Subject: color question
Posted by R.Bauer on Thu, 20 Apr 2000 07:00:00 GMT

View Forum Message <> Reply to Message

Hi David and all other.

I have true_color mode and I have already set device,decomposed=0.

If I start a widget sometimes the color table is destroyed and I have to start a new idl session to get my colors back.

Any ideas are welcome.

regards

R.Bauer

Subject: Re: Color question
Posted by davidf on Tue, 03 Apr 2001 21:18:37 GMT

View Forum Message <> Reply to Message

R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:

> I have a filled contour plot that I would like to
> display with a color bar. Everything looks fine
> on the screen but when I try to print, the colors
> on the bar are no longer correct. They appear
> washed out(dithered?). The contour colors, however,
> appear correct. If I look at the file with
> ghostscript, everything looks fine.
>
> I am generating the color bar with TV. It seems
> that the printer is treating a plot and an image
> differently.

Be sure you set BITS_PER_PIXEL to 8 on your
device command. There are only 4 bits per pixel
saved for image pixel by default. Enough for
only 16 colors.

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Color question
Posted by rkj on Tue, 03 Apr 2001 21:20:48 GMT

View Forum Message <> Reply to Message

David Fanning (davidf@dfanning.com) wrote:
: R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:
:
: > I have a filled contour plot that I would like to
: > display with a color bar. Everything looks fine
: > on the screen but when I try to print, the colors
: > on the bar are no longer correct. They appear
: > washed out(dithered?). The contour colors, however,
: > appear correct. If I look at the file with
: > ghostscript, everything looks fine.
: >
: > I am generating the color bar with TV. It seems
: > that the printer is treating a plot and an image
: > differently.

: Be sure you set BITS_PER_PIXEL to 8 on your
: device command. There are only 4 bits per pixel
: saved for image pixel by default. Enough for
: only 16 colors.

: Cheers,

: David
: --
: David Fanning, Ph.D.
: Fanning Software Consulting
: Phone: 970-221-0438 E-Mail: davidf@dfanning.com
: Coyote's Guide to IDL Programming: http://www.dfanning.com/
: Toll-Free IDL Book Orders: 1-888-461-0155

I have it set to 8 bits_per_pixels and I still get the same
result.

Kyle

Subject: Re: Color question
Posted by davidf on Tue, 03 Apr 2001 21:38:04 GMT

View Forum Message <> Reply to Message

R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:
>
> I have it set to 8 bits_per_pixels and I still get the same
> result.

Well, then, I've never heard of anything like it! :-)

I presume you are sure you are using exactly the same
color indices for both the contour colors and
the color bar colors? And that you have copied
the color table into your PostScript device?
If both of these are true, how could the colors
be different?

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Color question
Posted by Martin Schultz on Wed, 04 Apr 2001 07:46:17 GMT

View Forum Message <> Reply to Message

"R. Kyle Justice" wrote:
>
> I have a filled contour plot that I would like to
> display with a color bar. Everything looks fine
> on the screen but when I try to print, the colors
> on the bar are no longer correct. They appear
> washed out(dithered?). The contour colors, however,
> appear correct. If I look at the file with
> ghostscript, everything looks fine.
>
> I am generating the color bar with TV. It seems
> that the printer is treating a plot and an image
> differently.
>
> Kyle J.

If you have discrete colors in your contours and colorbar, you could
try out my colorbar (should be renamed to mgs_colorbar) routine which

you find at
http://www.mpimet.mpg.de/~schultz.martin/idl/html/libmartin_ schultz.html
If you supply this routine with a levels and c_colors keyword, it will
draw filled rectangles with the same c_colors (and levels) you use in
the contour command. If that doesn't show up correctly in your
printout then I have really no idea what's going on.

Cheers,

Martin

PS: Ooops. I just recognize there was a recent bugfix. So, I attach
the new version which is now called mgs_colorbar.pro

