

Viewing JPEGs on TV – Part 1

T.O.M.S.

[This series of three articles was originally published in Archive magazine in 2004-05 and is included on the AVLine CD by kind permission of Archive Publications. T.O.M.S. have updated the articles (especially Part 3) to reflect recent changes.

Cross-references to other articles in Archive are by issue and page number, e.g. 8.9 p10.

Note that Archive is aimed primarily at users of the RISC operating system, so most of the software titles mentioned will not run under Windows, etc, although possible contenders will be discussed at the end of Part 3.]

Back in Archive 16.5 p34, John Myers asked about the possibility of burning JPEGs onto CD, to run in a DVD player for display on TV.

At the time [2003], hardware existed to enable JPEGs to be loaded directly into a DVD recorder but the high cost made this an impractical option. However, after some false starts, John now uses the Windows application *JASC Paint Shop Photo Album 4*. He burns his JPEGs onto CD in 'Video CD' format (VCD), runs the discs in his DVD player and views the images as sequences of stills.

The VCD display quality was satisfactory but, unsurprisingly, was not up to full DVD standards. Also what John called "shimmer" was evident on the displayed images. We replicated the symptoms using *Microsoft Movie Maker* and burning still images onto a DVD, so the culprit is evidently not the application software, nor the VCD format which John was using. We've investigated the phenomenon and have come up with a probable reason and a working solution to deal with it.

The cause of the 'shimmer'

We deduced that a possible reason for the shimmer was related to the need for any image to be scaled down – or up – in order to fit the

standard, 576 horizontal line format of a PAL TV/video display. (Although PAL is a 625-'line' system, only 576 lines are used for the video display; the remainder carry audio, control data, Teletext pages, etc.)

For example, if we take an image from, say, a typical 1.3 megapixel (MP) digital camera, taken in the landscape format, the resolution will very likely be 1280×960 pixels, with the same 4:3 aspect ratio as a standard TV screen. In this instance, the 960 pixels in the vertical axis are scaled down, by the display circuitry, to fit exactly onto the 576 horizontal line display.

However, the re-scaling ratio will almost certainly not be an *exact* multiple of 1:2, 1:1, 2:1 or so on. In this case, it's $960 \div 576 = 1.667:1$. The difficulty then is that a given horizontal line of pixels won't necessarily fall *exactly* on a particular line on the interlaced TV/video display.

For instance, the calculation for the 481st line of pixels (i.e. the line of pixels immediately above the centreline of the image) would be $481 \times 576 \div 960 = 288.6$. So should that element of the image be displayed on line 288? Or on line 289?

In practice, what we suspect happens is that this line of pixels is first displayed on line 288, then on line 289, then back to 288, and so on, 'jittering' or 'dithering' up and down over two adjacent lines, at an approximate rate of 3-4Hz. The same argument goes for most other horizontal lines of pixels. It's thought the dithering appears to the viewer as 'shimmer'. Why the apparent rate of shift is 3-4Hz is not known, but that need not concern us.

Stopping the shimmer

Following a hunch, we tried scaling down some typical images *prior* to burning another test disc. In the vertical axis, the 960 horizontal lines of pixels were scaled down to 576, to exactly match

the 576 horizontal lines in the PAL TV/video display; the premise being that any one line of pixels would then always be displayed on *exactly* the same horizontal line of the display (numerically), without disturbance.

Proof of the pudding came when the second DVD was played – and gave a rock-solid display, without any hint of the previous shimmering. We then tried burning a VCD – again no evidence of instability. Finally, John repeated the process using *Photo Album 4* and, again, no instability on his VCD. Great.

So the fundamental conclusion is that, in order to forestall any vertical instability, which manifests itself as ‘shimmering’, the source images must first be re-scaled so that the vertical dimension is *exactly* 576 pixels (even a close approximation proves to be unsatisfactory).

