



Stabilization and Compression

David Hesselbom

1 Introduction

Video compression usually works by computing and compressing deltas between frames, i.e. the difference between one frame and the next, rather than storing each individual frame as a compressed image. The more similar a frame is to the previous one, the less data is required in order to describe the delta. Stabilization works by compensating for unwanted camera movements, effectively making each frame in the video more similar to the one before it.

1.1 The difference between two images

The difference (diff) between two pixels can be computed by taking the absolute value of the difference between the color of those pixels. An image diff can be created by computing the pixel diff for every pixel in two different images. If a pixel has the exact same color in both images, that pixel will be black in the diff (black being RGB (0,0,0)), and the more different the color is in the two images, the brighter the pixel will be in the diff. The more similar the images are, the darker the diff will be.

Figure 1 shows a single frame from the unstabilized video. Figure 2 shows an image diff between that one frame and the next. Because the two frames are mostly very similar images, the diff is mostly black. The outlines of the tree trunk can be seen as green lines, as the camera has panned horizontally, causing these pixels to go from a dark grey in the first frame to a bright green in the second, or vice versa. Similarly, the outlines of the girl's hair can be seen as purple lines where pixels have gone from green to white (adding purple (255,0,255) to green (0,255,0) results in white (255,255,255) in the RGB color space).

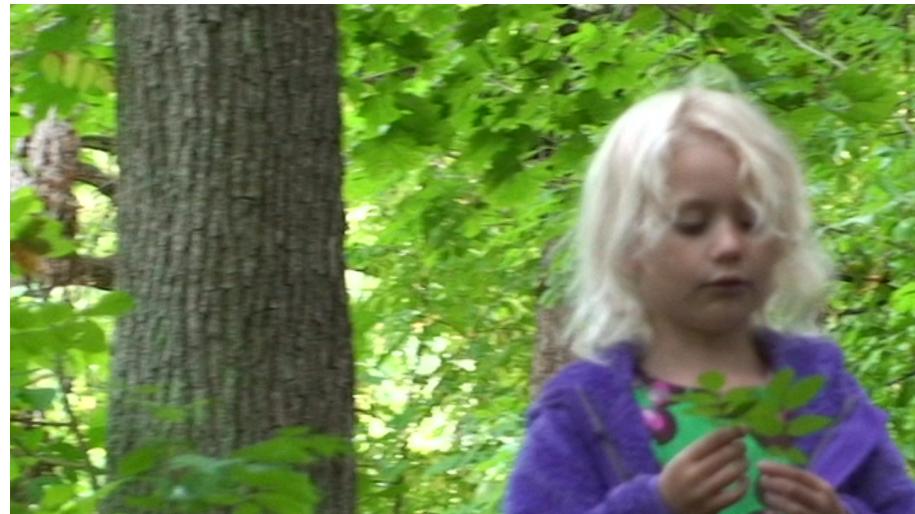


Figure 1: A frame from the unstabilized video.

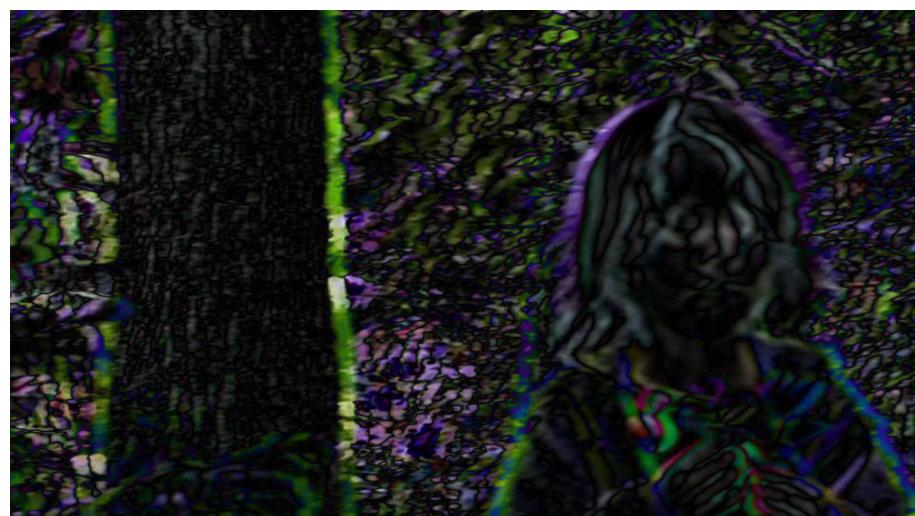


Figure 2: The diff between figure 1 and the following frame in the unstabilized video.