--
[[[[[[[[[
[[Dr. Martin Schultz   Max-Planck-Institut fuer Meteorologie   [[
[[Bundesstr. 55, 20146 Hamburg   [[
[[phone: +49 40 41173-308   [[
[[fax: +49 40 41173-298   [[
[[martin.schultz@dkrz.de   [[
[[[[[[[[[

```
;+
; NAME:
;       COLORBAR
;
; PURPOSE:
;       Draw a colorbar (legend) with labels
;
; CATEGORY:
;       Colors
;
; CALLING SEQUENCE:
;       COLORBAR [,options]
;
; INPUTS:
;
; KEYWORD PARAMETERS:
;       COLOR  -> the drawing color for boxes and labels
;            (default: !P.COLOR)
;
;       BOTTOM -> first color index to use (default: 1)
;
;       NCOLORS -> number of colors to use (default:
;            !D.N_Colors-bottom)
;
;       MIN, MAX -> range of the data (default bottom and
;            bottom+ncolors-1
```

```
;
;       LABEL -> array with label values (must be numeric).
;           Normally, it is easier to generate labels with the
;           DIVISIONS options, but this allows tighter control
;           (e.g. 1,2,5,10,20 labels on logarithmic scales).
;           Default (if no DIVISIONS are given): min and max
;
;       DIVISIONS -> number of labels to put on the colorbar.
;           Note that this keyword is overwritten by LABEL.
;           The labels will be equally spaced and the /LOG option
;           will be honored.
;
;       FORMAT -> output format of the labels. Default is determined
;           according to the range given in min and max. Label strings
;           will be trimmed, so you can safely specify f14.2 for example.
;
;       SCIENTIFIC -> If set, will call STRSCI to put the colorbar
;           labels in scientific notation format (e.g. in the form
;           A x 10^B).  STRSCI will use the format string specified
;           in the FORMAT keyword.
;
;       /LOG -> logarithmic spacing of labels (colors are *always*
;           linearily distributed)
;
;       C_COLORS -> array of color indices for "discrete" color bars
;           e.g. in filled contour plots. You must also use the
;           C_LEVELS keyword, otherwise there will most likely be
;           a mismatch between your plot colors and your colorbar
;           colors. COLORBAR normally limits the number of labels
;           it prints to 10. Use the SKIP keyword to force a different
;           behaviour. If C_COLORS is not undefined it overrides the
;           settings from NCOLORS, and BOTTOM.
;
;       C_LEVELS -> array with label values for discrete colorbars.
;           Use the LABEL keyword for continuous colors. C_LEVELS
;           must have the same number of elements as C_COLORS and
;           assigns one label to each color change (LABEL distributes
;           the labels evenly). Use the SKIP keyword to skip labels.
;           As default, COLORBAR limits the number of labels printed
;           to 10.
;           NB: In order to be consistent with the behaviour of the
;           CONTOUR procedure, colorbar disregards the first entry of
;           C_LEVELS.
;
;       SKIP -> print only every nth discrete label. Default is computed
;           so that COLORBAR will print no more than 10 labels.
;
;       STARTLABEL -> for discrete colorbars: the first left box side to be
;             labeled. Default is 1, i.e. between the first two boxes.
;
;         /VERTICAL -> set this keyword to produce a vertical colorbar
;             (default is horizontal). Note that out-of-range boxes are
;             only implemented for horizontal color bars.
;
;         POSITION -> a position value or 4-element vector. If POSITION
;             contains only one element, it will be centered at the
;             bottom or right side of the page and extend over 60% of
;             the total plotting area.
;
;         CHARSIZE -> character size (default !p.charsize)
;
;         TITLE -> a title string (similar to XTITLE or YTITLE for PLOT)
;
;         UNIT -> a unit string that will be added to the right (top) of
;             the labels
;
;         BotOutOfRange, TopOutOfRange -> a colorindex value for data
;             that falls below or above the normal plot range. If given,
;             an extra box will be drawn to the left or right of the color-
;             bar, and the colorbar will shrink in size. A default label
;             '<' ('>') will be placed below. Note that these options are
;             only available for horizontal colorbars.
;
;         BOR_Label, TOR_Label -> label values for BOTOutOfRange and
;             TopOutOfRange that replace the defaults.
;
; OUTPUTS:
;
; SUBROUTINES:
;
; REQUIREMENTS:
;         The user should have some knowledge about colortables and the
;         use of parts of a colortable.
;         This program uses STRSCI for labels in scientific notation.
;
; NOTES:
;         This routine was designed after David Fanning's colorbar
;         routine and enhanced. Some of the postscript
;         handling of DF was removed, positioning is a little easier but
;         maybe a little less flexible; out-of-range boxes have been
;         added.
;
; EXAMPLE:
;         ; draw a colorbar with all available colors on top of index 20
;         ; will be placed at the bottom of the page
;
```

```
;       colorbar,bottom=20,min=min(data),max=max(data)
;
;       ; draw another colorbar above the first one, use logarithmic scale