Other image sizes

Scaling down the 1280×960 pixel image (1.3MP), but retaining the 4:3 aspect ratio, results in 768×576 pixels, ready for burning onto DVD or VCD. The same theory applies to images from other digital cameras, e.g. 2048×1536 pixels from a 3MP camera or 2832×2128 pixels from a 6MP camera. If we scale them all down to the common 768×576 pixels, we should stop the shimmer, at source.

DVD/VCD standards (PAL)

Unfortunately, it’s not quite as simple as that, albeit for an unrelated reason.

It transpires that the video image format, as recorded onto DVDs or VCDs (PAL), does not retain the exact 4:3 aspect ratio of 768×576 pixels but, for some reason, is slightly ‘squished’, by around 8%, into the PAL/DVD standard of 720×576 pixels. We have no idea why this should be, but we have to live and work with that fact.

But if this ‘squished’ image were to be displayed on a standard 4:3 TV screen, it would be slightly too narrow. So, although we freely admit we’re

not sure of our facts here, we *think* the image is first re-expanded slightly, by around 8% – by the display circuitry – prior to being displayed on the screen. This seems to apply not only to DVDs but also VCDs and the intermediate ‘super’ VCD (SVCD) format.

To summarise so far: Firstly, in order to put a stop to the ‘shimmer’ phenomenon, it seems we must first re-scale all the images so that their vertical resolution is *exactly* 576 pixels, to match that of the standard PAL display.

Secondly, it seems we must also re-scale all (landscape-format) images so that their horizontal dimension is *not* 768 pixels (as we would expect for a 4:3 TV screen aspect ratio), but the narrower 720 pixels. However, the 4:3 aspect ratio is then restored by the display circuitry (we think), before display.

This topic is discussed in more detail in the *Video to DVD* article on the AVLine CD, Part 5, pp5-6. If anyone can offer a definitive explanation, we’d be glad to hear from you!

Other image formats

Although John Myers’s own requirement was particularly to display JPEGs from his digital camera, the suggested procedures can also apply in principle to all bitmap image formats – BMPs, sprites, TIFFs, etc – provided all have an initial aspect ratio of 4:3 before pre-scaling for display. Vector graphics (e.g. drawfiles) can also be processed and displayed but they must first be made into bitmaps using a suitable conversion utility such as *D2Sprite* or, more simply, the Snapshot facility in *Paint* (for RISC OS).

Pre-scaling multiple images

Happily, the requirement to pre-process lots of source images prior to burning a DVD, SVCD or VCD is an ideal candidate for Jochen Lueg’s shareware application *FSI_Batch* (described by Ross McGuinness in 17.6 p73), particularly as the underlying *Change_FSI* will be able to import JPEGs and also the bulk of ‘foreign’ file formats such as BMPs, TIFFs, etc.

However, as the source images may not all have the same pixel resolution (maybe from different cameras or other sources such as a scanner), in this instance we won't be able to use the fixed **Scale** settings discussed in 17.6 p73.

But as an alternative procedure, we can instead use the **Pixels** option (immediately below the **Scale** option, shown in Fig 1) which will do the necessary re-scaling for us – and still in a single pass or batch.

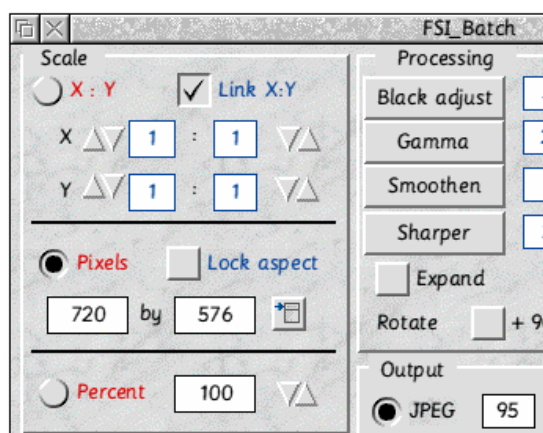


Fig 1 The *FSI_Batch* window (section).

