygons when you turn on HLHSR. When drawing a wireframe polygon in hidden-line mode, IDL first draws the polygon in fill mode in the background color. These "invisible" faces block the parts of the polygon that are behind the invisible faces. Then, when we draw the wireframe in the specified color, the Z buffer "clips" out the part of the lines that are behind the invisible faces, leaving just the lines that are visible.

The reason why the polygons were black here instead of white is that the temporary color override (to the background color) didn't make it out to the vector output, and so they came out black. They should have been drawn white. But even if they were white, they wouldn't do what they were supposed to do in vector mode (no Z buffer), so I don't think they should be drawn at all. I'll look into fixing this. It is better to just have the wireframe draw without HLHSR than have the shield polygons draw.

Karl

---

Subject: Re: OG polygon to EPS problem
Posted by mvukovic on Fri, 30 Aug 2002 17:11:42 GMT
View Forum Message <> Reply to Message

"Karl Schultz" <kschultz@devnull.researchsystems.com> wrote in message news:<akm838$4lb$1@news.rsinc.com>...
> "David Fanning" <david@dfanning.com> wrote in message
> news:MPG.17d81ca4bab3db5098998a@news.frii.com...
>> Mirko Vukovic (mvukovic@taz.telusa.com) writes:
>>
>>> In the following program, if I have hidden_lines on, the post-script
>>> is shaded black. If I have hidden_lines off, I can see the grid.
>>> This is in IDL5.4.on WinNT. The behavior with hidden_lines on does
>>> not make sense to me. Anyone else?
>>>
>>> My real application is a 3D polygon, with hidden lines on. However,
>>> all I currently get is a black blob. If I use surface elevation

>>> shading, hidden_lines have no effect, and I can trick IDL somewhat by
>>> using black-white color map with the gamma correction at 10.
>>
>> I think it is a mistake to think "vector" output is going
>> to work with polygons. I'd leave that keyword off, and then
>> see if the rest of it makes sense. I can't really tell here
>> what I am suppose to be seeing. :-)
>
> I know what the problem is.
>
> The short answer is to not try to do hidden line removal in vector mode.
> Hidden line/hidden surface removal (HLHSR) is often thought of as a
> "raster-space" operation, rather than a "vector" operation. That is, in
> vector mode, vectors and shapes are drawn without considering what other
> vectors or shapes have already been drawn. In vector mode, there is no
> frame buffer or "raster space". Thus, we don't do HLHSR on a vector device.
> The IDL docs do say that vector output is unable to reproduce certain
> effects, such as shading and depth buffering operations. We sort things in
> Z to make the best picture we can, however.
>
> Here's why you see solid polygons when you turn on HLHSR. When drawing a
> wireframe polygon in hidden-line mode, IDL first draws the polygon in fill
> mode in the background color. These "invisible" faces block the parts of
> the polygon that are behind the invisible faces. Then, when we draw the
> wireframe in the specified color, the Z buffer "clips" out the part of the
> lines that are behind the invisible faces, leaving just the lines that are
> visible.
>
> The reason why the polygons were black here instead of white is that the
> temporary color override (to the background color) didn't make it out to the
> vector output, and so they came out black. They should have been drawn
> white. But even if they were white, they wouldn't do what they were
> supposed to do in vector mode (no Z buffer), so I don't think they should be
> drawn at all. I'll look into fixing this. It is better to just have the
> wireframe draw without HLHSR than have the shield polygons draw.
>
> Karl

You see, all this is really David's fault. I really liked his
fsc_surface routine, and how easily I was able to modify it to plot
polygons instead of surfaces. And polygons are important to me, as
our data is inherently non-grid like, and polygon plots showed the
real data, with no other grids overlaid as result of tri-gridding.

So, I am converted to OG for these types of applications. But how can
I get then a decent quality rasterised printer output? If the
information is available on-line or in documentation, just give me the

keywords to facilitate the search (and I'll accept a flame or two for
not figuring out where to find it).

Thanks for all the responses,

Mirko

---

Subject: Re: OG polygon to EPS problem
Posted by David Fanning on Fri, 30 Aug 2002 17:48:24 GMT
View Forum Message <> Reply to Message

Mirko Vukovic (mvukovic@taz.telusa.com) writes:

> You see, all this is really David's fault.

