Subject: Re: New Image Processing Routines
Posted by jdsmith on Fri, 24 Feb 2006 03:37:39 GMT
View Forum Message <> Reply to Message

Looks good, David. If you really want to impress astronomers, you might add a few range compression methods:

1. histogram equalization (done well, is rare).
2. The ASINCH scaling, which is linear at the low end, and logarithmic at the high end (which is about perfect for showing noise properties and high contrast features all at once). Robert Lupton wrote such a beast in IDL already:

http://cheops1.uchicago.edu/idlhelp/sdssidl/plotting/tvasinh .html

His website links to a little paper describing the method (which has some very nice properties, but occasionally produces strange-looking images:

http://www.astro.princeton.edu/~rhl/PrettyPictures/

The trick will be coming up with an easier way to set the 2 parameters required that affect the scaling.

JD

Subject: Re: New Image Processing Routines
Posted by David Fanning on Fri, 24 Feb 2006 05:48:14 GMT
View Forum Message <> Reply to Message

J.D. Smith writes:

> Looks good, David. If you really want to impress astronomers, you might
> add a few range compression methods:
>
> 1. histogram equalization (done well, is rare).

I'm not sure what "done well" means. :-)

I've implemented the algorithm for histogram matching in Gonzalez and Woods (HistoMatch) and if I give that algorithm a flat histogram (Replicate(1,256)), I get nearly the identical result as the IDL Hist_Equal command. Certainly the shape of the resulting histogram is identical. We differ only a little bit on either end, but the results seems insignificant.

I presume that means both methods are "done well." :-)

> 2. The ASINCH scaling, which is linear at the low end, and logarithmic
> at the high end (which is about perfect for showing noise properties and
> high contrast features all at once). Robert Lupton wrote such a beast
> in IDL already:
>
> http://cheops1.uchicago.edu/idlhelp/sdssidl/plotting/tvasinh .html
>
> His website links to a little paper describing the method (which has
> some very nice properties, but occasionally produces strange-looking
> images:
>
> http://www.astro.princeton.edu/~rhl/PrettyPictures/
>
> The trick will be coming up with an easier way to set the 2 parameters
> required that affect the scaling.

I've been reading this paper the past couple of days. The algorithm seems dead simple, but I don't yet have an intuitive feel for what "alpha" and "nonlinearity" mean. (Although I think I am starting to understand the latter much better than the former.) Maybe I just don't have the right images. I've downloaded the ones used in the paper and will fool around with those tomorrow. Maybe enlightenment awaits. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: New Image Processing Routines
Posted by David Fanning on Fri, 24 Feb 2006 16:07:15 GMT
View Forum Message <> Reply to Message

J.D. Smith writes:

> 2. The ASINCH scaling, which is linear at the low end, and logarithmic
> at the high end (which is about perfect for showing noise properties and
> high contrast features all at once). Robert Lupton wrote such a beast
> in IDL already:
>
> http://cheops1.uchicago.edu/idlhelp/sdssidl/plotting/tvasinh .html
>

> His website links to a little paper describing the method (which has
> some very nice properties, but occasionally produces strange-looking
> images:
>
> http://www.astro.princeton.edu/~rhl/PrettyPictures/
>
> The trick will be coming up with an easier way to set the 2 parameters
> required that affect the scaling.

Ah, this is like a tuned gamma function! Nice. I get it! It
is helpful to have some pictures to view:

```
alpha = 0.2 ; 0.0 > alpha < 1.0
nonlinear = 8 ; 1 > nonlinear < inf
data = indgen(256)
Plot, asinh(alpha*nonlinear*image)/nonlinear
```

Now vary alpha and nonlinear and you will see how to "tune"
the function.

Cheers,

David

P.S. Requires the ASTRO library routine ASINH.
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: New Image Processing Routines
Posted by David Fanning on Fri, 24 Feb 2006 21:08:41 GMT
View Forum Message <> Reply to Message

J.D. Smith writes:

> 2. The ASINCH scaling, which is linear at the low end, and logarithmic
> at the high end (which is about perfect for showing noise properties and
> high contrast features all at once). Robert Lupton wrote such a beast
> in IDL already:
>
> http://cheops1.uchicago.edu/idlhelp/sdssidl/plotting/tvasinh .html
>
> His website links to a little paper describing the method (which has
> some very nice properties, but occasionally produces strange-looking
> images:
>

> http://www.astro.princeton.edu/~rhl/PrettyPictures/
>
> The trick will be coming up with an easier way to set the 2 parameters
> required that affect the scaling.

OK, sports fans. Here are two new programs. I have
to admit, with the right images (low signal to noise)
this inverse hyperbolic sine function is GREAT!
I can't wait to try this out on images with the new
CCD camera we just got for the school.

ASINHSCL -- Like BYTSCL and IMGSCL, but for images
with "asinh magnitudes" ala Lupton, et. al.

XSTRETCH_ASINH -- Like XSTRETCH, but with ASINH
scaling instead of gamma scaling. GUI front
end for ASINHSCL.

