lucid

Digital Video: A Practical Guide

Lucid Communications Ltd
Prepared by Neil Turner
January 2006

Document History

Version	Author	Comments
v1.0	Neil Turner	Initial Release

1.   Executive Summary

From time to time, it is necessary for web developers to make video content available to their users. But the world of digital video is littered with confusing acronyms, contradictory standards and confusing concepts. This document is intended to provide a rough guide to the use of digital video in a web context, clarifying the main concepts and comparing the major standards.

2.   Format vs Codec

The first stumbling block most people run across when trying to understand digital video is the difference between a video file's *format* and its *codec*. In fact, the difference is very simple.

The **format** of a video file defines its structure; how it's arranged on the disk, how a user agent should render it, what metadata is stored, and which application should be used to run it. The format of a file can generally be deduced based on the filename extension (e.g. *file*.**avi** or *file*.**mpg***)*

A **codec** (short for compressor-decompressor) is a compression algorithm that takes a video file and uses various techniques (discussed in section 3) to reduce its size on disk.

The confusion tends to arise because most codecs can be applied to several different file formats, and many codecs also have an associated file format. For example, the DivX codec can be applied to the **.avi**, **.mpg** or **.divx** formats, among others, while the H.264 codec can be applied to **.mov**, **.avi**, or **.mpg** formats.

3.   Compression Techniques

Good compression is essential for digital video, and nowhere is this more the case than in a web context. Consider the following:

Required Compression Ratios for Online Television

Channel	Bit Rate	NTSC/PAL 168 Mb/s	HDTV 933 Mb/s	Film Quality 2300 Mb/s
Modem Dialup	56kb/s	3000:1	17000:1	41000:1
ISDN	128kb/s	1313:1	7440:1	17950:1
ADSL 1Mb	1024kb/s	164:1	930:1	2240:1
Ethernet	10Mb/s	17:1	95:1	230:1
T3	42Mb/s	4:1	22:1	54:1
Fibre Optics	200Mb/s	1:1	5:1	11:1

Figure 1: Required Compression Ratios for Package TV via Commercial Channels

What this chart tells us is that in order to provide digital video at a reasonable speed (where it takes no longer to download than it does to play); one would need to achieve very good compression.

There are two types of compression. *Lossless* compression makes a file smaller while ensuring that the compressed file contains all of the information present in the original. Unfortunately, there is a limit to how effective lossless compression can be in reducing the size of a video file (the very best lossless compression algorithms can produce a compression ratio of around 3:1 for video).

On the other hand, *lossy* compression works on the basis that the human eye (and ear) doesn't process all of the information presented to it at any given moment. Lossy compression algorithms seek to present the viewer with only the most important information, discarding the rest. As a result, a ratio of around 300:1 can be achieved by current algorithms with little or no visible loss of quality.

This section discusses the techniques by which compression is achieved.

3.1.   Single Frame Compression

A video file can of course be seen as a series of images shown one after another. Naturally therefore, a good first step in compressing video is to try to make each individual frame as small as possible. There are a number of techniques which are used to compress images, including:

- **Fractal Compression**, which relies on the fact that in any image, certain parts of the image will look like other parts, so information for these parts need only be stored once.

- **Chroma Subsampling**, in which the number of colours is reduced by substituting a single colour for a series of other, very similar colours. This takes advantage of the fact that the human eye is relatively bad at distinguishing between similar shades. For example, these two blocks: ■ ■ are in fact different colours.

- The **Discrete Cosine Transformation** and other related mathematical conversions, which recognise that most of the information in an image is present in a relatively limited range of frequencies, and seeks to retain only these.

- **Wavelet Compression**, which first divides an image into several sub-images on varying frequency bands, then compresses each of these sub-images in turn. This makes this method very useful for representing images in which there is a high contrast, such as a picture of stars in the night sky.

3.2.   Inter-Frame Compression

The first logical step in compressing a video file made up of compressed image files is to remove all the redundant data. By the very nature of video, a lot of the image stays the same from frame to frame – the background may be static (or at least mostly static), there may be unmoving objects, and so on.

Consider the following series of images:

Figure 2: Three Successive Frames[*]

It is fairly obvious that the second and third images differ only very slightly from the original. As a result, a video file need only contain the first frame (referred to as the **key frame**), and the *changes* that are necessary to produce the second and third frames (these changes are stored in what is known as a **delta frame**).

