Subject: Re: XOR graphics
Posted by Liam Gumley on Wed, 15 Dec 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Ian Dean wrote:
> I am using XOR to plot a moving object. It works OK on a terminal using
> TRUECOLOR but screws up the colours on a terminal using PSEUDOCOLOR.
[IDL example removed]

Ian,

I think you'll find that using pixmaps for animation is more portable
than using XOR graphics. The following example program animates a moving
object on a plot fairly efficiently. The secret is to only restore the
area immediately surrounding the object itself. In your code, you were
restoring the entire window on every iteration. Here's my timing (PC
logged in to remote Unix box running IDL 5.1):

```
IDL> animate
Elapsed time (sec):       5.4369590
IDL> animate, delay=0.0
Elapsed time (sec):       1.5003740
```

Cheers,
Liam.

```
;---cut here---
PRO ANIMATE, DELAY=DELAY

;- Create a plot with an animated object
; $Id: animate.pro,v 1.2 1999/12/15 20:51:20 gumley Exp $

;- Check delay time (sec)
if n_elements(delay) eq 0 then delay = 0.02

;- Create graphics windows
xsize = 640
ysize = 512
window, /free, xsize=xsize, ysize=ysize
win1 = !d.window
window, /free, xsize=xsize, ysize=ysize, /pixmap
win2 = !d.window

;- Create the plot in the visible window
wset, win1
n = 200
x = findgen(n) * 0.1
y = sin(x)
```

```
plot, x, y, /nodata

;- Copy the plot to the pixmap window
wset, win2
device, copy=[0, 0, xsize, ysize, 0, 0, win1]

;- Save start time
t0 = systime(1.0)

;- Loop over all data points
xloc = -1
yloc = -1
for index = 1, n_elements(x) - 1 do begin

  ;- Restore current location from pixmap window
  if (xloc gt 0 ) and (yloc gt 0) then begin
    wset, win1
    device, copy=[xloc - 10, yloc - 10, 20, 20, $
      xloc - 10, yloc - 10, win2]
  endif

  ;- Plot the track in the pixmap window
  wset, win2
  plots, [x[index - 1], x[index]], [y[index - 1], y[index]]

  ;- Plot the track and object in the visible window
  wset, win1
  plots, [x[index - 1], x[index]], [y[index - 1], y[index]]
  plots, x[index], y[index], psym=5, symsize=2

  ;- Save the object location in device coordinates
  result = convert_coord(x[index], y[index], /data, /to_device)
  xloc = result[0]
  yloc = result[1]

  ;- Wait for delay time
  wait, delay

endfor

;- Print elapsed time
t1 = systime(1.0)
print, 'Elapsed time (sec): ', t1 - t0

END
;---cut here---

--
```

Liam E. Gumley
Space Science and Engineering Center, UW-Madison
http://cimss.ssec.wisc.edu/~gumley