Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 13:54:15 GMT
View Forum Message <> Reply to Message

Fab writes:

> In the following I am making two plots with cg* routines.
>
> img1 = cgDemoData(7)
> img2 = congrid(img1, 360, 180)
> cgDisplay, 900, 400
> !P.Multi = [0, 2, 1]
> multimargin = [1,1,1,1]
> cgImage, img1, MULTIMARGIN=multimargin, /KEEP_ASPECT_RATIO, /AXES
> cgImage, img2, MULTIMARGIN=multimargin, /KEEP_ASPECT_RATIO, /AXES
> !P.Multi = 0
>
> Nice, but I would like the top limit of the plots being the same
> (meaning translate the second plot upwards a bit). I could'nt find an
> easy solution. There must be one though! Any thoughts?

POSITION keyword. Heterogeneous positioning of plots is not the forte
of !P.MULTI, who is a guy who craves repetition and homogeneity. :-)

And, since you are asking to keep the aspect ratio, you will have to
be prepared to adjust the position of the second image in Y space.
I would probably try something like this.

```
;-----------------------------------------------
img1 = cgDemoData(7)
img2 = congrid(img1, 360, 180)
cgDisplay, 900, 400
cgImage, img1, /KEEP_ASPECT_RATIO, /AXES, $
   POSITION=[0.05, 0.1, 0.475, 0.90], OPOSITION=op

s = Size(img2, /Dimensions)
imgaspect = Float(s[1])/s[0]

xdist = 0.95-0.525
ydist = (xdist * imgaspect) * 2
pos2 = [0.525, op[3]-ydist, 0.95, op[3]]
cgImage, img2, /AXES, POSITION=pos2, /NoErase
;-----------------------------------------------
```

Cheers,

David

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

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by Fabzi on Tue, 12 Feb 2013 14:20:03 GMT
View Forum Message <> Reply to Message

Hi David,

Thanks, I adapted your idea in my code and after a few dozens of tries I
am satisfied with the output ;-) I won't change anything at the code
anymore though!

Cheers,

Fabien

On 02/12/2013 02:54 PM, David Fanning wrote:
> Fab writes:
>
>> In the following I am making two plots with cg* routines.
>>
>> img1 = cgDemoData(7)
>> img2 = congrid(img1, 360, 180)
>> cgDisplay, 900, 400
>> !P.Multi = [0, 2, 1]
>> multimargin = [1,1,1,1]
>> cgImage, img1, MULTIMARGIN=multimargin, /KEEP_ASPECT_RATIO, /AXES
>> cgImage, img2, MULTIMARGIN=multimargin, /KEEP_ASPECT_RATIO, /AXES
>> !P.Multi = 0
>>
>> Nice, but I would like the top limit of the plots being the same
>> (meaning translate the second plot upwards a bit). I could'nt find an
>> easy solution. There must be one though! Any thoughts?
>
> POSITION keyword. Heterogeneous positioning of plots is not the forte
> of !P.MULTI, who is a guy who craves repetition and homogeneity. :-)
>
> And, since you are asking to keep the aspect ratio, you will have to
> be prepared to adjust the position of the second image in Y space.
> I would probably try something like this.
>
> ;---
> img1 = cgDemoData(7)

```
>  img2 = congrid(img1, 360, 180)
>  cgDisplay, 900, 400
>  cgImage, img1, /KEEP_ASPECT_RATIO, /AXES, $
>       POSITION=[0.05, 0.1, 0.475, 0.90], OPOSITION=op
>
>  s = Size(img2, /Dimensions)
>  imgaspect = Float(s[1])/s[0]
>
>  xdist = 0.95-0.525
>  ydist = (xdist * imgaspect) * 2
>  pos2 = [0.525, op[3]-ydist, 0.95, op[3]]
>  cgImage, img2, /AXES, POSITION=pos2, /NoErase
>  ;-------------------------------------------------
>
>
>  Cheers,
>
>  David
>
```

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by Fabzi on Tue, 12 Feb 2013 14:32:04 GMT
View Forum Message <> Reply to Message

Just by pure curiosity: is this problem (displaying two images of
different aspect ratios + a colorbar in a line) anyhow easier in NG?

