Subject: Re: TV oddities
Posted by davidf on Thu, 01 Jul 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Axel vom Endt (endt@bu.edu) writes:

> I just ran into some odd behaviour with TVing an array to the PostScript
> device. I get an additional column in color 255 on the right side of the
> image. No problem occurs if the number of columns is even or the
> X-Window evice is used.

Actually, I found that it works correctly if I just set
the Bits_per_Pixel keyword on the DEVICE command to 8.
(Which is what I would expect to have to do given the
numbers you use in your image.) But why this should
be so is a mystery to me. :-(

Cheers,

David

P.S. I don't have time to pursue it this week, but
I am curious if anyone else has some time.

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

Subject: Re: TV oddities
Posted by Axel vom Endt on Fri, 02 Jul 1999 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Actually, I found that it works correctly if I just set
> the Bits_per_Pixel keyword on the DEVICE command to 8.
> (Which is what I would expect to have to do given the
> numbers you use in your image.) But why this should
> be so is a mystery to me. :-(
>
> Cheers,
>
> David
>

> P.S. I don't have time to pursue it this week, but
> I am curious if anyone else has some time.

Thanks, David, for looking into this.

Well, I played with TV a little more, andgot an idea what's going on here.
From the following code I get the impression that the number of columns in an array must be an integer multiple of 8/Bits_Per_Pixel to get a correct PostScript output. (I know that I would use a setting of Bits_Per_Pixel=8 for those data, but that's not the point.)

```
;-- cut here ---
titles       = ['nine', 'eight', 'seven', 'six', 'five', 'four', $
                'three', 'two', 'one']
columns      = [9, 8, 7, 6, 5, 4, 3, 2, 1]
bitsperpixel = [1, 2, 4, 8]

; some test data
image = [$
[ 255B, 179B, 130B, 165B, 120B,  85B,  97B, 151B, 144B],$
[ 102B, 102B,   0B, 124B, 179B, 119B, 118B, 154B, 124B],$
[   9B,  91B,  35B,  89B, 191B,  82B,  94B, 169B,  31B],$
[  41B,  90B,  66B,  74B, 151B,  76B,  58B, 142B,   0B] ]

FOR f = 0, 3 DO BEGIN

   filename = 'bpp' + StrTrim(String(bitsperpixel[f]), 2) + '.ps'
; set the plot position and ranges
   Set_Plot, 'ps'
   DEVICE, Filename=filename, /Landscape, /Color, $
     BITS_PER_PIXEL=bitsperpixel[f]

   position = [0.1, 0.1, 0.9, 0.9]
   xsize = (position[2] - position[0]) * !D.X_VSize
   ysize = (position[3] - position[1]) * !D.Y_VSize
   xstart = position[0] * !D.X_VSize
   ystart = position[1] * !D.Y_VSize

; use an inverted color table
   Loadct, 0
   Tvlct, r, g, b, /GET
   Tvlct, Reverse(r), Reverse(g), Reverse(b)

; plot the image
   FOR i = 0, n_elements(titles)-1 DO BEGIN
      IF i GT 0 THEN Erase
      TV, image[0:columns[i]-1, *], xstart, ystart, XSize=xsize,
YSize=ysize
        Plot, [1], /nodata, /noerase, position=position, color=255, $
          title=titles[i], $
          xstyle=1, xrange=[0, columns[i]], $
          ystyle=1, yrange=[0, (size(image))[2]]
    ENDFOR

    DEVICE, /Close

ENDFOR

END
;-- cut here ---
```