
Subject: Re: Colour maps overlaid on grey-scale (medical) images

Posted by [David Foster](#) on Thu, 11 Mar 1999 08:00:00 GMT

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

Jason Brookes wrote:

>
> Hi,
>
> I would like to know how to display colour overlays on medical images.
> For example, an overlay of bloodflow rate superimposed (in hot body
> colour scale) on grey-scale images of the brain. At the moment, I am not
> able to do this without obliterating the information in the original
> image. Is it possible to overlay a colour map onto a grey-scale image
> without obliterating information in the grey-scale image ? ie: by making
> the colour overlay "transparent" to some degree ?
>
> Jason

Jason -

Here is how I go about accomplishing this. Basically I scale the entire raw image into one-half the range of values using my `BYTE_SCALE()` routine, and then I take the regions-of-interest that I want in `colscale` and scale those into the upper-half of the color range. Then I use my `GRAYSCALE()` routine which allows you to define half the range as a gray-scale, and the other half as a color-scale (red, green, blue, cyan, yellow, etc); it would be easy to generalize this to any color). `GRAYSCALE()` allows you to adjust the gray-/color-scales interactively. If you like, you can split the `colscale` into three ranges (eg. gray, cyan, yellow).

I wrote `BYTE_SCALE()` because I wanted to be able to specify a minimum value for scaling as well as maximum.

Here is some code that will give you the general idea:

```
data.maxim = max(image)

upper = BYTE_SCALE(image, bot=data.mincol, top=!d.table_size/2-1, $
    max=data.maxim) + BYTE(!d.table_size/2)

lower = BYTE_SCALE(image, bot=data.mincol, top=!d.table_size/2-1, $
    max=data.maxim)

upper(roi_indices) = lower(roi_indices)
```

The scaling is done this way so that the same parameters are used for both scalings, which is necessary to make sure that elements in

the two ranges are in a one-to-one correspondence (since we use this for thresholding). This is why the upper half is scaled to the lower half and then incremented, instead of just scaling to the upper half to begin with. Let me know if this doesn't make sense!

Here is how you might use GRAYSCALE():

```
split_color = [0,1,1]          ; Cyan
tvlct, 255,0,175, 0
zero = [255,0,175]            ; Preserve color index 0
bottom = state.gray_bottom
top = state.gray_top

grayscale, bottom, top, split_color=split_color, min=state.mincol, $
    zero=zero, parent=state.base, /modal, right=right

state.gray_bottom(state.common_volume) = bottom
state.gray_top(state.common_volume) = top
```

You can get BYTE_SCALE and GRAYSCALE and many other routines at:

<ftp://bial8.ucsd.edu/pub/software/idl/share>

Hope this helps.

Dave

--

```
~~~~~
David S. Foster      Univ. of California, San Diego
Programmer/Analyst  Brain Image Analysis Laboratory
foster@bial1.ucsd.edu  Department of Psychiatry
(619) 622-5892      8950 Via La Jolla Drive, Suite 2240
                    La Jolla, CA 92037
~~~~~
```
