

Re: GIF encoding

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.programmer/2007-09/msg02270.html>

- *From:* kitkatrobins <kitkat_robins@xxxxxxxxxxx>
 - *Date:* Wed, 26 Sep 2007 10:47:19 -0000
-

On 26 Sep, 01:53, Joshua Cranmer <Pidgeo...@xxxxxxxxxxx> wrote:

Roedy Green wrote:

On Tue, 25 Sep 2007 16:04:43 GMT, "Andrew Thompson" <u32984@uwe> wrote, quoted or indirectly quoted someone who said :

I had not realised GIF's could have 'compression'.

This is a sore point Unisys got hold of the patent to LZW compression used in GIF and they started threatening people with it. It is expired now, though you never know what a lawyer could do.

I think though your code does do LZW, thumbing its nose at Unisys.

The US LZW patent expired on June 20, 2003. The counterpart patents in the United Kingdom, France, Germany and Italy expired on June 18, 2004, the Japanese counterpart patents expired on June 20, 2004 and the counterpart Canadian patent expired on July 7, 2004. Consequently, while Unisys has further patents and patent applications relating to improvements to the LZW technique, the GIF format may now be used freely.

I believe Sun added in to Java 6 full GIF compatibility in ImageIO because the patents had long since expired by then.

LZW compression as implemented in GIF is now fully free!

--

Beware of bugs in the above code; I have only proved it correct, not tried it. -- Donald E. Knuth

Re: GIF encoding

Thanks for your replies.

Having looked further into it, I can see the file size of the "remade" animated gif gets larger when adding the frame information (encoded by the LZW compression). I think its creating a palette based on the first frame and thus when adding the additional frames its not able to compress well as its only taking the first frame into account and leads to additional file size, whereas the application ive been comparing agaisnt (Easy Gif animated) is able to compress well using all the information from each frame.

To recap:

I have an animated gif (40k). I can split it into 15 frames, each of which are 256 colours and of good quality. I run these images through ngif (a c#.net port from Java, can be found on code project) and my new animated gif is 80k. However, if I run this using an animated gif application (Easy Gif Animated) it produces an animated gif of 40k, much like the original.

One thing I have noticed is if I process the original and the Easy gif animated version both have less colours in each of their frames. For e.g. The first frame has a yellow border, which should be in each of the frames but it is not if you look at each of the frames (but still displays correctly when animated in IE/firefox). The yellow border is however in each of the frames of the ngif produced version, which would account for the extra file sizes. I also believe the original and the easy gif animated version are not storing local colour palettes, whereas the ngif is.

Therefore I believe its in the encoding that is the issue, basically..
how can I improve this? People must have come across this issue before.

.