;
;        colorbar,bottom=20,min=0.1,max=10.,labels=[0.1,0.5,1.,5.10.] ,/log, $
;             position=0.3,unit='ppt'
;
;       ; (simply add keyword /vertical and you'll get it flipped)
;
;       ; colorbar with out-of-range information on right side only
;       ; Here we used 20 colors for the plot, the 21st is for
;       ; out-of-range data
;
;       colorbar,bottom=20,ncolors=20,min=0,max=100,divisions=5, $
;             TopOutOfRange=40
;
; MODIFICATION HISTORY:
;       mgs, 02 Jun 1998: VERSION 1.00
;       mgs, 14 Nov 1998: - changed default format to f14.2 from f6.2
;       mgs, 19 Nov 1998: - added cbdefaultformat function to better handle
;                           default labeling format.
;       mgs, 28 Nov 1998: - default labelling format now exponential for
;                           values gt 1.e6
;       mgs, 19 May 1999: - unit string placed a little further right
;                           in horizontal colorbars.
;       mgs, 27 May 1999: - added functionality for discrete colorbars
;                           (C_COLORS, C_LEVELS, and SKIP keywords)
;       bmy, 02 Jun 1999: - added /SCIENTIFIC keyword
;                         - updated comments
;       mgs, 14 Sep 1999: - charsize bug fix. Needed to check for 0.
;       mgs, 23 Sep 1999: - now uses !P.COLOR as default color (as it was originally)
;                         - BOTTOM default now 1 (because 0 is the standard beackground
;                           value)
;       mgs, 17 Dec 1999: - fixed zero division for SKIP=0
;                         - added StartLabel keyword for discrete colorbars
;       mgs, 20 Dec 1999: - disregard first entry of C_LEVELS
;       gjb, 16 Feb 2001: - fixed bug in coordinating levels and
;                           colors
;       mgs, 20 Feb 2001: - renamed to mgs_colorbar
;
;-
; Copyright (C) 1998, 1999, Martin Schultz and Bob Yantosca,
;  Harvard University
; This software is provided as is without any warranty
; whatsoever. It may be freely used, copied or distributed
; for non-commercial purposes. This copyright notice must be
; kept with any copy of this software. If this software shall
; be used commercially or sold as part of a larger package,
; please contact the author.
; Bugs and comments should be directed to mgs@io.harvard.edu
; or bmy@io.harvard.edu with subject "IDL routine colorbar"
 ;------------------------------------------------------------ --


function cbdefaultformat,minv,maxv,log=log

   ; return default format string depending on min and max value
   ; and log flag

   res = '(f14.2)'              ; general default

   ; determine necessary number of decimal places
   ndec = fix( 2.-alog10( (maxv-minv) > 1.0E-31 ) )
   ndecmin = fix( 2.-alog10( minv > 1.0E-31 ) )

   if (keyword_set(log)) then ndec = max([ndec,ndecmin-1])

   if (ndec gt 2) then res = '(e12.3)'
   if (ndec eq 3 AND log) then res = '(f14.3)'
   if (ndec le 0) then res = '(I14)'
   if (ndec le -6) then res = '(e12.3)'

   return,res

end



pro mgs_colorbar,color=color,bottom=bottom,ncolors=ncolors, $
      min=minv,max=maxv,label=label,divisions=divisions, $
       c_colors=c_colors,c_levels=c_levels,skip=skip,startlabel=sta rtlabel,  $
      format=format,log=log,    $
      vertical=vertical,position=position,charsize=charsize, $
      title=title,unit=unit, $
      BotOutOfRange=BotOutOfRange,TopOutOfRange=TopOutOfRange, $
       BOR_Label=BOR_Label,TOR_Label=TOR_Label,Scientific=Scientifi c


   ; Pass external functions
   FORWARD_FUNCTION StrSci

    ;=============================================================== ========
   ; set defaults and determine parameters
    ;=============================================================== ========
   IsDiscrete = n_elements(C_Colors) gt 0 ; discrete colorbar for
                                          ; filled contours?
   if (n_elements(color) eq 0) then color = !P.COLOR
   if (n_elements(bottom) eq 0) then bottom = 1
   MAXCOL = !D.N_COLORS<256
   if (bottom ge MAXCOL-1) then bottom = MAXCOL-1
   if (n_elements(ncolors) eq 0) then ncolors = MAXCOL-bottom
   if (ncolors gt MAXCOL-bottom) then ncolors = MAXCOL-bottom
   if (n_elements(minv) eq 0) then minv = bottom
   if (n_elements(maxv) eq 0) then maxv = ncolors
   log = keyword_set(log)
   if (log AND minv le 0.) then minv = 0.01

   if (n_elements(format) eq 0) then $
      format = cbdefaultformat(minv,maxv,log=log)

    ;========================================================== ========
   ; compute default labels
    ;========================================================== ========
   if (IsDiscrete AND n_elements(SKIP) eq 0) then $
      Skip = fix( (n_elements(C_Colors)-1)/10 ) + 1
   if (n_elements(Skip) eq 0) then Skip = 1
   if (n_elements(StartLabel) eq 0) then StartLabel = 0

   if (n_elements(divisions) eq 0) then divisions = 2
   if (IsDiscrete AND n_elements(C_Levels) eq 0) then $
      Divisions = fix( (n_elements(C_Colors)-1)/(Skip>1) ) + 1

; if (IsDiscrete) then begin
;    print,'#### SKIP,DIVISIONS=',skip,divisions
;    print,'C_Colors=',C_Colors
; endif

   if (n_elements(label) eq 0) then begin
      if (divisions gt 1) then begin
         if (log) then $
            label = 10^(findgen(divisions)/(divisions-1)*  $
