

Fun with bits and bytes

I have been doing some low-level video processing recently. I thought that since the I-frames of an mpeg movie are very similar to a jpeg image, it would be simple to extract these. While using a SampleGrabber in DirectShow or, I suppose, some switches with ffmpeg, it's probably not too difficult, I wanted a more "basic" approach.

So I started to study the Mpeg-1 standard carefully and write some code in C#. After awhile, my code parses the headers (pack, system, and packet) of a system stream, finds its way to an I-frame, decodes the DCT coefficients, feeds it to the IDCT, and stuffs the result in a buffer that I pass to a constructor of Bitmap object that I save.

I'm not a big fan (that's an understatement) of software patents and since the Mpeg standards are replete with these, I was happy with an abstract view like a DirectShow filter, for example. But using FFMpeg has forced me to pay more attention to low-level stuff.

It seems that most Mpeg codecs are based on the "reference codec", I didn't really look at it. I just read the standard and wrote code to do the task. For example, the code for the IDCT is a direct translation of the mathematical formula, hence the result is not very efficient. But it was certainly a good learning exercise.

I'll probably write some materials about this exercise in the near future but before I'd like to finish a more user-friendly version of my wrapper for FFMpeg from .Net.