

---

Subject: Re: Color table question

Posted by [hofer](#) on Tue, 11 Aug 1992 10:13:28 GMT

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

---

In article <1992Aug10.223854.14071@news2.cis.umn.edu>, patel@sparky.drad.umn.edu writes:

> In PV-WAVW, I need to display a gray scale image(256 grays). I need  
> few colors for graphics. I did the following to accomplish this:

>

>

> loadct,0 ; load a gray scale color table.

> ; Load the color table with 7 colors.

> Tvlct, 140, 140, 140, 1

> Tvlct, 130, 140, 220, 2

> Tvlct, 90, 80, 160, 3

> Tvlct, 255, 255, 255, 4

> Tvlct, 200, 200, 60, 5

> Tvlct, 50, 170, 80, 6

> Tvlct, 200, 70, 70, 7

>

> ; Byte scale the image in 'im'

> imb=bytsc1(im)

> ;to avoid funny colors in the image, set all pixels with value <10 to 10

> imb(where(imb lt 10))=10

> tvscl,imb

> end

>

>

> with this segment of the program, I donot expect to see any colors

> in my image. But I do see scattered pixels with different colors.

> What am I doing wrong here.

> Thanks for your help.

>

> Maqbool Patel

> patel@hippy.drad.umn.edu

Use `tv' instead of `tvscl' to plot the image. `tvscl' scales the image into the range 0..255, so all pixels with the value 10 in `imb' will become 0.

BTW, wouldn't it be more sensible to scale your data linear between 10 and 255 instead of setting all values <10 to 10. So I would use:

```
; Byte scale the image in 'im' to the range 10..255
```

```
imb = bytsc1(im, TOP=245) + 10B
```

```
; dump the image to the screen
```

```
tv, imb
```

Hope this helps.

Remo Hofer

--

RFC822: <hofer@urz.unibas.ch> or <hofer%urz.unibas.ch@CERNVAX.BITNET>

X.400: S=hofer;OU=urz;O=unibas;P=SWITCH;A=ARCOM;C=CH

HEPNET/SPAN: CHGATE::YOGI::HOFER or 20579::48130::HOFER

---

---

Subject: Re: Color table question

Posted by [snoopy](#) on Fri, 14 Aug 1992 21:38:36 GMT

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

---

If, after trying all the previous suggestions, the image still doesn't improve the problem may lie in the distribution of the data across the dynamic range of the display system. This can be improved by normalizing the data-set so that the frequency distribution (histogram) or the pdf (probability density function) of the data will be flat across the entire range of the data or the display system.

You can do a histogram equalization (or normalization) on the data with the following command:

```
newdata = HIST_EQUAL(olddata)
```

and then redisplay the modified data.

Although this will improve the dynamic range of the data itself, the result still depends somewhat upon the choice of the color-table.

--rv

P.S.

Also, in the original posting where the data was scaled to fit into bytes (using BYTSCL) and values less than 10 were zeroed out, the following version of BYTSCL using keywords min and max may be used instead with perhaps better results:

```
newdata = BYTSCL(olddata, min=10, max=max(olddata))
```

---