
Subject: Re: Crosshair data point extraction - Query

Posted by [oet](#) on Thu, 02 Dec 1993 15:58:09 GMT

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

In article AA19871@tigermoth.aero.gla.ac.uk.gla.ac.uk, gnaa38@aero.gla.ac.uk (Paul Porcelli) writes:

> Is there any facility in IDL or Wave to produce a plot and then identify the
> coordinates of a point by clicking the mouse or moving a crosshair over the
> point.

The procedure below plots a moving crosshair, example:

```
IDL> window
```

```
IDL> bighair, COLOR=!d.n_colors-1
```

-Thomas

***** CUT HERE *****

```
PRO BIGHAIR, x, y, KEYBOARD=KEYBOARD, ACCELERATE=ACCELERATE, $  
      PERMANENT=PERMANENT, COLOR=COLOR
```

```
;  
;  
;+  
; NAME:  
; BIGHAIR  
;  
; PURPOSE:  
;  
; Gets cursor location in current window,  
; using cursor lines across the entire window.  
; /Keyboard accepts arrow keys input from keyboard, rather than mouse  
; Accelerate = number of pixels to jump in "/keyboard" mode.  
; The X and Y cursor locations (in pixels) are returned.  
; Further modification could allow Normalized or  
; Data coordinates to be returned.  
;  
;  
;  
; CATEGORY:  
; Interactive Drawing  
;  
;  
; CALLING SEQUENCE:  
;  
; IDL> BIGHAIR, x, y [, /Keyboard [, Accelerate=# pixels]]  
;  
; KEYWORDS:
```

```

; KEYBOARD Use keyboard instead of mouse
; ACCELERATE increase steps on usage with keyboard
; PERMANENT show current position permanently
; COLOR color to draw bighair
;
; SEE ALSO:
;
;
; MODIFICATION HISTORY:
; Written by: Mike Mayer, 8/29/92
;           Jim Pendleton, Dept. Physics & Astronomy
;           Northwestern U.
;           Added to idlmeteo 11-june-1993 oet@sma.ch
;           28-june-1993, oet@sma.ch, inserted short
;           wait (0.2) after first plot, to get
;           more stability on first plot.
;           9-Aug-1993, oet@sma.ch, added keywords
; PERMANENT and COLOR
;-

```

```

if keyword_set(COLOR) then col=color else col=!d.n_colors-1

```

```

DEVICE, Get_Graphics = oldmode
DEVICE, Set_Graphics = 6 ; XOR write mode.
xmax = !D.X_Vsize
ymax = !D.Y_Vsize
!Err = 0 ; Reset.
oldx = 0
oldy = 0
If (not Keyword_Set(Accelerate)) then Begin
  Accelerate = 1.
EndIf
If (not Keyword_Set(KeyBoard)) then Begin
  PRINT, 'Press left mouse button to quit...'
  CURSOR, oldx, oldy, /Device, /Change
EndIf Else Begin
  PRINT, 'Press space bar to quit...'
  oldx = xmax/2
  oldy = ymax/2
  x = oldx
  y = oldy
EndElse
PLOTS, [oldx, oldx], [0, ymax], /Device, COLOR=col ; Draw first crosshairs.
PLOTS, [0, xmax], [oldy, oldy], /Device, COLOR=col ; Draw first crosshairs.
wait, 0.3
EMPTY
Up = String(27B) + '[A'
Down = String(27B) + '[B'

```

```

Right = String(27B) + '[C'
Left = String(27B) + '[D'
Clear = Get_Kbrd(0)
Null = "
WHILE !Err NE 1 DO BEGIN
  If (Keyword_Set(Keyboard)) then Begin
    A = String(Byte(Get_Kbrd(0)))
    While ((A ne String(27B)) and (StrUpCase(A) ne ' ')) Do Begin
      A = String(Byte(Get_Kbrd(0)))
    EndWhile
    B = String(Byte(Get_Kbrd(0)))
    C = String(Byte(Get_Kbrd(0)))
    Key = A + B + C
  EndIf Else Begin
    CURSOR, x, y, /Device, /Change
  EndElse
  PLOTS, [oldx, oldx], [0, ymax], /Device, COLOR=col ; Erase last crosshairs.
  PLOTS, [0, xmax], [oldy, oldy], /Device, COLOR=col ; Erase last crosshairs.
  If (Keyword_Set(Keyboard)) then Begin
    !Err = 0
    Case Key Of
      Up : y = oldy + Accelerate
      Down : y = oldy - Accelerate
      Left : x = oldx - Accelerate
      Right : x = oldx + Accelerate
      Null : x = x
    Else : !Err = 1
    EndCase
  EndIf
  If ((x ne oldx) or (y ne oldy)) then Begin
    if keyword_set(PERMANENT) then begin
      coords=convert_coord(x,y,/DEVICE, /TO_DATA)
      tmp_string= 'x: ' + strtrim(coords(0),2) + $
        ' y: ' + strtrim(coords(1),2)
      print, tmp_string
    endif

    PLOTS, [x, x], [0, ymax], /Device, COLOR=col ; Draw new crosshairs.
    PLOTS, [0, xmax], [y, y], /Device, COLOR=col ; Draw new crosshairs.
    oldx = x
    oldy = y
    EMPTY
  EndIf
ENDWHILE
If (not Keyword_Set(Keyboard)) then Begin
;
; Already erased in Keyboard mode.

```

```
;
PLOTS, [oldx, oldx], [0, ymax], /Device, COLOR=col ; Erase last crosshairs.
PLOTS, [0, xmax], [oldy, oldy], /Device, COLOR=col ; Erase last crosshairs.
EndIf
Empty
DEVICE, Set_Graphics = oldmode
PRINT, 'X = ' + STRTRIM(x, 2), ' Y = ' + STRTRIM(y, 2)
END
```
