How it works:
high definition video

In the old days, television was a simple analogue signal. Now it's all zeroes and ones, but how do TVs display clear footage in different resolutions and frame rates?

Next time you slip a DVD into your player and relax in front of your new 1080p HDTV to watch a crystal clear movie, take a moment to consider the digital video manipulation that's going on. After all, your DVD is encoded in one resolution and your HDTV is another, and not an exact multiple at that. How does it display a clear video with minimal image artifacts?

A bit of history

Television in the old days seemed pretty simple: it was analogue, it existed in the air around us and it used cathode ray tubes (CRTs). That was about it really, from the viewer's point of view.

Behind it all though, there was some pretty clever stuff going on – for the time. First, the frame rate was geared to the frequency of the AC power that fed the television. In England and the rest of Europe, which uses the PAL encoding system, the AC frequency was 50Hz, and so the frame rate was set at 25Hz – enough to fool the eye into interpreting a rapid succession of frames as continuous motion. (The SECAM system is similar to PAL, at least as far as this article is concerned.) In the US, which uses the NTSC system (people joked that NTSC stood for 'never twice the same colour'), the AC frequency is 60Hz and so the frame rate was set to 30Hz (29.97Hz to be precise).

That's oversimplifying the situation a little. A television CRT displays the signal as a series of lines from left to right – a process known as scanning – jumping back to the left before scanning the next line. Scanning proceeds not by processing the lines in order, but by skipping lines. The CRT scans all the even lines, jumps back to the top left, and then scans the odd lines, as shown in Figure 1 on p82. This is known as interlacing. The display of all the even lines (or all the odd lines) is known as a field, with two fields per frame. This means that the field display rate is equal to the AC frequency (50Hz in Europe and 60Hz in the
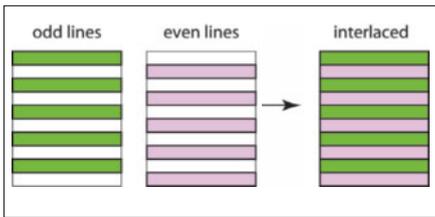
US). For the PAL system, there are 625 lines per frame (only 576 lines are visible, the rest carry other information), and for NTSC there are 525 (of which 486 are visible).

Interlaced video reduces the bandwidth of the television signal by half, and helps smooth out motion (at least on an interlaced TV). The odd field is displayed in 1/25 of a second, and remains on the screen (the phosphors on the screen maintain the image for another 1/25 of a second as they decay) as the even field is displayed. This remains on the screen for the 1/25 of a second that it takes for the next odd field to be displayed, and so on. The fields are taken 1/50 of a second apart. In other words, a single frame isn't broken apart into odd and even frames; the frames are shot separately.

Converting frame rates

Since film is shot at 24 frames per second, it must be converted to another frame rate for viewing on TV – a process called telecine. For Europe, the solution was to speed up the film by 4 per cent and convert the frames of the film into fields (usually by duplicating frames). The audio had to be corrected to ensure it wasn't too high pitched. This meant films ran quicker on television (a 90-minute film is 129,600 frames, which would take 86.24 minutes to show on TV), but this wasn't noticeable.

For the US, the problem was more acute: the frame rate had to be increased from 24 to ▶

Motion compensation

In preparing video for DVD, one way to improve compression is to use motion compensation. This recognises that the set of frames in a sequence are often similar and contain much of the same information. The differences between one frame and the next are minimal, and due to movement of either the camera or an object in the frame.

Motion compensation encodes the video as a reference frame, followed by a set of difference frames that describe the differences from the reference frame. These contain less data, so the compression of the video is higher. If the camera is panning, a later frame could be obtained by shifting the reference frame by a couple of pixels. This may produce better overlap between it and the reference frame, resulting in yet more compression. For video showing lots of motion, this technique isn't effective. ■

▲ **Figure 1: Interlacing, as used by CRT screens.**

▶ 30 frames per second. One simple way would be to duplicate one frame in four. However, this would produce jerky motion and was therefore never used. In fact, the US telecine process (known as 3:2 pulldown) works by adding extra duplicate fields. In essence, four frames (or eight fields) were converted to five frames (10 fields) by duplicating two of the fields. This ensured much smoother motion in the video.