                        (alog10(maxv/minv))+alog10(minv) ) $
         else $
            label = findgen(divisions)/(divisions-1)*(maxv-minv)+minv
      endif else $
         label = -1
   endif

   ; Overwrite standard labels with C_Levels if given
   if (IsDiscrete AND n_elements(C_Levels) gt 0) then begin
      myC_Levels = C_Levels[1:*]    ; omit first entry to be consistent with CONTOUR
      Ind = indgen( (n_elements(myC_Levels)-1)/(Skip>1) + 1) * Skip + StartLabel
      okInd = where(Ind ge 0 AND Ind lt n_elements(myC_Levels))
    if (okInd[0] lt 0) then begin
      message,'Invalid combination of Skip and StartLabel!',/Continue
      return
    endif
    Ind = Ind[okInd]
    Label = myC_Levels[Ind]
; print,'### C_Levels=',myC_Levels
; print,'### Ind = ',Ind,'   Label = ',Label
  endif

print,'# IsDiscrete = ',IsDiscrete
print,'#### LEVELS = ',myC_Levels
print,'#### LABELS = ',label

  if (n_elements(charsize) eq 0) then charsize = !p.charsize
  if (charsize eq 0) then charsize = 1.

  if (n_elements(title) eq 0) then title = ''
  if (n_elements(unit) eq 0) then unit = ''

  if (n_elements(BotOutOfRange) eq 0) then BotOutOfRange = -1
  if (n_elements(TopOutOfRange) eq 0) then TopOutOfRange = -1
  if (n_elements(BOR_Label) eq 0) then BOR_Label = '<'
  if (n_elements(TOR_Label) eq 0) then TOR_Label = '>'

  vertical = keyword_set(vertical)

   ;============================================================ ========
  ; keep simple: out-of-range boxes only allowed for horizontal bar
  ; Also, these do not make sense in discrete color bars
   ;============================================================ ========
  if (vertical OR IsDiscrete) then begin
    BotOutOfRange = -1
    TopOutOfRange = -1
  endif

   ;============================================================ ========
  ; position: if only one element then center bar according
  ; to vertical keyword and give it a width of 60%
   ;============================================================ ========
  if (n_elements(position) eq 0) then position = abs((vertical) - 0.10)
  if (n_elements(position) ne 4) then begin
    if (vertical) then $
      position = [ position[0], 0.2, position[0]+0.03, 0.8 ] $
    else $
      position = [ 0.2, position[0], 0.8, position[0]+0.03 ]
  endif
```

```
    ;================================================================= ========
   ; make space for extra boxes for out of range
    ;================================================================= ========
   barpos = position
   x10 = (position[2]-position[0])/10. > 0.03
   truecharsize = float(!D.Y_CH_SIZE*charsize)

   labelpos = ( position[1] * !D.Y_VSIZE - truecharsize*1.05 ) / $
               !D.Y_VSIZE

    ;================================================================= ========
   ; draw rectangles and fill them (out of range boxes)
    ;================================================================= ========
   if (BotOutOfRange ge 0) then begin
     rpos = [ position[0], position[1], position[0]+x10, position[3] ]
     RectAngle,rpos,px,py

;     px = [ position[0], position[0]+x10,position[0]+x10,  $
;           position[0], position[0] ]
;     py = [ position[1], position[1] ,position[3],  $
;           position[3], position[1] ]
     polyfill,px,py,/norm,color=BotOutOfRange,/fill
     plots,px,py,/norm,color=color,thick=!p.thick

     ; annotate
     xyouts,position[0]+x10/2.,labelpos,BOR_Label,/norm, $
        color=color,align=0.5,charsize=charsize

     ; shorten central bar
     barpos = [ barpos[0]+x10+0.01, barpos[1], $
                barpos[2], barpos[3] ]
    endif

    if (TopOutOfRange ge 0) then begin
      rpos = [ position[2], position[1], position[2]-x10, position[3] ]
      RectAngle,rpos,px,py

    ;   px = [ position[2], position[2]-x10,position[2]-x10,  $
    ;         position[2], position[2] ]
    ;   py = [ position[1], position[1] ,position[3],  $
    ;         position[3], position[1] ]
      polyfill,px,py,/norm,color=TopOutOfRange,/fill
      plots,px,py,/norm,color=color,thick=!p.thick

      ; annotate
      xyouts,position[2]-x10/2.,labelpos,TOR_Label,/norm, $
               color=color,align=0.5,charsize=charsize
      ; shorten central bar
      barpos = [ barpos[0], barpos[1], $
                barpos[2]-x10-0.01, barpos[3] ]
    endif

    ; reset bar position in case of error
    if (barpos[0] gt barpos[2]) then barpos = position

     ;================================================================= ========
    ; create (central) colorbar
     ;================================================================= ========
    xstart = barpos[0] * !D.X_VSIZE
    ystart = barpos[1] * !D.Y_VSIZE
    xsize = (barpos[2] - barpos[0]) * !D.X_VSIZE
    ysize = (barpos[3] - barpos[1]) * !D.Y_VSIZE

    if (IsDiscrete) then begin
       ;================================================================= =====
      ; discrete color bar: need to polyfill individual rectangles
      ; compute position for each rectangle
       ;================================================================= =====
      if (vertical) then begin
        dx = 0.
        dy = ysize/n_elements(C_Colors)
      endif else begin
        dx = xsize/n_elements(C_Colors)
        dy = 0.
      endelse