The batch-processing procedure is as follows:

1. Click on **Pixels**.
2. De-select **Lock aspect** (*unticked*).
3. Click on the **X by Y** sub-menu button.
4. Click on the **720*576** option (*v3.6 or later*).
5. Select the output format as **JPEG**.
6. Set the JPEG Q factor to (we suggest) **95**.
7. Load the source images ready for processing.
8. Click on the **Start** button to process the batch.

(Note: If you have an earlier version, the numbers can of course be entered manually. But *FSI_Batch* v3.6 or later, including this option, is available for download from Jochen's website at <http://www.jochenlueg.freeuk.com>).

The reason for de-selecting **Lock aspect** (Item 2) is that we are now *intentionally* overriding the 4:3 aspect ratio of the original images in order to ensure the 'squished' dimensions of 720×576 pixels are attained for the whole batch.

If all has gone to plan, you'll finish up with the batch of source JPEGs having been scaled down to 720×576 pixels which, when burnt onto DVD, SVCD or VCD, will be expanded by the display circuitry (we think) and shown on TV at 4:3, without the troublesome shimmer.

Images in portrait format

If the camera has been rotated through 90° to produce an image in portrait format, it will need to be rotated back through 90° for viewing 'right way up', before being burnt onto DVD or CD.

For JPEGs, we find the best RISC OS software for this job is either *JCut* for single images, or *JClean* for multiple images, as they retain the JPEG format. Both are freely available for download and non-commercial use via David Dacha's website at <http://www.dacha.freeuk.com>

Note however, that if you're using RISC OS v3.6-4.0x (without *Select*), rotating JPEG images may well result in the 'banding' problem discussed in the *JCut* Help file.

This is believed to be a hiccup with the RISC OS *SpriteExtend* module up to RO v4.0x which is fixed in RO v5 and in *Select/Adjust*. An 'unapproved' but entirely effective third-party fix is available for RO 4.0x or earlier (contact T.O.M.S. for details).

Having rotated a JPEG in portrait format through 90°, the aspect ratio then becomes 3:4, e.g. 960×1280 pixels from a 1.3MP camera.

But when re-scaling it, we still need to retain *exactly* 576 pixels in the vertical Y axis in order to avoid the shimmer phenomenon, *and* 'squish' the image by around 8% to allow for the pre-display expansion by the display circuitry.

This works out at 405 pixels wide, so the settings to enter (manually) in the **Pixels** windows in *FSI_Batch* become **405 by 576**.

Post-processing, all portrait-format images will then become 405×576 pixels and, when viewed on the monitor, should appear to be a little 'squished' (Fig 2).

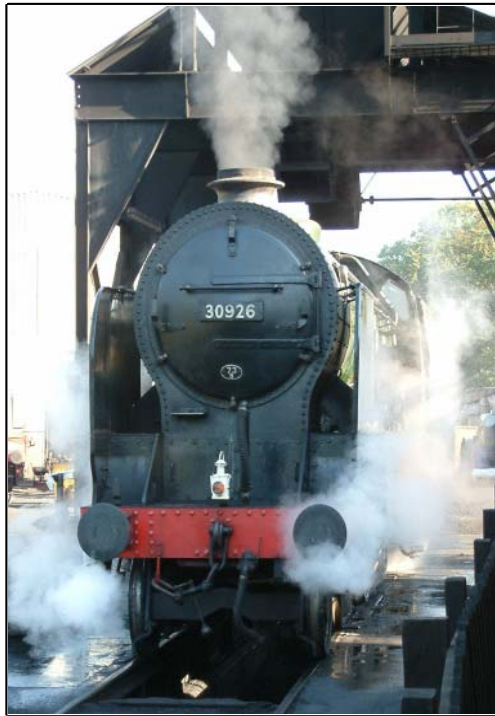


Fig 2 Portrait format – pre-scaled to 405×576 pixels

But by the time they've been re-expanded by the display circuitry and displayed on TV, in the 'vertical letterbox' format (3:4 aspect ratio), they should look OK.