1.2 Stabilization

Most modern video encoders have some form of motion compensation built-in. H.264 employs local (but not global) motion compensation, and the same goes for its future successor, HEVC. This compensation can often reduce the delta between two frames to a few vectors describing how elements in the frame have moved in the next. The encoders do not, however, compensate for unwanted movement in the video. Professional video

is typically very stable to begin with, but the same cannot be said for amateur video, captured using a camcorder or a cell phone. Vidhance stabilization keeps the image stable by compensating for unwanted movement moving the *frame* of the video, keeping the object being viewed in place. This makes for a more comfortable viewer experience and takes some of the workload off the video encoder. It does, however, require a little headroom for a cropping, which in turn requires the sensor of the camera to have a higher resolution than the resulting video.

Figure 3 is the result of superimposing the same two frames as in figure 2, but translating one of them by (23,-2) to compensate for unwanted camera movement. The diff is almost completely black aside from the girl's hand. This is not due to unwanted camera movement; the girl moved her hand. The area where the two frames overlap is now only 1224x716, which explains why some overhead is necessary in order for the stabilized video to be kept at a certain resolution.



Figure 3: The diff between the same two frames, manually "stabilized" by translating one of the frames.

1.3 x264

x264 is an H.264 encoder. It allows for 51 different quality settings where 0 is lossless and 51 is the most lossy, and where increasing the quality by 6 doubles the bitrate. Encoding a video at quality 23 (default) thus results in a file twice as large as encoding one at quality 29. The FFmpeg and x264 encoding guide¹ suggests qualities 18-28 to be a "sane range" and that anything below 18 should be considered visually lossless. There are also 8 different encoding presets, which give the encoder an idea of how long you're willing to wait for the video to be encoded. Forcing the encoder to work faster means less visual quality and/or larger files.

1. <http://trac.ffmpeg.org/wiki/x264EncodingGuide>

1.4 Constant bitrate

When encoding video at a constant bitrate, the quality of the video is limited by the amount of data allowed for each frame, so the delta must be reduced to that amount of data, even if this means losing a lot of information. By stabilizing the video before compressing it, the delta will contain less information because the stabilized frame is more similar to the preceding one. This translates into higher video quality with stabilization than without for a given bitrate.

1.5 Variable bitrate

When encoding video at a variable bitrate, the bitrate of the video is governed by the quality setting, so in order to maintain a given quality score, the delta may become very complex. A complex delta requires more data to store it without losing information. By stabilizing the video before compressing it, the delta will be smaller in size because the stabilized frame is more similar to the previous frame, which translates into a smaller video with stabilization than without for a given quality setting.

2 Procedure

Several video clips were shot using a handheld Sony HD camcorder at 1440x1080i. After deinterlacing, one stabilized and one non-stabilized version of each clip, upscaled to 1920x1080 and then cropped to 1280x720, were converted into H.264 using 4 of the 8 encoder presets available (ultrafast, faster, medium, and veryslow), and each of the 52 available quality settings.

3 Results and analysis

In all of our experiments, the stabilized video results in smaller files than does the unstabilized video for the lower quality settings, especially for very low quality settings. As a rule, the lower the quality setting, the greater the benefit of stabilization. In the sane range of qualities, the benefit is rarely less than 5%. Figures 4 and 5 show the file size reduction for 2 different video files at all 4 encoder presets, while figure 6 shows the file size reduction with the "veryslow" encoder preset for all videos in the experiment.

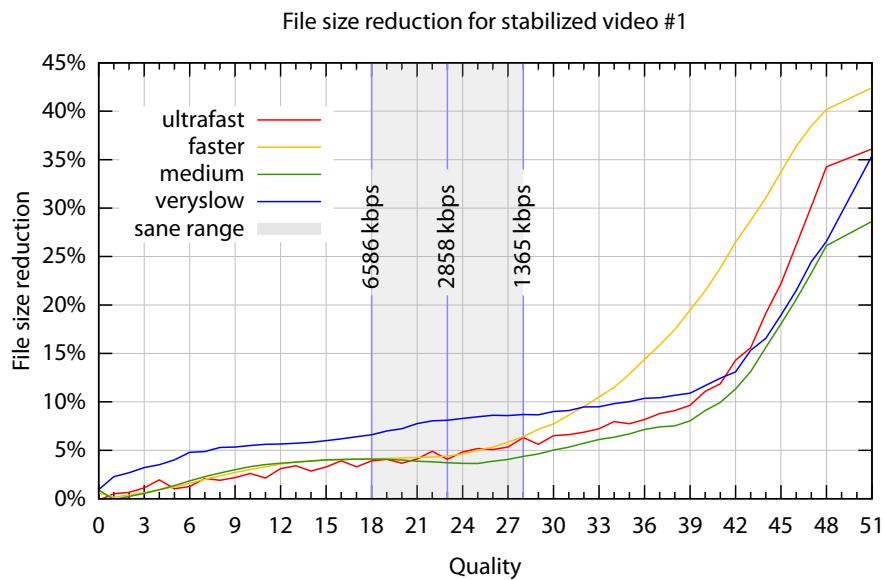


Figure 4: A plot demonstrating how file size reduction generally increases as video quality decreases for the video in figures 1, 2, and 3. The bitrates indicated are of the stabilized video for qualities 18, 23 and 28 at the "veryslow" preset.