3.3. Predictive Compression

This compression trick is also based on the idea that video doesn't change much from frame to frame, and that objects moving around tend to follow relatively predictable paths. By breaking up a video file into a series of small blocks (rather than into individual pixels), a compression algorithm can track the movement of an object (that is, a block that looks the same from frame to frame) around the screen. If the compressor knows where a particular object is in this frame, and where it was in the previous frame, it can make a fairly good guess at where it will be in the next frame. Since a *decompressor* will be able to make the same guess exactly, all that needs to be stored is the difference between what was predicted and what was actually shown. This can make a big difference to the size of a video file containing a lot of motion.

3.4. Interpolation

Finally, compressors can take advantage of the fact that most video is shown at such a high frame rate that the human eye doesn't actually register the details of every frame. The high frame rate is necessary to prevent the picture from looking jerky, but the fine detail of most images isn't especially important. Consequently, instead of storing every frame, a compressor can store every other frame (or every third or fourth frame, depending on the algorithm and the quality desired), and tell the decompressor to *interpolate* the missing frames – that is, to create a frame which is halfway between the frames to either side of it. This frame may not be exactly the same as the original frame it replaces, but it is generally close enough that the difference isn't noticed.

4. The Major Codecs

Every compression technique can be implemented in many different ways, and techniques can be combined in many different ways. This means there are thousands of possible compression algorithms that could be applied to a piece of digital video.

In other words, there are lots of codecs. Some are proprietary; some are available as open standards. But unless you operate a professional digital video production company, you're probably not interested in a codec you have to pay a lot of money to use.

---

[*] Images from http://www.animemusicvideos.org

There are three criteria to pay attention to when comparing codecs: The compression ratio (the extent to which the file size is reduced), the quality of image that is delivered after compression, and the amount of processing power that is needed to decompress the video in real time. There is always a trade-off between these factors, with each codec taking a slightly different approach.

In the web context, the most commonly used codecs are:

4.1. MPEG-4

MPEG-4 is in fact a collection of standards for the design of codecs, so a number of codecs fall under this heading, including codecs by Microsoft and Apple. Since they are all based on the same standard, there is a high degree of interoperability between MPEG-4 codecs. MPEG-4 provides good image quality at relatively low data rates, and works particularly well for video with relatively little motion. Incarnations of the MPEG-4 decompressor are included in Windows Media Player and Apple QuickTime, making the standard widely useable.

4.2. Sörenson

This codec is a standard component of Apple's QuickTime media player, and is generally considered the best codec for use in a Mac environment. Sörenson delivers superior compression to MPEG-4, but is not widely used outside of the Macintosh world because the codec must be downloaded separately in order be used on non-Macs.

4.3. DivX

The first version of the DivX codec was a hacked version of Microsoft's proprietary implementation of MPEG-4. Released and developed on the Internet, it was the first widely-available codec that allowed compression of files in the commonly-used *.avi* format, and became hugely popular as the codec of choice for pirated film downloads. Notwithstanding this somewhat dubious honour, it has kept pace with the development of MPEG-4, and offers superior video quality at equivalent compression. It is also entirely platform-independent and free encoders are available, making it widely usable. The downside of DivX is that it requires a relatively large amount of processing power to decompress, causing video to lag on slower computers.

4.4. H.264

The result of collaboration between the Motion Picture Experts Group and the International Telecommunication Union, this codec is in many ways the natural extension of the MPEG standard (in fact, it is often referred to a "MPEG-4 Level 10"). This codec offers extremely impressive compression, and though its use is not yet as widespread as its predecessor, it looks likely to be the future standard and is, for example, the standard codec used for iPod Video. The downside of this codec from a web perspective is that it has relatively high overheads, meaning small files have a lower effective compression rate than larger ones. In addition, H.264 is a processor-intensive codec, and is not recommended if your target audience has slow computers.

5. Video Formats

The actual format of a video file is becoming less and less important as media players offer wider support of formats. However, each format does have its strong and weak points, and these are worth considering.

5.1. Audio-Video Interleave (AVI)

This format has been part of Windows since 1992, and its longevity can be attributed to its limitations: An AVI file is essentially just a container for media, with very limited formatting. This means that video compressed with virtually any codec can be stored in AVI format. It also means that AVI files are the "lowest common denominator", playable on virtually any hardware.

5.2. QuickTime Movie (MOV)

The Apple standard, QuickTime movies are playable on both MacOS and Windows (though a separate player must be downloaded for Windows). The MOV format supports a wide range of codecs, though Sorenson and MPEG-4 are the favourites, and H.264 is growing fast.

