Subject: Re: positioning a TV image within plotting region
Posted by greg.addr on Thu, 29 Sep 2011 14:39:44 GMT
View Forum Message <> Reply to Message

When I do this, I plot the image first (with /noerase, as you have) and then the axes.

good luck!
Greg

Subject: Re: positioning a TV image within plotting region
Posted by greg.addr on Thu, 29 Sep 2011 14:42:07 GMT
View Forum Message <> Reply to Message

I mean the axes with /noerase, of course...

Subject: Re: positioning a TV image within plotting region
Posted by greg michael on Thu, 29 Sep 2011 14:47:31 GMT
View Forum Message <> Reply to Message

One more thing - I would use device coordinates for the plot instead of normal, because you can't size your image in normal coordinates. Otherwise, you might sometimes get a 1-pixel mismatch at the edges.

Subject: Re: positioning a TV image within plotting region
Posted by Matthew on Thu, 29 Sep 2011 19:19:00 GMT
View Forum Message <> Reply to Message

> When I do this, I plot the image first and then the axes.

This just draws the axes on top of the image and hides the fact that
the image is in the wrong spot.

> One more thing - I would use device coordinates for the plot instead of normal, because you can't size your image in normal coordinates. Otherwise, you might sometimes get a 1-pixel mismatch at the edges.

Thanks! You made me realize that I was positioning the plot with
fractions of a pixel, which does not work too well. I still have the
same problem, but now it is a fixed offset, not a varying one. It
seems as though the PLOT procedure draws the bottom and left axes
within the plotting window (defined by !x.window and !y.window),
whereas the top and right axes are outside the plotting window.

In the code below, I used the floor function to truncate any partial pixels from the plot position, reduce the x- and y-size of the image by 1-pixel (the apparent width of the axes), and shifted it up and to the right by 1 pixel. This works for the display window and for all plot sizes that I have tried so far. Now to try postscripts...

```
position = [x0, y0, x1, y1]                    ;position in normal
coordinates
position[0] = floor(position[0] * !d.x_vsize)
position[1] = floor(position[1] * !d.y_vsize)
position[2] = floor(position[2] * !d.x_vsize)
position[3] = floor(position[3] * !d.y_vsize)
xsize = (position[2] - position[0]) - 1
ysize = (position[3] - position[1]) - 1
xstart = position[0] + 1
ystart = position[1] + 1

loadct, 13
plot, time, freq, position=position, /device, /nodata, /noerase
tvscl, congrid(alog(transpose(E_pwr_dens)), xsize, ysize), xstart,
ystart
```

Subject: Re: positioning a TV image within plotting region
Posted by David Fanning on Fri, 30 Sep 2011 15:34:16 GMT
View Forum Message <> Reply to Message

Matthew writes:

```
>  position = [x0, y0, x1, y1]                    ;position in normal
>  coordinates
>  position[0] = floor(position[0] * !d.x_vsize)
>  position[1] = floor(position[1] * !d.y_vsize)
>  position[2] = floor(position[2] * !d.x_vsize)
>  position[3] = floor(position[3] * !d.y_vsize)
>  xsize = (position[2] - position[0]) - 1
>  ysize = (position[3] - position[1]) - 1
>  xstart = position[0] + 1
>  ystart = position[1] + 1
```

I think these two lines:

```
>  position[2] = floor(position[2] * !d.x_vsize)
>  position[3] = floor(position[3] * !d.y_vsize)
```

should use CEIL instead of FLOOR.

But, I think this is essentially correct. I've changed
the way I calculate the image size and start positions
in cgImage this morning to this, and I like the results
very much.

```
; Calculate the image size and start locations.
xsize = Ceil((position[2] - position[0]) * !D.X_VSIZE)-1
ysize = Ceil((position[3] - position[1]) * !D.Y_VSIZE)-1
xstart = Round(position[0] * !D.X_VSIZE)+1
ystart = Round(position[1] * !D.Y_VSIZE)+1
```

