
Subject: Re: Drawing satellite pixels on maps?

Posted by [Andy Loughe](#) on Tue, 19 Nov 1996 08:00:00 GMT

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

Philippe Peeters wrote:

>
> I've already posted this question some time ago but did not get any
> answer. Here I go again:
>
> I want to draw satellite data on a map. Each ground pixel is defined by
> the latitude and longitude of the 4 corners. It is not a regular
> rectangle or square and depend on the viewing geometry of the satellite
> instrument.
> I have tried a simple polyfill, long, lat but I have serious problems with
> orthographic maps when the pixel is partly off the map. I got strange
> filled polygons from the edge of the map to the corner of the window.
> Someone on the net advice me to use a new polyfill routine which checks
> polygons boundaries before drawing it. Though slower than the original
> polyfill, it solved the problem.

This is NOT a perfect solution, but it may give you some
ideas on how to proceed...

```
;
; Name ..... my_polyfill.pro
;
; Description ..... Before executing polyfill, check that
;                   the (x,y) coordinates are valid.
;
; Parameters ..... x,y contain vertices of the polygon.
;
; Keywords ..... See documentation for polyfill.
;
; Originator ..... Andrew F. Loughe, NOAA/ERL/CDC, 21 MAY 1996.
```

```
pro my_polyfill, x, y, _EXTRA=e
```

```
on_error, 2
```

```
; Convert the x,y coordinates to the normalized system.
result = CONVERT_COORD(x, y, /data, /to_normal)      ; DEFAULT:
From /data
```

```
; Check that /normal or /device might be set.
if ( KEYWORD_SET(e) ) then begin
    names = TAG_NAMES(e)
    for i = 0, N_TAGS(e)-1 do begin
```

```

    if ( names(i) eq 'NORMAL' ) then $
        result = CONVERT_COORD(x, y, /normal, /to_normal) ; From
/normalized
    if ( names(i) eq 'DEVICE' ) then $
        result = CONVERT_COORD(x, y, /device, /to_normal) ; From
/device
    endfor ; i-loop
endif ; KEYWORD_SET(e)

; Check that the normalized coordinates are within the bounds
; set by !x.window and !y.window
out_of_bounds = 0
if ( MIN(result(0,*)) lt !x.window(0) or $
    MAX(result(0,*)) gt !x.window(1) or $
    MIN(result(1,*)) lt !y.window(0) or $
    MAX(result(1,*)) gt !y.window(1) ) then out_of_bounds = 1

if (out_of_bounds eq 1) then print, '(x,y) coordinates are out of
bounds!'
if (out_of_bounds eq 0) then polyfill, x, y, _EXTRA=e

end

```

> But I still have another problem with several maps when the pixel to be
 > drawn is at the edge, i.e. when one or several pixel corner is on one
 > side of the map (lon > -180) and the other on the other side (lon < -180)
 > of the map.
 > example longitude=[-179,-181,-179.5,-180.5] or [179,181,179.5,180.5]

What if you add 360 to your points vector and
 make your map go from 180 to 540?

map_set, 0, 360, /cont

Does that work? |

--

Andrew F. Loughe afl@cdc.noaa.gov
 University of Colorado, CIRES http://cdc.noaa.gov/~afl
 Campus Box 449 phn:(303)492-0707 fax:(303)497-7013
 Boulder, CO 80309-0449 "He who laughs last thinks slowest!"
