Subject: update vert_colors of new graphics
Posted by Markus Schmassmann on Wed, 30 Aug 2017 15:36:28 GMT
View Forum Message <> Reply to Message

Hi everybody,

I have a series of images with an overploted line with changing color.
While I know how to update the underlying image and the coordinates of
the line I have problems updating the colors.
A single image is plotted using the "new" graphics with:

```
help, x,y,c,conti
% X       double     [0,256,1501]
% Y       double     [0,256,1501]
% C       byte       [0,256,1501]
% CONTI   float      [100,100,1501]
cwheel=byte( colortable( reform( [ !color.red,!color.orange, $
     !color.yellow,!color.green,!color.blue,!color.purple,!color. red ], $
    [3,7] ),/transpose ) )

i=800
i1=image(conti[*,*,i])
p1=plot( [x[0,*,i], y[0,*,i]], overplot=i2, $
         vert_colors=c[0,*,i], rgb_table=cwheel )

; and updated with
for i=0,1500 do begin
    i1.putData, conti[*,*,i]
    p1.putData, [x[0,*,i],y[0,*,i]]
;;; ----> UPDATE VERT_COLORS HERE<----- ;;;
endfor

; but I don't know how to update the VERT_COLORS
; a limited workaround is keeping VERT_COLORS constant
; but changing the RGB_TABLE as follows:

i=800
i2=image(conti[*,*,i])
p2=plot([x[0,*,i], y[0,*,i]],overplot=i2,vert_colors=bindgen(256), $
        rgb_table=cwheel[*,c[0,*,i]])

for i=0,1500 do begin
    i2.putData, conti[*,*,i]
    p2.putData, [x[0,*,i],y[0,*,i]]
    p2.rgb_table=cwheel[*,c[0,0:255,i]]
endfor

; but this works only if for every time step there are at most 256
```

; values to plot

does anybody know how to properly update VERT_COLORS ?

Any help would be appreciated, thanks, Markus

Subject: Re: update vert_colors of new graphics
Posted by Dick Jackson on Fri, 01 Sep 2017 14:33:05 GMT
View Forum Message <> Reply to Message

On Wednesday, 30 August 2017 08:36:32 UTC-7, Markus Schmassmann wrote:
> Hi everybody,
>
> I have a series of images with an overploted line with changing color.
> While I know how to update the underlying image and the coordinates of
> the line I have problems updating the colors.
> A single image is plotted using the "new" graphics with:
>
> help, x,y,c,conti
> % X double [0,256,1501]
> % Y double [0,256,1501]
> % C byte [0,256,1501]
> % CONTI float [100,100,1501]
> cwheel=byte(colortable(reform([!color.red,!color.orange, $
> !color.yellow,!color.green,!color.blue,!color.purple,!color. red], $
> [3,7]),/transpose))
>
> i=800
> i1=image(conti[*,*,i])
> p1=plot([x[0,*,i], y[0,*,i]], overplot=i2, $
> vert_colors=c[0,*,i], rgb_table=cwheel)
>
> ; and updated with
> for i=0,1500 do begin
> i1.putData, conti[*,*,i]
> p1.putData, [x[0,*,i],y[0,*,i]]
> ;;; ----> UPDATE VERT_COLORS HERE<----- ;;;
> endfor
>
> ; but I don't know how to update the VERT_COLORS
> ; a limited workaround is keeping VERT_COLORS constant
> ; but changing the RGB_TABLE as follows:
>
> i=800
> i2=image(conti[*,*,i])
> p2=plot([x[0,*,i], y[0,*,i]],overplot=i2,vert_colors=bindgen(256), $

```
>          rgb_table=cwheel[*,c[0,*,i]])
>
>  for i=0,1500 do begin
>       i2.putData, conti[*,*,i]
>       p2.putData, [x[0,*,i],y[0,*,i]]
>       p2.rgb_table=cwheel[*,c[0,0:255,i]]
>  endfor
>
>  ; but this works only if for every time step there are at most 256
>  ; values to plot
>
>
>  does anybody know how to properly update VERT_COLORS ?
>
>  Any help would be appreciated, thanks,                Markus
```

Hi Markus,

Using this version, I don't have any trouble adjusting VERT_COLORS or RGB_TABLE:

IDL Version 8.5.1, Microsoft Windows (Win32 x86_64 m64).

See if this works for you (put into a file, compile and run):

```
;    Set up variables
h = 100           ; height of test image
vc = bindgen(256)  ; indices of vert_colors
cwheel=byte( colortable( reform( [ !color.red,!color.orange, $
      !color.yellow,!color.green,!color.blue,!color.purple,!color. red ], $
     [3,7] ),/transpose ) )

;    Make background image
img =  image(bytscl(LIndgen(256,h)),dimensions=[256,h]*3,position=[ 0,0,1,1])

;    Make coloured squiggle across image
p =  plot(overplot=img,thick=3,randomu(seed,256)*h,vert_colors=vc , $
        rgb_table=cwheel)

;    Cycle colours around by shifting vert_colors
img.window_title='Cycle colours around by shifting vert_colors...'
wait, 2
for i=0b, 255b do p.vert_colors=shift(vc, i)

;    Dim and brighten colours by scaling rgb_table
img.window_title='Dim and brighten colours by scaling rgb_table...'
wait, 2
for i=-1.0,1.0,0.01 do p.rgb_table=byte(cwheel*abs(i))
```

end

Does that work for you?

(For anyone wondering about my spelling: I speak English with a Canadian accent, and I speak IDL with an American accent :-)

Cheers,
-Dick

Dick Jackson Software Consulting Inc.
Victoria, BC, Canada --- http://www.d-jackson.com