Figure 2 below shows the process. Each of the fields for each frame is shown in different tints of the same colour, with the earlier field as the lighter hue. As you can see, frame three of the video is formed from the earlier field of film frame two, coupled with the later field of film frame three; frame four is formed from the earlier field of the film frame three, coupled with the later field of film frame four.

The computer age
This was all good until computer monitors came along with the ability to display video, especially from DVDs. The biggest issue with computer monitors is that they don't use an interlaced scan system – they use progressive scan. The display is still made up of lines, but the monitor scans them in strict sequence to build up a complete frame, not as an odd field followed by an even field. A PC monitor is also pixel-based – the signal displayed is digital.

The main issue then with displaying video on a computer monitor is that the signal, which is interlaced, needs to be converted to a progressive scan. This is deinterlacing. Each frame must be recreated from two frames, one odd and one even. Since each of the frames in a pair is taken at two slightly different times

(1/50th second apart), if they were simply combined in a single frame, the motion of objects being photographed would be discernible as visible defects. These defects are commonly known as combing, because the scans from odd and even fields wouldn't match up and would be visible as short lines. Figure 3 shows the effect with a box moving from right to left. The top part shows the interlaced version playing on an interlaced scan device; the bottom part shows that simple deinterlacing and playing on a progressive scan device produces a comb-like pattern.

Deinterlacing methods
Deinterlacing isn't just a simple algorithm. Much research has been done to improve the quality of deinterlacing algorithms, but the processing remains time consuming and complex, requiring the display to buffer fields, process them into frames and display them.

There are three key deinterlacing systems: field combination deinterlacing, field extension deinterlacing and motion compensation. The first combines two succeeding frames to form
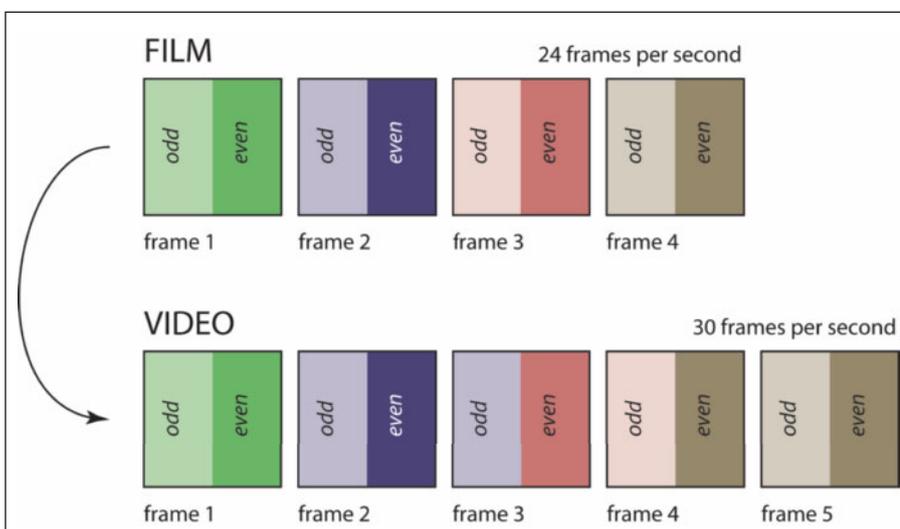
a frame. Since the simple algorithm of weaving the odd and even scan lines in order produces combing effects, there are other techniques, such as blending (averaging out succeeding lines) and selective blending (averaging out succeeding lines for motion only, otherwise weaving them for frames that don't change). The problem with these techniques is that the resulting video tends to look soft or blurred.

The second system, field extension, either views the fields as actual frames (at half-height, essentially) so other techniques such as edge detection or sharpening can come into play, or doubles each line in each field. The problem with the latter method is that stationary objects in the video tend to bob up and down, and the height resolution is halved. This is known as the line-doubling technique.

The final system blends output from all the others to produce the best picture quality possible. The algorithms here try to predict the direction and speed of motion of objects in the video, minimising combing and softening.