      for i=0,n_elements(C_Colors)-1 do begin
        if (vertical) then $
          box = [ xstart, ystart+i*dy, xstart+xsize, ystart+(i+1)*dy ] $
        else  $
          box = [ xstart+i*dx, ystart, xstart+(i+1)*dx, ystart+ysize ]

; print,'### box=',box
; print,'### xstart,xsize,ystart,ysize=',xstart,xsize,ystart,ysize
        RectAngle,box,px,py
        PolyFill,px,py,/DEVICE,color=C_Colors[i]
      endfor

    endif else begin
       ;================================================================= =====
      ; continuous colorbar: use TV to display a smooth range of colors
       ;================================================================= =====
      bcol = bindgen(ncolors) + bottom
      if (vertical) then $
        bar = replicate(1B,5) # bcol  $
      else $
```

```
         bar = bcol # replicate(1B,5)

      IF (!D.Name eq 'PS') THEN BEGIN
         TV, bar, xstart, ystart, XSIZE=xsize, YSIZE=ysize
      ENDIF ELSE BEGIN
         bar = CONGRID(bar, CEIL(xsize), CEIL(ysize), /INTERP)
         TV, bar, xstart, ystart
      ENDELSE
   endelse

   ; Draw frame around colorbar
;  px = [ barpos[0], barpos[0], barpos[2], barpos[2], barpos[0] ]
;  py = [ barpos[1], barpos[3], barpos[3], barpos[1], barpos[1] ]
   RectAngle,barpos,px,py
   plots,px,py,/norm,color=color,thick=!p.thick

    ;============================================================= ========
   ; labelling : set up plot coordinates with x or y range eq to
   ; device size, then use position parameters for unit and title
    ;============================================================= ========
   if (n_elements(label) lt 2) then return

    ;============================================================= ========
   ; Convert LABEL to string representation SLABEL here
   ; If /SCIENTIFIC is set, then call STRSCI to put the labels into
   ; the form A x 10^B.
    ;============================================================= ========
   if ( Keyword_Set( Scientific ) ) then begin
      SLabel = StrSci( Label, /Trim, Format=Format, _EXTRA=e )
   endif else begin
      SLabel = StrTrim( String( Label, Format=Format ), 2 )
   endelse

    ;============================================================= ========
   ; Vertical colorbar
    ;============================================================= ========
   if (vertical) then begin
       if (IsDiscrete) then begin
          yrange = [0,n_elements(c_colors)]    ; one more!
          ;; bug fix gjb
          ypos = 1+findgen(n_elements(Label))*Skip+StartLabel
          npos = n_elements(label)+1
       endif else begin
          yrange = [minv,maxv]
          ypos = Label
          npos = n_elements(label)
       endelse
```

```
        plot,[0],[0],/nodata,ylog=log,xstyle=5,ystyle=5, $
          xrange=[-(barpos[2]-barpos[0])*!D.X_VSIZE, 0.], $
          yrange=yrange, $
          position=barpos,/NOERASE

        nypos = convert_coord(replicate(0,npos),ypos,/DATA,/TO_NORMAL)
print,ypos
help,slabel & print,slabel
    ;    xyouts,replicate(truecharsize*1.05,n_elements(label)),label, $
        xyouts,replicate(truecharsize*1.05,npos),ypos, $
           SLabel,/data,color=color,charsize=charsize,align=0.5,orient= 90

        plot,[0],[0],/nodata,xstyle=5,ystyle=5, $
          xrange=[-(barpos[2]-barpos[0])*!D.X_VSIZE, 0.], $
          yrange=[0,1], position=position,/NOERASE

        xyouts,truecharsize*1.05,1.1,unit,/data, $
          color=color,charsize=charsize,align=0,orient=90
        xyouts,truecharsize*2.15,0.5,title,/data, $
          color=color,charsize=charsize,align=0.5,orient=90

     ;============================================================ ========
    ; Horizontal colorbar
     ;============================================================ ========
    endif else begin

; print,'### minv,maxv = ',minv,maxv
        if (IsDiscrete) then begin
          xrange = [0,n_elements(c_colors)]    ; one more!
          ;; bug fix gjb
          xpos = 1+findgen(n_elements(Label))*Skip+StartLabel
          npos = n_elements(label)+1
        endif else begin
          xrange = [minv,maxv]
          xpos = Label
          npos = n_elements(label)
        endelse
; print,'### xpos, label=',xpos,label
; if (n_elements(myc_levels) gt 0) then print,'#### c_levels=',myc_levels
; if (n_elements(c_colors) gt 0) then print,'#### c_colors=',c_colors

        plot,[0],[0],/nodata,xlog=(log AND not IsDiscrete),  $
          xstyle=5,ystyle=5, $
          xrange=xrange, $
          yrange=[0, (barpos[3]-barpos[1])*!D.Y_VSIZE], $
          position=barpos,/NOERASE

; print,'#### barpos=',barpos
```

```
      nxpos = convert_coord(xpos,replicate(0,npos),/DATA,/TO_NORMAL)
print,labelpos,truecharsize
   ;  xyouts,nxpos[0,*],replicate(labelpos-truecharsize*1.05, npos), $
      xyouts,xpos,replicate(-truecharsize*1.05, npos), $
        slabel, /DATA,color=color,charsize=charsize,align=0.5

      plot,[0],[0],/nodata,xstyle=5,ystyle=5, $
        xrange=[0,1], $
        yrange=[0., (barpos[3]-barpos[1])*!D.Y_VSIZE], $
        position=position,/NOERASE

      xyouts,1.15,-truecharsize*1.05,unit,/data, $
        color=color,charsize=charsize,align=0
      xyouts,0.5,-truecharsize*2.15,title,/data, $
        color=color,charsize=charsize,align=0.5

   endelse


   return
end
```