On 02/12/2013 03:20 PM, Fab wrote:
> Hi David,
>
> Thanks, I adapted your idea in my code and after a few dozens of tries I
> am satisfied with the output ;-) I won't change anything at the code
> anymore though!
>
> Cheers,
>
> Fabien
>
> On 02/12/2013 02:54 PM, David Fanning wrote:
> Fab writes:
>
> In the following I am making two plots with cg* routines.
>
>   img1 = cgDemoData(7)
>   img2 = congrid(img1, 360, 180)
>   cgDisplay, 900, 400
>   !P.Multi = [0, 2, 1]

```
multimargin = [1,1,1,1]
cgImage, img1, MULTIMARGIN=multimargin, /KEEP_ASPECT_RATIO, /AXES
cgImage, img2, MULTIMARGIN=multimargin, /KEEP_ASPECT_RATIO, /AXES
!P.Multi = 0
```

Nice, but I would like the top limit of the plots being the same
(meaning translate the second plot upwards a bit). I could'nt find an
easy solution. There must be one though! Any thoughts?

POSITION keyword. Heterogeneous positioning of plots is not the forte
of !P.MULTI, who is a guy who craves repetition and homogeneity. :-)

And, since you are asking to keep the aspect ratio, you will have to
be prepared to adjust the position of the second image in Y space.
I would probably try something like this.

```
;-------------------------------------------------
img1 = cgDemoData(7)
img2 = congrid(img1, 360, 180)
cgDisplay, 900, 400
cgImage, img1, /KEEP_ASPECT_RATIO, /AXES, $
    POSITION=[0.05, 0.1, 0.475, 0.90], OPOSITION=op

s = Size(img2, /Dimensions)
imgaspect = Float(s[1])/s[0]

xdist = 0.95-0.525
ydist = (xdist * imgaspect) * 2
pos2 = [0.525, op[3]-ydist, 0.95, op[3]]
cgImage, img2, /AXES, POSITION=pos2, /NoErase
;-------------------------------------------------
```

Cheers,

David

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 14:37:18 GMT
View Forum Message <> Reply to Message

Fab writes:

> Thanks, I adapted your idea in my code and after a few dozens of tries I
> am satisfied with the output ;-) I won't change anything at the code
> anymore though!

Here is a programming challenge that will make this very much easier next time.

Modify ASPECT (and feel free to rename the new program cgAspect) to accept an input position for which it can calculate a final position with the correct aspect ratio. This ought to be possible, since cgImage can apparently do this. ;-)

Cheers,

David

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

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 14:40:19 GMT
View Forum Message <> Reply to Message

Fab writes:

> Just by pure curiosity: is this problem (displaying two images of
> different aspect ratios + a colorbar in a line) anyhow easier in NG?

If this is what you are trying to do, then I would definitely get cgLayout involved! Same idea as before, but you will have a lot more flexibility to leave room and space for color bars.

Cheers,

David

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

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 14:43:41 GMT

David Fanning writes:

> If this is what you are trying to do, then I would definitely get
> cgLayout involved! Same idea as before, but you will have a lot more
> flexibility to leave room and space for color bars.

I forgot to mention that the /FIT keyword on cgColorbar does a pretty good job of fitting a color bar to an image or contour plot without you having to do much calculation. It works well in conjunction with cgLayout, for example.

Cheers,

David

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

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 16:31:11 GMT

David Fanning writes:

>
> Fab writes:
>
>> Thanks, I adapted your idea in my code and after a few dozens of tries I
>> am satisfied with the output ;-) I won't change anything at the code
>> anymore though!
>
> Here is a programming challenge that will make this very much easier
> next time.
>
> Modify ASPECT (and feel free to rename the new program cgAspect) to
> accept an input position for which it can calculate a final position
> with the correct aspect ratio. This ought to be possible, since cgImage
> can apparently do this. ;-)

Alright, this was easy enough to do, so I just had Coyote do it. :-)

Find it here:

  http://www.idlcoyote.com/programs/cgaspect.pro

And here is how I would use the routine to display these two images with
colorbars:

```
;-----------------------------------------------
img1 = cgDemoData(7)
img2 = congrid(img1, 360, 180)
cgDisplay, 900, 450

pos = cgLayout([2,1], OYMargin=[4, 11])
cgImage, img1, /KEEP_ASPECT_RATIO, CTIndex=3, $
   /AXES, POSITION=pos[*,0], OPOSITION=op
cgColorbar, CTIndex=3, /Fit

pos2 = pos[*,1]
ydiff = op[3] - pos2[3]
pos2[3] = op[3]
pos2[1] = pos2[1] + ydiff

img2pos = cgAspect(Position=pos2, Aspect=img2, Align='top')
cgImage, img2, /AXES, POSITION=img2pos, CTIndex=2, /NoErase
cgColorbar, CTIndex=2, /Fit
END
;-----------------------------------------------
```