All modern TVs (LCDs and plasmas) are progressive scan systems, but there are still problems once we factor HDTVs into the picture. Consider the standard DVD – it contains a video in what's known as 480p format. This is 480 pixels high (and usually 640 pixels wide for the 4:3 aspect ratio), and uses progressive scanning (the 'p' stands for 'progressive'). Some DVDs are 480i, where 'i' stands for 'interlaced'. A normal DVD player can read either type of DVD, decode and decompress, and produce output using interlaced scan for CRT televisions or progressive scan for LCD or plasma boxes. If we connect the output from such a DVD player to a TV via the analogue inputs (composite or S-Video), we'll will get an analogue, interlaced display of the video on the DVD. It won't be completely sharp – it's not digital, after all.

It's better to pipe the video from your DVD into an HDTV digitally using the DVI or HDMI connection – avoid converting to analogue at all costs. The problem here is the

---

Spotlight on… Digital image stabilisation

Another area where real-time video processing comes into play is image stabilisation. In many cameras and camcorders, there is a capability to reduce blurring due to camera shake as the image or video is being taken. Some of these use hardware sensors to detect physical jitter, and floating lenses moved with tiny electromagnets to counter it immediately – the shake is resolved before the camera's sensor records the picture or frame.

In other camcorders, digital image stabilisation occurs as the video is being recorded. The algorithm uses edge detection and other techniques to identify camera shake from frame to frame, and then shifts each frame electronically to counter the jitter.

For this to work, there are a couple of caveats: either the logical border of the video frames needs to be slightly smaller than the actual shot border so there's room to shift individual frames slightly to counter the shake, or the algorithm has to extrapolate pixels that were never shot in order to shift the frame.

The main disadvantage of digital video stabilisation is that it relies on well-defined objects in the field of view. This in turn means you need to shoot in good light conditions. For low light conditions, the exposure times might be longer (leading to more visible camera shake), and there might be more video 'noise' (resulting in blurred edges, and cramping the algorithm's ability to detect shake). ■

▲ **Figure 2: The US Telecine process (3:2 pulldown) increases frame rates by adding duplicate fields.**

difference in resolution. Suppose your HDTV is 720p. That means it has at least 1,280 x 720 pixel resolution for a widescreen display, and uses progressive scan. Considering only the vertical pixel height, we have to convert 480 pixels into 720, unless we want to view the DVD output directly in a tiny window in the middle of the HDTV. We have to stretch the video approximately 1.5 times in size.

Video upscaling

Enter the video upscaler. This device sits somewhere on the path from the DVD player to the HDTV. It can be integrated into either device or can be a separate box altogether, although it's usually part of the DVD player (on the premise that upscaling should be done as close to the source as possible). What the video upscaler does is analyse the pixel values and interpolate the values of extra pixels between them in order to stretch the 480 lines into 720. This is sometimes known as resampling.

The simplest scaling algorithm is bilinear interpolation. This is a simple extension of linear interpolation that you probably learned at school. With linear interpolation, you're trying to find the value of a function at some point given two known values at either side. Essentially, you 'draw' a line between the two values at the end and interpolate along that line the value at the point in question. In the top part of Figure 4, we know the populations of England and Wales in 1811 and 1861, and estimate the population in 1851 using linear interpolation.

Bilinear interpolation moves that algorithm into two dimensions: you know the values of the function (in our case, colour values) at the corners of a square and you want to find the value at some point within the square. Look at the bottom of Figure 4, where you're trying to find the value at x given the values at a, b, c and d. You perform a linear interpolation

across the top (corners a and c, interpolated as ac) and bottom (corners b and d, interpolated as bd) of the square, and linearly interpolate between those two new points (from ac to bd, interpolated as x). Bilinear interpolation is simple to code up, but when applied to images as we stretch them to fit a larger resolution, it causes aliasing artifacts and visual defects.

The process of bilinear interpolation relies on four pixel values to calculate another pixel value. This assumes that the values don't vary wildly across the image (we assume the colour function is fairly continuous with no major breaks), but as we know, images and video frames aren't like that. There are object edges where colour changes dramatically – 'breaks' in the smoothness of colour information.