However, unavoidably, when displayed on a standard, 4:3 (landscape) TV display, there will be a black, vertical band to the left and right sides of the image (Fig 3).



Fig 3 Portrait format image on 4:3 TV display

To summarise the pre-scaling procedures, note therefore that the pre-processing must be done in two separate batches:

- *Batch 1* for landscape-format images (720×576 pixels).
- *Batch 2* for portrait-format images (405×576 pixels).

Widescreen TVs

If you display an image (in either 4:3 landscape or 3:4 portrait format) on a widescreen TV, there will of course be a black band either side of the image.

This is unavoidable for portrait-format images. However, it can be very classy to have *landscape*-format images completely filling the display.

To do that, we need not only to re-scale the image so that its X axis fits the wider screen, but also to ensure that the Y axis is still exactly 576 pixels to avoid the shimmer phenomenon.

Assuming the aspect ratio of the widescreen TV display is 16:9, the processed images will therefore need to be 576 pixels in the vertical and – allowing for the 16:9 screen *and* the pre-display expansion by the display circuitry – 960 pixels in the horizontal. We can achieve this by two separate processes.

Firstly, set up *FSI_Batch* to re-scale all the landscape-format JPEGs, initially down (or up) to 960×768 pixels and setting the output to save the processed images as JPEGs.

Now load in all the 4:3 aspect ratio, landscape-format JPEGs to be processed and click on **Start**. On completion, the output directory will contain the batch of processed JPEGs; at this intermediate stage, all scaled to the common size of 960×768 pixels.

That sorts out the image width ('squished' by c. 8%) but still leaves us to sort out the image height to stop the 'shimmer'.

So secondly, load each JPEG – *singly* – into *JCut* which opens the image viewer (Fig 4).