Cheers,

David

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

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 17:06:48 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Alright, this was easy enough to do, so I just had Coyote do it. :-)

>
> Find it here:
>
> http://www.idlcoyote.com/programs/cgaspect.pro
>
> And here is how I would use the routine to display these two images with
> colorbars:
>
> ;---
> img1 = cgDemoData(7)
> img2 = congrid(img1, 360, 180)
> cgDisplay, 900, 450
>
> pos = cgLayout([2,1], OYMargin=[4, 11])
> cgImage, img1, /KEEP_ASPECT_RATIO, CTIndex=3, $
>   /AXES, POSITION=pos[*,0], OPOSITION=op
> cgColorbar, CTIndex=3, /Fit
>
> pos2 = pos[*,1]
> ydiff = op[3] - pos2[3]
> pos2[3] = op[3]
> pos2[1] = pos2[1] + ydiff
>
> img2pos = cgAspect(Position=pos2, Aspect=img2, Align='top')
> cgImage, img2, /AXES, POSITION=img2pos, CTIndex=2, /NoErase
> cgColorbar, CTIndex=2, /Fit
> END
> ;---

That's a little convoluted. Just for the record, here is a better
formulation that takes better advantage of cgAspect.

```
 ;------------------------------------------------------- --
img1 = cgDemoData(7)
img2 = congrid(img1, 360, 180)
cgDisplay, 900, 450

pos = cgLayout([2,1], OYMargin=[4, 11], OXMargin=[5, 8], XGap=6)
pos1 = pos[*,0]
img1pos = cgAspect(Position=pos1, Aspect=img1, Align='top')
cgImage, img1, CTIndex=3, /AXES, POSITION=img1pos, OPOSITION=op
cgColorbar, CTIndex=3, /Fit

pos2 = pos[*,1]
ydiff = op[3] - pos2[3]
pos2[3] = op[3]
pos2[1] = pos2[1] + ydiff
```

```
img2pos = cgAspect(Position=pos2, Aspect=img2, Align='top')
cgImage, img2, /AXES, POSITION=img2pos, CTIndex=2, /NoErase
cgColorbar, CTIndex=2, /Fit
END
 ;------------------------------------------------------------ --
```

Cheers,

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

Subject: Re: cGimage, Multiplot with /KEEP_ASPECT_RATIO
Posted by David Fanning on Tue, 12 Feb 2013 17:12:57 GMT
View Forum Message <> Reply to Message

David Fanning writes:

```
>  That's a little convoluted. Just for the record, here is a better
>  formulation that takes better advantage of cgAspect.
>
>   ;------------------------------------------------------------ --
>  img1 = cgDemoData(7)
>  img2 = congrid(img1, 360, 180)
>  cgDisplay, 900, 450
>
>  pos = cgLayout([2,1], OYMargin=[4, 11], OXMargin=[5, 8], XGap=6)
>  pos1 = pos[*,0]
>  img1pos = cgAspect(Position=pos1, Aspect=img1, Align='top')
>  cgImage, img1, CTIndex=3, /AXES, POSITION=img1pos, OPOSITION=op
>  cgColorbar, CTIndex=3, /Fit
>
>  pos2 = pos[*,1]
>  ydiff = op[3] - pos2[3]
>  pos2[3] = op[3]
>  pos2[1] = pos2[1] + ydiff
>
>  img2pos = cgAspect(Position=pos2, Aspect=img2, Align='top')
>  cgImage, img2, /AXES, POSITION=img2pos, CTIndex=2, /NoErase
>  cgColorbar, CTIndex=2, /Fit
>  END
>   ;------------------------------------------------------------ --
```

Now that I look at the code in the light of day, it is even simpler than
that! Try this.

```
 ;---------------------------------------------------------- --
img1 = cgDemoData(7)
img2 = congrid(img1, 360, 180)
cgDisplay, 900, 450

pos = cgLayout([2,1], OYMargin=[4, 11], OXMargin=[5, 8], XGap=6)
pos1 = pos[*,0]
img1pos = cgAspect(Position=pos1, Aspect=img1, Align='top')
cgImage, img1, CTIndex=3, /AXES, POSITION=img1pos, OPOSITION=op
cgColorbar, CTIndex=3, /Fit

pos2 = pos[*,1]
img2pos = cgAspect(Position=pos2, Aspect=img2, Align='top')
cgImage, img2, /AXES, POSITION=img2pos, CTIndex=2, /NoErase
cgColorbar, CTIndex=2, /Fit
END
 ;---------------------------------------------------------- --
```

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