Found in the usual places. Normal images can look
a little weird, as JD suggests. Astronomers and
people with bad photographic skills will probably
find these programs useful. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: New Image Processing Routines
Posted by George N. White III on Sun, 05 Mar 2006 19:35:00 GMT
View Forum Message <> Reply to Message

On Fri, 24 Feb 2006, David Fanning wrote:

> OK, sports fans. Here are two new programs. I have
> to admit, with the right images (low signal to noise)
> this inverse hyperbolic sine function is GREAT!
> I can't wait to try this out on images with the new
> CCD camera we just got for the school.
>
> ASINHSCL -- Like BYTSCL and IMGSCL, but for images
> with "asinh magnitudes" ala Lupton, et. al.
>

> XSTRETCH_ASINH -- Like XSTRETCH, but with ASINH
> scaling instead of gamma scaling. GUI front
> end for ASINHSCL.
>
> Found in the usual places. Normal images can look
> a little weird, as JD suggests. Astronomers and
> people with bad photographic skills will probably
> find these programs useful. :-)

You might find techniques for "high dynamic range" images interesting, include an IDL implementation of the iCAM (image color appearance model) <http://www.cis.rit.edu/mcsl/iCAM/>.

At lot of this work comes out of places like Industrial Light and Magic, who use a file format called EXR based on the same 16-bit f.p. format as nVidia graphics engines. This protects against posterization that is a problem when doing heavy image transformations on 8-bit data. With IDL I think we are stuck with floats in tiff files for the present, although there has been some discussion of using the graphics engine for image processing.

In remotes sensing there are some variables (ocean chlorophyll, which ranges from 0.01 to 100 mg/m^3) with a natural range far beyond any display device. It is customary to display these with bytscl(image) and a color lookup table with 150-255 entries, thereby demolishing much of the fantastic detail in the original dataset. Ultimately this needs a different approach from the problem of digital photos taken in poor light, but I expect there are useful tools from the HDR domain that can be adapted for use with remote sensing images.

--
George N. White III <aa056@chebucto.ns.ca>

Subject: Re: New Image Processing Routines
Posted by Paolo Grigis on Mon, 06 Mar 2006 10:20:52 GMT
View Forum Message <> Reply to Message

J.D. Smith wrote:
> Looks good, David. If you really want to impress astronomers, you might
> add a few range compression methods:
>
> 1. histogram equalization (done well, is rare).

> 2. The ASINCH scaling, which is linear at the low end, and logarithmic
> at the high end (which is about perfect for showing noise properties and
> high contrast features all at once).

I have used log(x+const) as a scaling for that purpose (where the ratio of const to the noise level controls how much the noise is emphasized), and was quite happy about the results for X-ray dynamic spectrograms.

I guess the results are probably quite similar to the asinh scaling, after all both functions are close to each other. I'll try out the asinh scaling next time I'll need that kind of rescaling. Thanks for the tip.

Ciao,
Paolo

Robert Lupton wrote such a beast
> in IDL already:
>
> http://cheops1.uchicago.edu/idlhelp/sdssidl/plotting/tvasinh .html
>
> His website links to a little paper describing the method (which has
> some very nice properties, but occasionally produces strange-looking
> images:
>
> http://www.astro.princeton.edu/~rhl/PrettyPictures/
>
> The trick will be coming up with an easier way to set the 2 parameters
> required that affect the scaling.
>
> JD

Subject: Re: New Image Processing Routines
Posted by Marshall Perrin on Fri, 21 Apr 2006 12:02:31 GMT
View Forum Message <> Reply to Message

David Fanning <davidf@dfanning.com> wrote:
> OK, sports fans. Here are two new programs. I have
> to admit, with the right images (low signal to noise)
> this inverse hyperbolic sine function is GREAT!
> I can't wait to try this out on images with the new
> CCD camera we just got for the school.
>
> ASINHSCL -- Like BYTSCL and IMGSCL, but for images
> with "asinh magnitudes" ala Lupton, et. al.

Hi David,

I've finally gotten around to trying out asinh scaling (something I've been meaning to do since, oh, 2004 when I first saw the paper by Lupton et al.). First off, let me say that your new and improved xstretch is really

spectacular. I love the drag and drop setting of the min and max! Ah, simple things can seem so spectacular to those of us used to direct graphics. :-/

But on to my main point. I think your ASINHSCL can be simplified considerably, because the Alpha parameter is unnecessary. Or rather, Alpha and Beta aren't independent. What actually sets the shape of the scaling law is their product.

Try varying them inversely (i.e. move Alpha up by a factor of 5, Beta down by a factor of 5, so that their product remains constant) and I think you'll see that the scaled image is unchanged. I've also verified this behavior with some simple 1D plots (code attached below). I've thought about the equations for a bit, but I don't yet have a rigorous proof that this is true, but it is in all the numerical examples I've tried. Then again, it's 4:30 am and even on astronomer hours that's a bad time for doing math, so maybe I'll figure it out tomorrow.

In any case, I think you can just remove alpha entirely and use beta alone to tune the scaling. I note that Lupton et al. only mention Beta as a tuning parameter as far as I can tell. I see there's an alpha in Eric Sheldon's tvasinh.pro that you based your code on, but I have no idea where he got it from...

- Marshall

```
pro testasinh

x = findgen(100)

plot,bytscl(x)
for i=-3,5 do begin
       oplot,x,asinhscl(x,alpha=10,beta=10.^i)
endfor

for i=-3,5 do begin
   oplot,x,asinhscl(x,alpha=10.^i,beta=10),lines=2,$
   color=fsc_color('red')
endfor

xyouts,60,100,"Varying Beta, with Alpha=10"
xyouts,60,120,"Varying Alpha, with Beta=10",color=fsc_color('red')

end
```