File Attachments

1) mgs_colorbar.pro, downloaded 166 times

Subject: Re: Color question
Posted by rkj on Wed, 04 Apr 2001 16:24:51 GMT
View Forum Message <> Reply to Message

David Fanning (davidf@dfanning.com) wrote:
: R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:
: >
: > I have it set to 8 bits_per_pixels and I still get the same
: > result.

: Well, then, I've never heard of anything like it! :-)

: I presume you are sure you are using exactly the same
: color indices for both the contour colors and
: the color bar colors? And that you have copied
: the color table into your PostScript device?
: If both of these are true, how could the colors
: be different?

: Cheers,

: David

: --
: David Fanning, Ph.D.
: Fanning Software Consulting
: Phone: 970-221-0438 E-Mail: davidf@dfanning.com
: Coyote's Guide to IDL Programming: http://www.dfanning.com/
: Toll-Free IDL Book Orders: 1-888-461-0155

How indeed? I am using the same colors and I have copied
the table into the postscript device.

Martin's solution is to generate a discrete color bar using
POLYFILL. I am sure this will work but I would still like
to know why there is a difference between a color bar created
with TV and one with a plot routine (POLYFILL, CONTOUR, PLOTS).
Whether the bar is made up of continuous colors or a discrete
number of colors should not make a difference, should it?

I don't know enough about postscript to know what is going
on inside the file.

Kyle

Subject: Re: Color question
Posted by davidf on Wed, 04 Apr 2001 16:47:02 GMT
View Forum Message <> Reply to Message

R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:

> How indeed? I am using the same colors and I have copied
> the table into the postscript device.
>
> Martin's solution is to generate a discrete color bar using
> POLYFILL. I am sure this will work but I would still like
> to know why there is a difference between a color bar created
> with TV and one with a plot routine (POLYFILL, CONTOUR, PLOTS).
> Whether the bar is made up of continuous colors or a discrete
> number of colors should not make a difference, should it?
>
> I don't know enough about postscript to know what is going
> on inside the file.

