Subject: Re: Images with coordinate axes, log scale etc.
Posted by jkj on Tue, 26 Feb 2008 17:05:34 GMT
View Forum Message <> Reply to Message

On Feb 25, 5:58 pm, pgri...@gmail.com wrote:
> Hi folks,
>
> there seems to be a recurrent theme in this newsgroup of questions
> about how to plot images with plot axes (be it linear or log
> scale) and/or irregularly gridded images.
>
> For example, people may have this image
>
> im=dist(128,128)
>
> and two coordinates vectors
>
> x=findgen(128)+1
> y=findgen(128)+1
>
> and may want to plot the image with the axes overlaid on it,
> in linear or log scale.
>
> There are some way to accomplish this in direct graphics:
> a) use a routine like "spectro_plot" or "plot_image" from various
> sources
> b) use contour (strange as it may seem,
> contour,im,x,y,levels=findgen(256)/255*max(im),/fill,/xstyle ,/ystyle
> does the job, although it is slow and introduces some kind of
> (possibly unwanted) smoothing. If you don't believe this can be
> used as a first approximation, try it on the "rose.jpg" image
> for the IDL examples. Works quite well (for a one liner).
> c) do it from scratch
>
> Now, a) requires a library that people may not have or want,
> and b) is not really an option for serious work because of
> artifacts from the contouring algorithm, so I decided it was
> time to write a short stand-alone program for plotting images
> with axes. I took the main ideas from SSW's "spectro_plot",
> which in turn stole it from the Wind software package, I think.
>
> The routine can be found athttp://hea-www.harvard.edu/~pgrigis/idl_stuff/pg_plotimage .pro
> and can be used pretty much as one would use "plot", only
> with the additional required inputs of the x and y vectors
> with the coordinates.
>
> Here's an example (using the above definitions)
>

```
>  loadct,5
>  pg_plotimage,im,x,y,/xlog,/xstyle,/ystyle,yrange=[-10,200]$
>  ,title='distorted dist() image'
>
>  or, if one likes some smoothing
>
>   pg_plotimage,im,x,y,/xlog,/xstyle,/ystyle,yrange=[-10,200],/ smooth $
>  ,title='smoothed',xtitle='some units',position=[0.2,0.2,0.8,0.8]
>
>  Using the "rose" image example:
>
>  rose = READ_IMAGE(FILEPATH('rose.jpg', $
>  SUBDIRECTORY=['examples','data']))
>  dim = SIZE(rose,/DIM)
>  nx = dim[1] & ny = dim[2]
>  x = findgen(nx)+1
>  y = findgen(ny)+1
>  im=bytscl(total(rose,1))
>  loadct,0
>
>  pg_plotimage,im,x,y,/xstyle,/ystyle,/smooth,/ylog $
>  ,title='Gray Rose',xtitle='some units',position=[0.2,0.2,0.8,0.8]
>
>  Maybe this can be of some use. I haven't really tested heavily,
>  so let me know of you find bugs or have suggestion for improvement.
>  Just be aware that if you have negative or 0 axis values and try the
>  log keywords, you are beyond redemption ;-)
>
>  How does the routine work? The idea is to interpolate the
>  input image at the location of every pixel in the plot window
>  (the coordinates associated with each pixel depend then on
>  the plot & window size and the linear vs. log axis choice).
>  This is reasonably fast, as interpolation of, say, about
>  1 million data points is not a hard thing to do.
>
>  Cheers,
>  Paolo
```

Worked fine for me on Linux running IDL-5.5, using all combinations of x/ylog. Most of my focus right now is object graphics, but I'll make a note of this thread and place it in the code so that if I make any changes which seem useful I will know who to contact!
Thanks,
-Kevin