In these cases, we should use more data points in our interpolation. This has the benefit

> ## "The video upscaler can't create data out of nothing, but it does a good job"
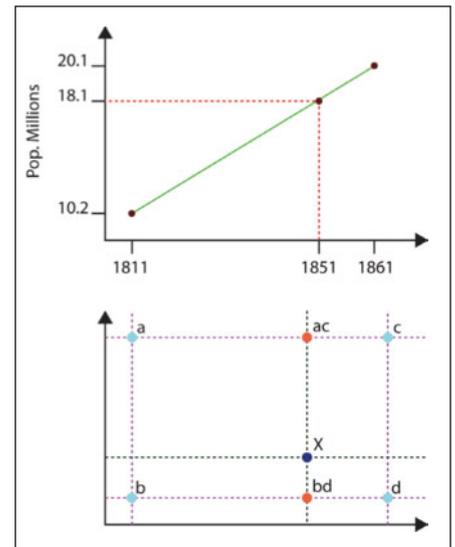
▲ **Figure 4: Bilinear interpolation (below) is a simple extension of linear interpolation (above).**

of coping with edges in the original image, and providing a 'smoother' interpolation. This is normally equivalent to using a quadratic interpolation. For a video upscaler, one of the most popular non-adaptive algorithms is bicubic interpolation, which relies on not the four closest pixels, but the 16 closest. Those pixels closest to the required point have a higher weighting than those further away, but all 16 are involved in the calculation. The algorithm is known as non-adaptive because the whole frame is treated equally; no effort is made to identify fast-changing parts of the scene and other video effects.

Non-adaptive algorithms

Bicubic interpolation has the benefit of a fast calculation (slower than bilinear, but fast enough for video), but isn't the only non-adaptive algorithm. Others include nearest neighbour, spline, sinc and Lanczos. To achieve better visual results, there are many patent-protected scaling or sampling algorithms that detect edges and apply algorithms to them to minimise the visual interpolation defects in those areas. Others try to detect motion of objects in the video over several frames and apply algorithms to make motion smoother. With video, these algorithms have to be written to spot scene changes and not assume that they are fast motion. These algorithms have to be fine-tuned for performance; they do, after all, scale video at 25 or 30fps.

The upscaler can't create detail out of nothing, but it does a good job of converting video without artifacts. For better video, there's nothing for it but to use Blu-ray discs in a Blu-ray player. These are designed to provide 1080p video at 24 frames per second, and will play on a 1080p HDTV without upscaling. **PCP**

Julian M Bucknall has worked for companies ranging from TurboPower to Microsoft, and is now CTO for Developer Express.
feedback@pcplus.co.uk
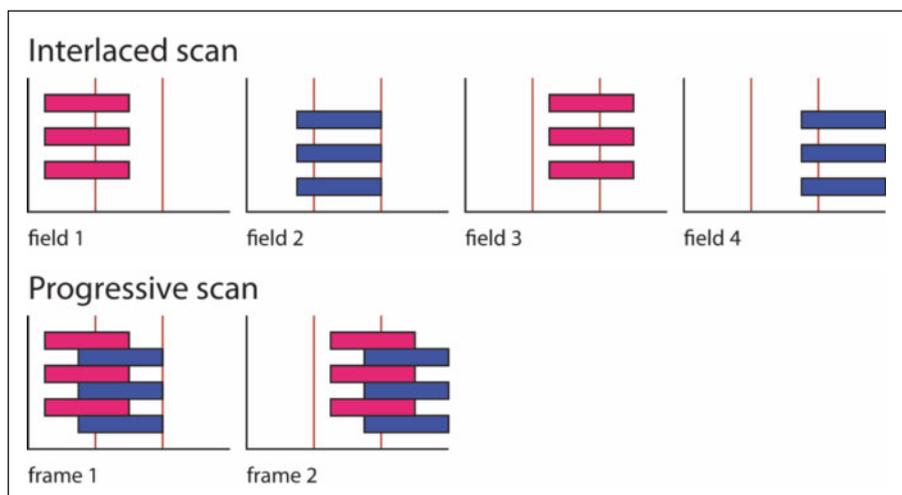
▲ **Figure 3: A box moving from left to right – playback on a progressive scan TV produces a comb effect.**