Subject: Re: Changing the Colour Table for RGB images on Linux/Solaris
Posted by Karl Schultz on Fri, 26 Mar 2004 17:57:51 GMT
View Forum Message <> Reply to Message

You can use a color table in this situation if your machine has a DirectColor visual.

You can check this by running the command 'xdpyinfo' and see if the X server lists a DirectColor visual.

Another way is to issue 'HELP, /DEVICE' in IDL and see which visual you are using. However, this may be affected by any DEVICE commands you may have in an IDL startup file.

Anyway, I just sat down at a machine with a DirectColor 24-bit visual and issued:

```
read_jpeg, FILEPATH('rose.jpg', SUBDIRECTORY=['examples','data']), image

v = 255 - BINDGEN(256)

tvlct, v, v, v

tv, image, /true
```

and I got the expected reverse-colored image.

There is also a useful note in the IDL documentation that discusses X Windows visuals:

"IDL's color table does not map directly to a hardware color table for a TrueColor visual. If IDL's color table is modified, for example using the LOADCT or TVLCT routines, then the new color table will only take effect for graphics that are drawn after it has been modified."

This makes sense because IDL cannot modify a static color map.

However, you do need to pay attention to the color table translation control in the DEVICE command.
If you are using a TrueColor visual and you want to really apply a color table while writing the image to the screen, issue

```
DEVICE, BYPASS_TRANSLATION=0
```

and then the above code will draw the reverse-colored image while using a TrueColor visual.

The default setting for bypass_translation is a little complicated, but in this case, the translation is bypassed for performance reasons. In most cases, the true-color image is ready to be displayed without modification. But you do get the choice of applying the color table to your image data yourself, or letting IDL do it via the bypass_translation control in the DEVICE command.

Karl

"Philip Kershaw" <p.j.kershaw@rl.ac.uk> wrote in message
news:c41n71$3ds@newton.cc.rl.ac.uk...
> Yes, I did need to check that I'd done that. The problem persisted but it
> seems I have a solution in that I actually apply the RGB stretch to the
> image itself.
>
> Thanks for your help,
> Phil
> "Antonio Santiago" <d6522117@est.fib.upc.es> wrote in message
> news:406441FD.6010606@est.fib.upc.es...
>> Maybe you forgot update your image with the new colors. As you see in
>> "color_example.pro" of the master Fanning, he cautch the event
>> "xcolors_load" and then does redraw the image.
>>
>> Bye :)
>>
>>
>> Philip Kershaw wrote:
>>> Hello,
>>>
>>> I'm developing a colour editor for use with an image display program
for
>>> Linux and Sun Solaris. I've found that when I update the colour table
> for
>>> an RGB image, the image display isn't updated. However, when I run
the
> same
>>> program under Windows 2000, the image IS updated as expected.
>>>
>>> I'm running IDL 6.0 set with
>>>
>>> Device, Decomposed=0
>>>
>>> The display is 24-bit in each case for Windows, Linux and Solaris
> platforms.
>>>
>>> When I alter the colour table I re-display the image using
>>>

>>> TV, image, True=3
>>>
>>> The image has dimensions (m, n, 3)
>>>
>>> As a test, I tried adapting one of Dave Fanning's programs (thank you
> for
>>> this!), "color_example.pro" to run loading the RGB image. Again, it
> works
>>> under Windows but not Linux or Solaris.
>>>
>>> Has anyone come across anything like this? Any ideas?
>>>
>>> With thanks,
>>> Phil
>>> ____
>>>
>>> Philip Kershaw
>>> Space Science & Technology Department
>>> Rutherford Appleton Laboratory
>>> UK
>>> ____
>>>
>>>
>>>
>>>
>>
>
>