Subject: Re: New Image Processing Routines
Posted by David Fanning on Mon, 24 Apr 2006 19:24:54 GMT
View Forum Message <> Reply to Message

Marshall Perrin writes:

> Actually, it was while brushing my teeth that I had the epiphany.
> The relelvant line of code is the scaling, of course:
>
> output = Scale_Vector(ASinhScl_ASinh(alpha*beta*Temporary(output))/be ta, $
> minOut, maxOut, /NAN, Double=1)
>
> The only way in which alpha affects things is as a multiplicative constant
> times the input (also named 'output' above just to be tricky :-) We know that
> it's the ratio of beta to the input that sets the amount of nonlinearity
> in the data. Quoting from Lupton et al 1999, right after eq. 4:
>
> For x -> \infinity, \mu approaches m for any choice of \beta. On
> the other hand, when |x| <~ b, \mu is linear in x.
>
> So it's the ratio of beta to x that matters, and since alpha just has the
> effect of scaling x, it doesn't actually add an additional degree of freedom.
> I'm fully convinced now that alpha doesn't give you anything new.

Ok, I've convinced myself I agree with you. :-)

> Next problem: the beta in your code (and in all the other IDL implementations
> of asinh scaling, as far as I can tell!) is actually the *inverse* of the
> parameter b in Lupton et al. Quoting now from Lupton et al. 2004, shortly
> after Eq. 2:
>
> We take F = arcsinh(x/Beta), where the softening parameter Beta is
> chosen to bring out the desired details.
>
> So they say they are dividing the input by Beta, while you're multiplying it.
> Actually, they *say* they are dividing the input by Beta, but if you download
> their code from http://cosmo.nyu.edu/hogg/visualization/ and look at
> nw_asinh_fit.pro, the relevant line of code is
>
> val = asinh(radius*nonlinearity)/nonlinearity
>
> The same approach is used in Dave Schlegel's djs_rgb_make available in his
> IDLUTILS library. So it seems that they've defined nonlinearity = 1./beta .
> Right? And thus your Beta is the inverse of theirs.
>
> Does this actually matter? No, not really. Either way works fine. It's
> just a slightly confusing state of affairs for anyone trying to match
> up IDL code to published algorithms. I *think* the simplest solution

> is to rename in your code "beta" to "nonlinearity" and put in a note
> that nonlinearity = 1./b. But hey, who am I to tell you what to name
> your variables? :-)

I'm still confused about this part. I've named the variable nonlinearity, and I am happy that a value of 0 means a linear fit. Increasing the value of the variable changes the shape of the curve, so that the linear part gets shorter and steeper and the logarithmic part gets longer and flatter with increasing values. But, the values have to increase in a logarithmic fashion for this to make much sense. This suggests to me that I haven't hit on exactly the right formula yet.

In any case, I modified XSTRETCH to take advantage of the new ASINHSCL routine. And while I was at it, added several more features. You can now specify the gamma value, as well as the nonlinearity value with a combobox widget. This allows you to select decent choices off a pull-down menu, but if you don't like the choices I give you, you can just type your own values in. I even put in the linear "scaling curve" just for JD. (I previously had assumed people would know what a linear scaling curve would look like. But JD's probably right. You never know what people need. :-)

As a working hypothesis I think my "nonlinearity" parameter doesn't directly correlate with Lupton's "beta" parameter, but I think this is because I scale the image data into the range of 0 to 1 before I apply the scaling. This is necessary, in my case, because I can't rely on image data values falling into a particular range. I need to start with something I can depend on.

The result is certainly more intuitive, if not more correct. :-)

http://www.dfanning.com/programs/asinhscl.pro
http://www.dfanning.com/programs/xstretch.pro

Cheers,

David

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