5.3. Motion Picture Experts Group (MPG)

MPG is the format designed to make optimum use of the MPEG-4 codec. In many ways, this format is just a tweaked version of AVI, and there are very few differences in terms of cross-platform support or performance.

5.4. Windows Media Video (WMV)

Windows Media is what it says it is: The standard video format defined by Microsoft for use on Windows platforms. WMV files utilise a proprietary codec based on the MPEG-4 standard, and are easily created using Windows Movie Maker, a tool that ships with Windows XP. This format is therefore the simplest to create for amateurs, but is not supported on non-Windows systems.

5.5. Real Media (RM)

Circa 1998, Real Media was by far the best format for Internet distribution of video. With a proprietary codec that was ahead of its time and a revolutionary streaming mechanism, Real Media was way ahead of its rivals. Unfortunately, development has been slow since then, and Real Media is no longer considered one of the leading formats.

5.6. Macromedia Flash (SWF)

Originally designed as a mechanism for displaying interactive animated elements on a website, Flash now includes its own video codec. While picture quality tends not to be as good as that offered by the other formats, Flash does have the distinct advantage of being designed to run embedded in Web browsers, making it ideal for video that is to be viewed on the site rather than downloaded.

6. Streaming vs Downloading

It is worth considering your delivery mechanism for video. Should the video be embedded in the page, playing while it downloads, or should the video be available as a simple file, to be downloaded and played by the user?

Each option has its good points. The latter option is significantly simpler to implement, less prone to errors, and won't stop playing halfway through if the viewer has a slow connection. On the other hand, embedding a video file means that the viewer doesn't have to leave your page, requires no software except a browser plugin (which is typically downloaded automatically), and can start viewing the video immediately, rather than having to wait for it to finish downloading.

The general practice in the industry is to utilise streaming video, since more and more people are using faster Internet connections which can take advantage of the technology. Ultimately, though, your decision must be based on your particular circumstances.

Of course, there's nothing at all stopping you from offering the video as *both* an embedded stream *and* a file for downloading.

7.   What to Use

Different combinations of format and codec can be effective in different circumstances. The key consideration is always going to be your target audience: What types of computers are they using? Do they have fast Internet connections? Which codecs are they likely to have?

So let's consider some standard format/codec combinations. The five we're going to look at are:

- AVI with DivX

- MPG with Microsoft MPEG-4 Codec

- MOV with H.264

- WMV with Windows Media Video Codec

- Flash Video with Flash codec (FLV)

Comparing the combinations is a matter of creating and viewing files based on the same piece of uncompressed video. The sample video was 60 seconds long, with dimensions of 320x240, a frame rate of 25 frames per second, and a bitrate of 960 kbps. The results are as follows:

Head-to-Head					
Format	Codec	Size	Time	Quality	Compatibility
AVI	DivX	6.41MB	11 sec	Good	Any With Download
MPG	MS MPEG-4	6.69MB	15 sec	Good	Windows / Mac With Download
MOV	H.264	6.50MB	13 sec	Good	Mac / Windows With Download
WMV	Windows Media	6.61MB	11 sec	Good	Windows
SWF	Flash	6.94MB	10 sec	OK	Any (with Browser Plugin)

Figure 3: Head-to-Head Comparison

Unsurprisingly, it is the DivX codec that delivers the best compression, though there is very little in it for a file this small. H.264 came in a close second, hampered by it large overheads. The MPEG-4 codec showed its age somewhat, achieving lower compression, while WMV would probably have performed better on a larger file.

As expected, the Flash video delivered the worst compression and the poorest picture quality, but as previously mentioned, its broad compatibility offers some compensation for this.

A minor consideration is the time it takes to compress the video file. It is worth noting that the MPEG-4 file took 50% longer to compress than the same file using Flash. For an individual file this makes little difference, but if you are going to be creating lots of video files, this might be relevant.

What does this all tell us? Well, it largely depends on your circumstances. If compression is vital for you and your users don't mind making an extra download, go for the AVI/DivX combination. If you want a no-hassle solution for Windows users only, WMV looks best. And if you don't mind losing a bit of picture quality in exchange for the assurance that anyone will be able to view the video, look at Flash.

Once you've made your choice, outputting your video to the right format is easy. eRightSoft provide the excellent **Simplified Universal Player Encoder and Renderer (SUPER)**, a free tool which converts virtually any video format to any other, as well as handling most of the major codecs. It is available at http://erightsoft.com/SUPER.html. (Lucid is in no way affiliated with eRightSoft.)