You can test the "fit" by typing these commands and resizing
the window:

```
Loadct, 0, NColors=20
cgImage, BytScl(Dist(20), TOP=19), /Axes, /Keep, $
    Background='sky blue', /Erase, /Window
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: positioning a TV image within plotting region
Posted by Matthew on Wed, 05 Oct 2011 17:26:50 GMT

View Forum Message <> Reply to Message

On Sep 30, 11:34 am, David Fanning <n...@dfanning.com> wrote:
> Matthew writes:
>> position = [x0, y0, x1, y1] ;position in normal
>> coordinates
>> position[0] = floor(position[0] * !d.x_vsize)
>> position[1] = floor(position[1] * !d.y_vsize)
>> position[2] = floor(position[2] * !d.x_vsize)
>> position[3] = floor(position[3] * !d.y_vsize)
>> xsize = (position[2] - position[0]) - 1
>> ysize = (position[3] - position[1]) - 1
>> xstart = position[0] + 1
>> ystart = position[1] + 1
>
> I think these two lines:
>

>> position[2] = floor(position[2] * !d.x_vsize)
>> position[3] = floor(position[3] * !d.y_vsize)
>
> should use CEIL instead of FLOOR.
>
> But, I think this is essentially correct. I've changed
> the way I calculate the image size and start positions
> in cgImage this morning to this, and I like the results
> very much.
>
> ; Calculate the image size and start locations.
> xsize = Ceil((position[2] - position[0]) * !D.X_VSIZE)-1
> ysize = Ceil((position[3] - position[1]) * !D.Y_VSIZE)-1
> xstart = Round(position[0] * !D.X_VSIZE)+1
> ystart = Round(position[1] * !D.Y_VSIZE)+1
>
> You can test the "fit" by typing these commands and resizing
> the window:
>
> Loadct, 0, NColors=20
> cgImage, BytScl(Dist(20), TOP=19), /Axes, /Keep, $
> Background='sky blue', /Erase, /Window
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Just a quick note... unless the aspect ratio of the postscript file is the same as that of the display (e.g. as is the case when using psCONFIG()), the variable pixel size will make the initial problem here persist. To fix the problem,

```
x_size_old = !d.x_size
y_size_old = !d.y_size
set_plot, 'PS'

x_size_ps = !d.x_size
y_size_ps = !d.y_size

px_size_x = x_size_ps / x_size_old
px_size_y = y_size_ps / y_size_old
```

```
position[0] = floor(position[0] * !d.x_vsize)
position[1] = floor(position[1] * !d.y_vsize)
position[2] = ceil(position[2] * !d.x_vsize)
position[3] = ceil(position[3] * !d.y_vsize)
xsize = (position[2] - position[0]) - px_size_x
ysize = (position[3] - position[1]) - px_size_y
xstart = position[0] + px_size_x
ystart = position[1] + px_size_y
```

This has worked so far...

Subject: Re: positioning a TV image within plotting region
Posted by David Fanning on Wed, 05 Oct 2011 18:06:07 GMT
View Forum Message <> Reply to Message

Matthew writes:

> Just a quick note... unless the aspect ratio of the postscript file is
> the same as that of the display (e.g. as is the case when using
> psCONFIG()), the variable pixel size will make the initial problem
> here persist. To fix the problem,
>
> x_size_old = !d.x_size
> y_size_old = !d.y_size
> set_plot, 'PS'
>
> x_size_ps = !d.x_size
> y_size_ps = !d.y_size
>
> px_size_x = x_size_ps / x_size_old
> px_size_y = y_size_ps / y_size_old
>
> position[0] = floor(position[0] * !d.x_vsize)
> position[1] = floor(position[1] * !d.y_vsize)
> position[2] = ceil(position[2] * !d.x_vsize)
> position[3] = ceil(position[3] * !d.y_vsize)
> xsize = (position[2] - position[0]) - px_size_x
> ysize = (position[3] - position[1]) - px_size_y
> xstart = position[0] + px_size_x
> ystart = position[1] + px_size_y
>
>
> This has worked so far...

Humm. I don't see how. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: positioning a TV image within plotting region
Posted by Matthew Argall on Mon, 10 Oct 2011 23:22:50 GMT
View Forum Message <> Reply to Message

On Oct 5, 2:06 pm, David Fanning <n...@dfanning.com> wrote:
> Matthew writes:
>> Just a quick note... unless the aspect ratio of the postscript file is
>> the same as that of the display (e.g. as is the case when using
>> psCONFIG()), the variable pixel size will make the initial problem
>> here persist. To fix the problem,
>
>> x_size_old = !d.x_size
>> y_size_old = !d.y_size
>> set_plot, 'PS'
>
>> x_size_ps = !d.x_size
>> y_size_ps = !d.y_size
>
>> px_size_x = x_size_ps / x_size_old
>> px_size_y = y_size_ps / y_size_old
>
>> position[0] = floor(position[0] * !d.x_vsize)
>> position[1] = floor(position[1] * !d.y_vsize)
>> position[2] = ceil(position[2] * !d.x_vsize)
>> position[3] = ceil(position[3] * !d.y_vsize)
>> xsize = (position[2] - position[0]) - px_size_x
>> ysize = (position[3] - position[1]) - px_size_y
>> xstart = position[0] + px_size_x
>> ystart = position[1] + px_size_y
>
>> This has worked so far...
>
> Humm. I don't see how. :-)
>

> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Me neither :-p Perhaps I did not have my contacts in.

I am using cgImage to do this and it looks great, but the problem keeps bugging me and I keep returning to my own program. In the PS file, the plot still covers the axes, which is bad when I am trying to find synchronous events between plots... Anyway, I will figure it out eventually, but I wanted to ask: Do you find any use for PS anymore? Only the high end printers come equipped with PS processing. Until IDL added the PDF option, I usually ended up converting my PS files to TIFF of PNG to have a file anyone could open.
