Subject: IDLgrWindow zoomIn bug?
Posted by Erik[1] on Mon, 07 Apr 2008 11:03:06 GMT
View Forum Message <> Reply to Message

Hi Folks,

I have a problem using the ZoomIn function from the IDLgrWindow object in IDL 6.3. According to the IDL help file I expected the following to happen:

The IDLgrWindow::ZoomIn procedure method causes the current zoom factor for this window to be increased (that is, multiplied by the factor given by the window's ZOOM_BASE property). The current zoom factor, the virtual canvas dimensions, and the location of the visible portion of the window are updated to reflect the new zoom factor.

I already wrote some code to do the job for me, but what it doesn't do is update the Roi's in the zoomed area. I thought this function could help me out, but unfortunately I got some strange results. I use the following code to zoom at a mouseclick (self.View is added to self.Window);

```
self.Window->setProperty, ZOOM_BASE=2.0
self.Window->zoomIn
self.Window->Draw, self.View
self.Window->getProperty, Current_Zoom=cz, ZOOM_NSTEP=nstep
print, nstep
print, cz
```

What happens is that nstep and cz are updated as expected. On the first zoom, the coordinates of the drawwidget suddenly changes to 0,0 so the drawwidget moves on my widget to the upper left corner... When I click once more, the drawwidget jumps back to it's original position again, and luckily stays there. Very strange... But even worse; there is absolutely no zoom effect to be found in the drawwidget!

Given the fact this function is introduced in IDL 6.1 I expected it to be functional in 6.3 (or is this too optimistical? ;-)). Anyway I would love to hear that this is my fault and not a bug, because this function will be very useful!

Regards,
Erik

Subject: Re: IDLgrWindow zoomIn bug?
Posted by Erik[1] on Wed, 09 Apr 2008 10:29:51 GMT
View Forum Message <> Reply to Message

On 8 apr, 17:47, Rick Towler <rick.tow...@nomail.noaa.gov> wrote:
> Erik wrote:
>> On 8 apr, 06:02, David Fanning wrote:
>>> I've never used the ZoomIn/ZoomOut features before. But
>>> I just put it into a program that was zooming in a different
>>> way. For me, when I zoom in, the draw widget gets twice its
>>> current size!! Zoom out, it shrinks down again.
>
>>> I don't know about you, but that is not what I expected
>>> at all. (IDL 6.4 on Windows). I was sort of hoping the
>>> view would zoom in and out. Why would you want the draw
>>> widget changing size!?
>> Wow that isn't what I expected either and I was hoping the same thing
>> as you! I can't check it in 6.4, but in 6.3 it didn't work that way.
>> Well, it seems that I have to continue writing my own zoom code and
>> that it wasn't a waste of time writing it. Only some minor issues I
>> have to face (like re-drawing the ROI's in the same proportion as the
>> zoomed image) ;-)
>
> Erik, How are you zooming? I never use ROI's, but I would think that
> if you are zooming by changing your viewplane rectangle that the ROI's
> would "zoom" too. Here's the zoom method from my camera code:
>
> pro Camera::Zoom, zoom
> ; Zoom the camera view by the specified zoom factor.
>
> compile_opt idl2
>
> if (N_ELEMENTS(zoom) eq 1) then begin
> case 1 of
> (zoom lt -1.0) : self.zoom = (-1.0D / zoom)
> (zoom gt 1.0) : self.zoom = zoom
> else : self.zoom = 1.0D
> endcase
>
> viewplaneRect = dblarr(4, /NOZERO)
> viewplaneRect[0:1] = self.viewcoord[0:1] - $
> (self.viewRect[2:3] / (2.0D * self.zoom))
> viewplaneRect[2:3] = self.viewRect[2:3] / self.zoom
>
> self -> IDLgrView::SetProperty, VIEWPLANE_RECT= viewplaneRect
> endif
>
> end
>

> Set up the vars that store the initial state once, after you have set up
> your initial view:
>
> self -> IDLgrView::GetProperty, VIEWPLANE_RECT=viewplaneRect
> self.viewRect = viewplaneRect
> self.viewcoord[0] = ((2. * self.viewRect[0]) + $
> self.viewRect[2]) / 2.
> self.viewcoord[1] = ((2. * self.viewRect[1]) + $
> self.viewRect[3]) / 2.
>
> -Rick

Hi Rick,

Thank you very much! Altering the viewplane_rect property does the
trick for me. Great!

Regards,
Erik