After thinking about this a while longer, I wonder (you
have posted no code, alas) whether you are *specifically*
indicating the color indices for both the contour plot
and the colorbar or letting IDL (Wave?) choose
them for you. There can certainly be problems if you

don't choose color indices specifically. (Although not
the ones you describe, I shouldn't imagine.)

I have a FSC_Contour plot program that is similar
to MPI_PLOT that allows you to modify contour plots
interactively. It is *almost* ready for public
consumption. I'm quite sure the filled contours in
this match the colors of the attached colorbar. :-)

Let me know if you are interested, and I'll see what
I can do to get it ready to go. (IDL only, of course,
since it uses a GUI interface.)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Color question
Posted by Liam E. Gumley on Wed, 04 Apr 2001 17:02:06 GMT

View Forum Message <> Reply to Message

"R. Kyle Justice" wrote:

> How indeed? I am using the same colors and I have copied
> the table into the postscript device.
>
> Martin's solution is to generate a discrete color bar using
> POLYFILL. I am sure this will work but I would still like
> to know why there is a difference between a color bar created
> with TV and one with a plot routine (POLYFILL, CONTOUR, PLOTS).
> Whether the bar is made up of continuous colors or a discrete
> number of colors should not make a difference, should it?
>
> I don't know enough about postscript to know what is going
> on inside the file.

Could you post the smallest code example which reproduces the problem?

Cheers,
Liam.

Subject: Re: Color question

Posted by rkj on Wed, 04 Apr 2001 17:15:15 GMT

View Forum Message <> Reply to Message

Ok, here is some sample code that will hopefully demonstrate problem of grainy colors using TV on a postscript device. You will need a color postscript printer to verify this.

```
SET_PLOT, 'PS'
DEVICE, Bits_Per_Pixel=8, /Color
TEK_COLOR     ; or create your own discrete colormap
bar=BYTARR(200,1000)
FOR i=0,999 DO bar(*,i)=i*9/1000
TV, bar, /Device
PLOT, INDGEN(10), Color=7, Charthick=3, Thick=5, /Noerase
DEVICE, /Close
```

Print it out and you should see that the plot is a "pure" yellow but the bar yellow is "grainy."

I guess there is a difference between a plot and an image on a postscript device. Plots seem to use filled polygons (which give pure colors) while images appear to use pixels or dots.

Kyle

Subject: Re: Color question

Posted by davidf on Wed, 04 Apr 2001 17:41:12 GMT

View Forum Message <> Reply to Message

R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:

> I guess there is a difference between a plot and an image
> on a postscript device. Plots seem to use filled polygons
> (which give pure colors) while images appear to use pixels
> or dots.

Possibly. But if so, I would expect this to be a configurable item on the printer setup. I find it hard to believe it is a PostScript thing. How else would "scalable pixels" be implemented except as polygons?

But we are clearly out of my depth with regard to exactly how PostScript printers work. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Color question
Posted by John-David T. Smith on Wed, 04 Apr 2001 18:21:07 GMT

View Forum Message <> Reply to Message

David Fanning wrote:

```
>
>  R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:
>
>>  I guess there is a difference between a plot and an image
>>  on a postscript device.  Plots seem to use filled polygons
>>  (which give pure colors) while images appear to use pixels
>>  or dots.
<SNIP>
>>  SET_PLOT, 'PS'
>>  DEVICE, Bits_Per_Pixel=8, /Color
>>  TEK_COLOR     ; or create your own discrete colormap
>>  bar=BYTARR(200,1000)
>>  FOR i=0,999 DO bar(*,i)=i*9/1000
>>  TV, bar, /Device
>>  PLOT, INDGEN(10), Color=7, Charthick=3, Thick=5, /Noerase
>>  DEVICE, /Close
```

The immediate solution is to use *only* those colors as appear in countours in the color bar.

And no for loops are needed. You could write:

```
bar=LINDGEN(200,1000)/(200*100)
```

or dispense with the chicanery altogether and simply use:

```
bar=1b#bindgen(10)
TV, bar, 0,0,XSIZE=.5,YSIZE=4,/inches
```

Using the /inches (or /centimeters) keyword to TV for postscript output
is highly advised.

