
Subject: Re: SHOWIMAGE posted (companion to SAVEIMAGE)
Posted by [Liam Gumley](#) on Thu, 05 Aug 1999 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning <davidf@dfanning.com> wrote in message
news:MPG.1213d946b5ccbaeb98987a@news.frii.com...
> I have a couple of questions and a bug report. :-(

Given that I dashed these programs off in a couple of days, I'd be surprised
if no bugs showed up!

> Why are you doing this in ShowImage:
>
> ; - Sort the color table from darkest to brightest
>
> table_sum = total([long(r), long(g), long(b)], 2)
> table_index = sort(table_sum)
> image_index = sort(table_index)
> r = r[table_index]
> g = g[table_index]
> b = b[table_index]
> oldimage = image
> image[*] = image_index[temporary(oldimage)]
>
> I just can't figure out what this does.

The color tables returned by COLOR_QUAN usually look rather odd; when you
bring up XLOADCT (or preferably XCOLORS), the color table is not pleasing to
the eye. If you then grab one of the min/max sliders to adjust the color
table, the color table is corrupted and the image is spoiled. However if you
sort the color table entries from darkest to brightest, moving the
XLOADCT/XCOLORS sliders at either end won't hurt the image quality too much.
I must admit I shamelessly copied this idea from the CLEAN_GIF routine
written by Craig DeForest (see
<http://www.astro.washington.edu/deutsch/idl/htmlhelp/index.html>).

> And why do you reduce the number of colors on an 8-bit
> display? Is it because a GIF image, for example, may have
> more colors than your IDL session?

Yes. According to the documentation for REDUCE_COLORS, it "reduces the
number of colors used in an image by eliminating pixel values without
members".

> What if it does and you
> can't reduce the number of colors? Is there any way to
> display the GIF image correctly then?

Yes. If the number of colors in the current session is smaller than the size of the image color table, I go through this tortuous process to squeeze the number of colors:

- (1) Convert the 8-bit image to 24-bit,
- (2) Use COLOR_QUAN to make a quantized version of the image with COLORS=ID.TABLE_SIZE.

> I happened to be fooling around in IDL when this post
> showed up, so I just downloaded the code and tried it
> out. I'm running IDL 5.2 in 24-bit mode. I thought I
> would see how it did on a 24-bit image, so I chose
> the rose.dat file in the IDL/examples/data directory:
>
> ShowImage, 'C:\RSI\IDL52\EXAMPLES\DATA\ROSE.JPG'
>
> It worked great, except the colors were all screwed up. :-(
> The problem, of course, is that I had a color table loaded
> in my IDL session (and the DECOMPOSED keyword set to 0,
> naturally), and IDL took these RGB colors through
> the color tables because in this Windows version of
> IDL it does that.

Thanks for pointing this out. I actually tried this on my SGI box, and it seemed like the DECOMPOSED keyword wasn't required. I'll patch the code as you have suggested.

> But here is the REALLY good news! Your program works *perfectly*
> in IDL 5.2.1, because the folks at RSI really do listen to users
> sometimes, and they fixed this annoying bug! :-)

Good news indeed.

> But all in all, some really nice, extremely useful software.
> Thanks for taking the time to do it. I have a feeling I'm going
> to be using it just about every day. :-)

Thanks for the compliment. Many ideas in these two programs were prompted by the vigorous discussions we've had about color in the newsgroup.

> P.S. Martin Schultz hasn't been in contact with you, has he?
> I used to hear from him several times a day for a week after
> I posted new software. I hope he's not going soft back there
> in Germany. :-)

Not recently. But if I was in Germany in the summertime, IDL would be the last thing on my mind (the castles, the forests, the rivers, the beer, oh my!).

Cheers,
Liam.
<http://cimss.ssec.wisc.edu/~gumley/>
