Subject: Re: Constructing Color Tables in IDL
Posted by Kirt Schaper on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Folks,
>
> Several people have asked me this week how to construct a
> color table. I've taken this to mean (perhaps erroneously)
> that there is general interest in the subject. :-)
>
> So here is how I might go about it. I'll construct a simple
> color table first, then a more complicated one to give you
> the general idea of how even very complex color tables might
> be constructed.
>
> First, suppose you want a color table that goes from yellow
> in the first color index to red in the last color index. And
> suppose further that you want to have 200 colors or shades
> in the color table. You know that the color yellow is
> represented by the RGB color triple (255,255,0) and that
> the color red is represented by the RGB triple (255,0,0).
>
> You know also that to create a color table you need three
> vectors, representing the red, green, and blue color values
> of the colors that make up the color table. The trick, therefore,
> is to know how to make up those color vectors. Another way
> to say this is that we need to know the RGB values of all
> the colors in a smooth progression from yellow to red.

[rgb implementation deleted]

Why not deal directly with the colors in the natural
hue/liteness/saturation space? That is, convert color1 to hls1,
convert color2 to hls2, and linearly interpolate between hue1
and hue2, converting each resulting hls to rgb space.

```
; start out with rgb values for red [255,0,0]
IDL> rr = 255   &   rg = 0   &   rb = 0
; convert to hls space
IDL> color_convert,rr,rg,rb,rh,rl,rs,/rgb_hls
IDL> print,rh,rl,rs
     0.00000     0.500000      1.00000

; convert ending rgb values (for yellow) to hls space
IDL> yr = 255   &   yg = 255   &   yb = 0
IDL> color_convert,yr,yg,yb,yh,yl,ys,/rgb_hls
```

```
IDL> print,yh,yl,ys
      60.0000     0.500000      1.00000

; (note that these values are fully saturated, and half way between
; white and black wrt liteness)

; now interpolate between the hues, converting each hls value to rgb
IDL> steps = 5
IDL> .run
- for h=rh,yh,(yh-rh)/(steps-1) do begin
-    color_convert,h,0.5,1.0, r,g,b, /hls_rgb
-    print,r,g,b
- endfor
- end
 255   0   0
 255  63   0
 255 127   0
 255 191   0
 255 255   0
```

This guarantees that the colors are maximally spread in hue space.
Unfortunately, hue space is not perceptual space (or indeed
monitor display space), so the colors may NOT be maximally
separate wrt a human observer on a given monitor.

Kirt Schaper | Bell: (612) 725-2000 x4791
PET Center (11P) | net: kirt@pet.med.va.gov
VA Medical Center | FAX: (612) 725-2068
MPLS, MN 55417 | URL: http://pet.med.va.gov:8080