
Subject: Re: Overlay multiple filled contour plots?

Posted by peter.albert@gmx.de on Wed, 09 Nov 2005 07:48:04 GMT

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

Hi Sarah,

if I get it right, after plotting the terrain height the ocean areas are still black (or white, or whatever your background colour is), which you'd like to change by plotting the landsea mask? Well, it might be that David Fanning's `image_blend` routine might help (http://www.dfanning.com/programs/image_blend.pro), but it is actually more about overlaying images using transparency. Your point seems to be not to overlay the terrain information but to cut out the ocean areas, put a different information there and leave the rest untouched.

I'd suggest creating one array first using all input data you have and the subsequent usage of the `WHERE` function, combined with a definition of an appropriate colour bar using `TVLCT`. You can find a crude elevation colour bar here (<http://tinyurl.com/9682g>), which uses index 0 as blue for the ocean, colours 1 to 80 for shades of green and brown and finally white. Using this your code could look like:

```
...
add the definiton of the colour bar here
...
...
You can similarly use the upper colours > 80 for your whatever else
you want to display
...

; Scale the topography between 0 and 3000 m,
; such that only the 80 colours of thr elevation colour bar are used:

plot_array = bytscl(topography, min = 0, max = 3000, top = 80)

; Mask ocean areas

idx = where(landsea eq 0, n)

; Setting the ocean pixels to 0 will make them appear blue.
if n gt 0 then plot_array[idx] = 0

; Now you want to ass "some other variable".
; I guess this variable does not fill the entire plot region,
; but is given e.g. in an array with the same dimension as the
; topography and the landsea mask. Let's assume it can also be
; identified by a threshold, in this example the values between 1 and
100
```

```
; should be displayed using the colours 81 to 90 (which you have to add  
to  
; the colour bar in advance)
```

```
idx = where(additional_data gt 0, n)  
if n gt 0 then plot_array[idx] = $  
    bytscl(additional_data[idx], min = 1, max = 100 top = 9) + 81
```

; And finally:

```
tv, plot_array
```

Cheers,

Peter