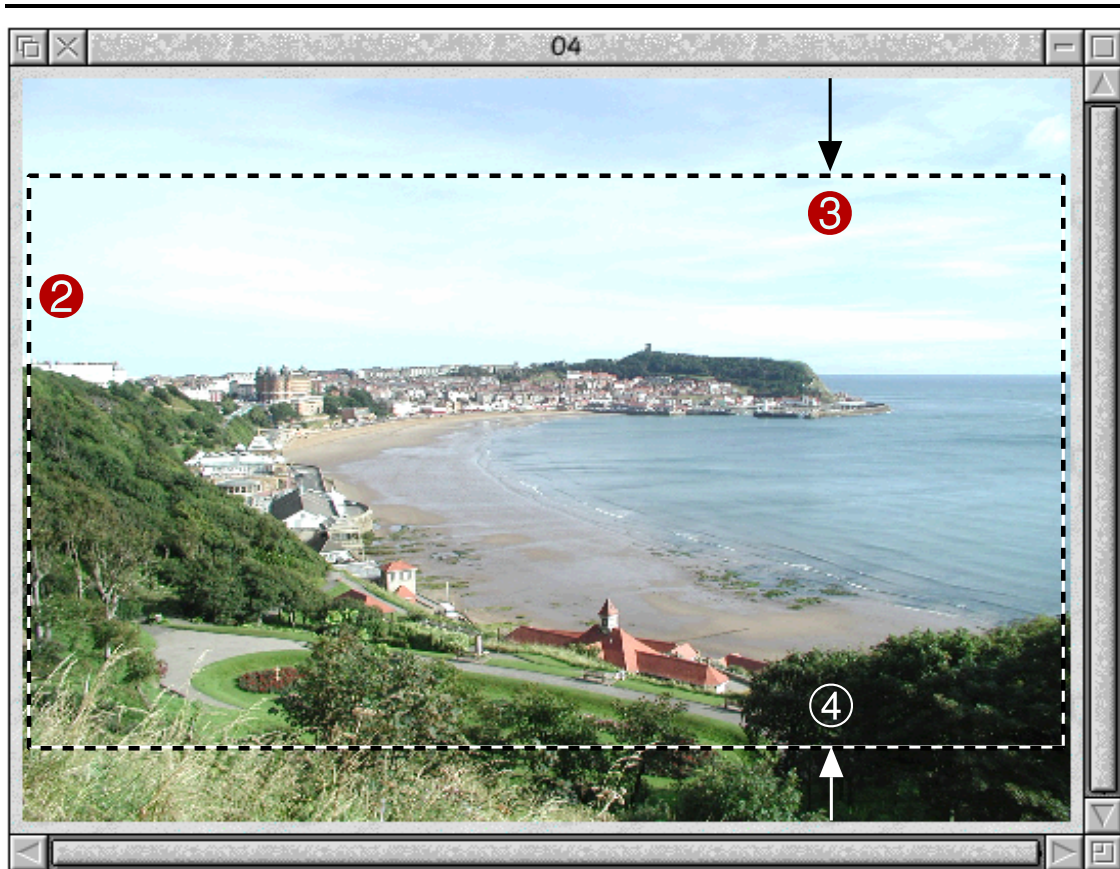


Fig 4 JCut image viewer

Click <menu> over the viewer to open the *JCut* menu and then click on its **Toggle Size** icon to expand it (Fig 5).

Confirm the full image size is correctly shown in the *JCut* menu as **960 by 768** pixels. ①

Initially, the select box in the *JCut* image viewer ② (in pale blue in the original but hatched black/white for clarity in Fig 4) will cover the full image area, i.e. 960×768 pixels.

However, we can now use <adjust-drag> on any of the four corners to move the top line down ③ and/or the bottom line up ④, to 'crop' the image to suit, until the height of the central dark-grey section in Fig 5 ⑤ indicates exactly 576 pixels. ⑥

(With JPEGs, this is quite easy as the crop lines and pixel reading will move in steps of typically 8 or 16 pixels, depending on the source image.)

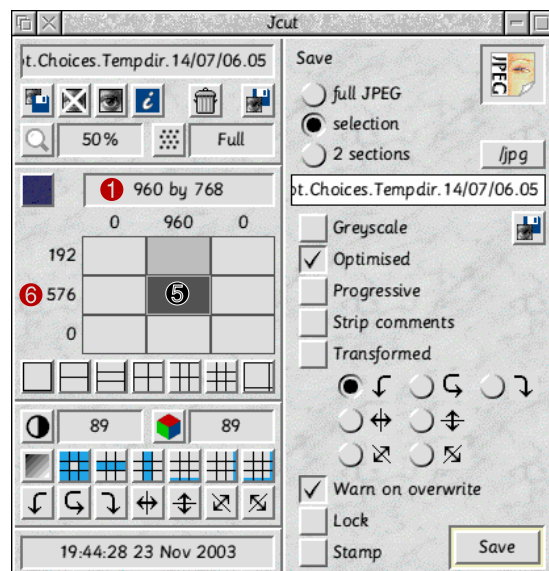


Fig 5 JCut menu

Note that the area now contained within the adjusted select box (Fig 4) represents the cropped image, as it will eventually be displayed on the widescreen TV (16:9) although, for the moment, it will appear 'squished'.

Click on the **selection** button (Fig 5), give the image a name and drag the **Save** icon to its destination. The 960×576 pixel image is now ready for burning onto DVD or VCD.

This second procedure needs to be repeated – individually – for each 'widescreen' image, but this gives freedom to position the top and bottom crop lines in *JCut*, to suit the particular image, whilst ensuring its height is set exactly to the required 576 pixels.

T.O.M.S., toms@ndirect.co.uk