Oh, right. I'll take the blame. :-)

> I really liked his
> fsc_surface routine, and how easily I was able to modify it to plot
> polygons instead of surfaces. And polygons are important to me, as
> our data is inherently non-grid like, and polygon plots showed the
> real data, with no other grids overlaid as result of tri-gridding.
>
> So, I am converted to OG for these types of applications. But how can
> I get then a decent quality rasterised printer output?

Have you tried removing the VECTOR keyword? What happened?
I should think you would get a nice polygon surface. That's what
I get when I save FSC_SURFACE output without setting the
VECTOR keyword, anyway.

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
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: OG polygon to EPS problem
Posted by Karl Schultz on Fri, 30 Aug 2002 21:53:40 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.17d95dd7d6f4b57298998b@news.frii.com...
> Mirko Vukovic (mvukovic@taz.telusa.com) writes:
>
>> You see, all this is really David's fault.
>
> Oh, right. I'll take the blame. :-)
>
>> I really liked his
>> fsc_surface routine, and how easily I was able to modify it to plot
>> polygons instead of surfaces. And polygons are important to me, as
>> our data is inherently non-grid like, and polygon plots showed the
>> real data, with no other grids overlaid as result of tri-gridding.
>>
>> So, I am converted to OG for these types of applications. But how can
>> I get then a decent quality rasterised printer output?
>
> Have you tried removing the VECTOR keyword? What happened?
> I should think you would get a nice polygon surface. That's what
> I get when I save FSC_SURFACE output without setting the
> VECTOR keyword, anyway.

Yes, as David says, using bitmap mode (the default) will give you the same quality as you'd get on the screen. Also, you can use the Dimensions keyword to make the clipboard bigger, if you need more pixels.

IDLgrPrinter will also do a good job in bitmap (default) mode by using the full resolution of the printer.

If you insist on using vectors, I found a workaround for the original problem. If you specify VERT_COLORS, you'll avoid the bug and draw that pyramid with hidden line removal. Since your polygon is black, just add "oPolygon->SetProperty, VERT_COLORS=BYTARR(3,5)", or initialize the vert_colors array to whatever color you need.

Karl

---

Subject: Re: OG polygon to EPS problem
Posted by mvukovic on Tue, 03 Sep 2002 16:00:14 GMT
View Forum Message <> Reply to Message

"Karl Schultz" <kschultz@devnull.researchsystems.com> wrote in message
news:<akoph2$rd8$1@news.rsinc.com>...
> "David Fanning" <david@dfanning.com> wrote in message
> news:MPG.17d95dd7d6f4b57298998b@news.frii.com...
>> Mirko Vukovic (mvukovic@taz.telusa.com) writes:
Two replies

David:

>> Have you tried removing the VECTOR keyword? What happened?
>> I should think you would get a nice polygon surface. That's what
>> I get when I save FSC_SURFACE output without setting the
>> VECTOR keyword, anyway.

The little test routine used the Vector kwd. That is when the problem
of the black polygons appears.

Kar:
> If you insist on using vectors, I found a workaround for the original
> problem. If you specify VERT_COLORS, you'll avoid the bug and draw that
> pyramid with hidden line removal. Since your polygon is black, just add
> "oPolygon->SetProperty, VERT_COLORS=BYTARR(3,5)", or initialize the
> vert_colors array to whatever color you need.
>
On my real data it worked, almost (it is more complex than the flat
figure I submitted). IDL generated the plot OK, but when I exported
into eps, the polygon lines were not of constant thickness. Some were
thicker than others, and some non-existent.

Furhtermore, I'm a bit unhappy with the tick-marks in eps output.
They are not parallel. I suspect it is due to the drawing commands
were geared to a pixelated device and being literally translated to
the vector output). I'll try to come up with a little test routine,
but it will take time, as I am not versed in OG.

I am also re-considering my insistance on eps. After all, all I need
is file that can be included into a print version of a document.
Thus, I need a high resolution output file. So I'll try to figure
that one out. I've seen several kwds dealing with dimensions and
pixels.

The reason I prefer eps, is that I can easily include it into latex
documents, and also convert it after the fact to any format I want
(tiff, jpeg, png, bmp, etc) using ghostscript. This last format (etc)
is a proprietary format with which I will achieve world domination of
picture and figure formats :-)

Thanks to all so far.

Mirko