I get the same pure yellow either way: #ffff00

JD

Subject: Re: Color question
Posted by rkj on Wed, 04 Apr 2001 18:47:59 GMT

View Forum Message <> Reply to Message

JD Smith (jdsmith@astro.cornell.edu) wrote:
: David Fanning wrote:
: >
: > R. Kyle Justice (rkj@dukebar.crml.uab.edu) writes:
: >
: > > I guess there is a difference between a plot and an image
: > > on a postscript device. Plots seem to use filled polygons
: > > (which give pure colors) while images appear to use pixels
: > > or dots.
: <SNIP>
: > > SET_PLOT, 'PS'
: > > DEVICE, Bits_Per_Pixel=8, /Color
: > > TEK_COLOR   ; or create your own discrete colormap
: > > bar=BYTARR(200,1000)
: > > FOR i=0,999 DO bar(*,i)=i*9/1000
: > > TV, bar, /Device
: > > PLOT, INDGEN(10), Color=7, Charthick=3, Thick=5, /Noerase
: > > DEVICE, /Close

: The immediate solution is to use *only* those colors as appear in
: countours in the color bar.

: And no for loops are needed. You could write:

: bar=LINDGEN(200,1000)/(200*100)

: or dispense with the chicanery altogether and simply use:

: bar=1b#bindgen(10)
: TV, bar, 0,0,XSIZE=.5,YSIZE=4,/inches

: Using the /inches (or /centimeters) keyword to TV for postscript output
: is highly advised.

: I get the same pure yellow either way: #ffff00

: JD

I just tried the above code and noticed something very interesting.
If I use bar=1b#bindgen(10), then the plot colors come out correct
and pure. However, if I use bar=LINDGEN(200,1000)/(200*100), I get
the same problem I had before. But at least there is no FOR loop :-P

This is may be a printer specific thing. I don't know. But to be
on the safe side, a one-column array should be used. Plus using the
# operator looks better . . .

But many thanks to JD for solving the problem.

Kyle

Subject: Re: Color Question
Posted by Karsten Rodenacker on Fri, 22 Nov 2002 09:49:24 GMT
View Forum Message <> Reply to Message

Humm..., seemingly the send button starts brain activity.

A solution is device,true=24 in the startupfile and than fuction seems
similar to Gatish truecolor display.

Sorry

Rodenacker wrote:

>
> Hi, all
>
> There was in April 2000 a long discussion concerning idl display an linux,
> but not helpful for my situation.
>
> I am in the maybe unhappy situation to find under linux (suse 8.1, XFree86
> version: 4.2.0 and idl 5.5) a well functioning object graphics but bad
> direct graphics (channels cannot be loaded directly, b/w is by no means
> b/w). I have actually DirectColor 24bpp.
>
> Is there perhaps any knowledge not expressed in the newslist what to do?
>
> Regards

--
 Karsten Rodenacker ()

-- -------------:-)
GSF - Forschungszentrum   Institute of Biomathematics and Biometry
D-85758 Oberschleissheim   Postfach 11 29
Tel: +49 89 31873401 | FAX: ...3369 | rodena@gsf.de | Karsten@Rodenacker.de
http://www.gsf.de/ibb/homepages/rodenacker

Subject: Re: Color Question
Posted by Rodenacker on Fri, 22 Nov 2002 09:53:20 GMT

View Forum Message <> Reply to Message

Hum..., brain activity is seemingly started by the send button.

A solution is device,true=24 in the startup file.

Sorry

--
Karsten Rodenacker ()
-- -------------:-)
GSF - Forschungszentrum   Institute of Biomathematics and Biometry
D-85758 Oberschleissheim   Postfach 11 29
Tel: +49 89 31873401 | FAX: ...3369 | rodena@gsf.de | Karsten@Rodenacker.de
http://www.gsf.de/ibb/homepages/rodenacker

Subject: Re: Color Question
Posted by savoie on Fri, 22 Nov 2002 15:39:25 GMT

View Forum Message <> Reply to Message

Rodenacker <rodena@gsf.de> writes:

> Humm..., seemingly the send button starts brain activity.

Always the case, isn't it?

> A solution is device,true=24 in the startupfile and than fuction seems
> similar to Gatish truecolor display.

From Liam Gumley's fantastic book

```
; The following configuration was lifted from
; Liam Gumley's book, "Practical IDL Programming" p. 204 for color management

device, true=24
```

```
window, /free, /pixmap, colors=-10
wdelete,  !d.window
device, retain=2, decomposed=0, set_character_size=[10,12]
device,  get_visual_depth = depth
print,  'display depth: ',  depth
print,  'color table size: ',  !d.table_size
;print, 'must buy Liam's Book'
;print, ' http://www.amazon.com/exec/obidos/tg/detail/-/1558607005/qid
=1037979515/sr=8-1/ref=sr_8_1/102-2113620-1610518?v=glance&a mp;s=books'
```

Putting this in your startup.pro file may help you. Also, there is a linux IDL5.5 color bug. If you are seeing striping in your windows, you probably need to go to RSI's homepage and download the patch.

Hope this helps you.

Matt Savoie
National Snow and Ice Data Center