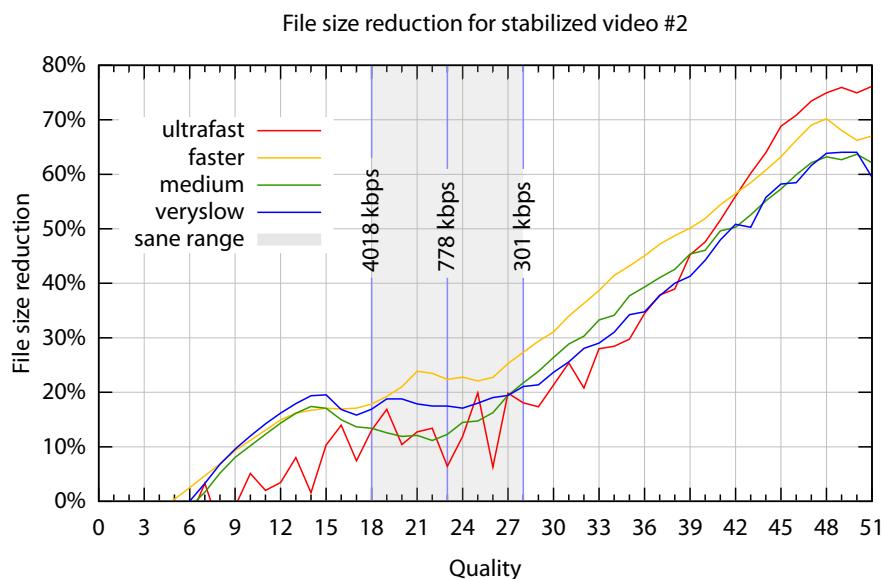


Figure 5: A similar plot demonstrating the same tendency in another video. In the sane range of qualities, the file size for this video is reduced by 16.9%-21.1%.

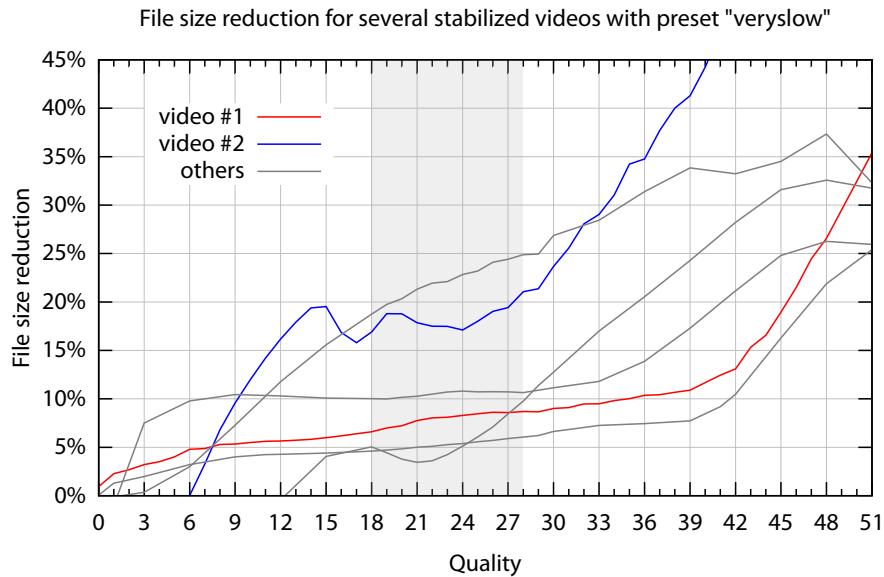


Figure 6: A plot demonstrating the same tendency in all of the videos used in the experiment, for only the "veryslow" encoder preset.

For very high quality video, the benefit of stabilized video is sometimes insignificant, and in rare cases, significantly negative – the stabilized video is actually larger than the original video in these cases. In our experiment, this only happened at very high quality settings, outside the "sane range". Part of the explanation could be that stabilization causes a misalignment of the blocky compression artifact pattern already in the video. These blocks can hardly be spotted without extreme saturation and brightness/contrast adjustments (as in figure 7), but at very high (above visually lossless) quality settings, they will still be "seen" by the encoder. Without stabilization, this blocky pattern stays in the same place throughout the video. If this misalignment can explain the contradictory results, uncompressed video should not suffer from the same problem, and should see a greater improvement across the full range.

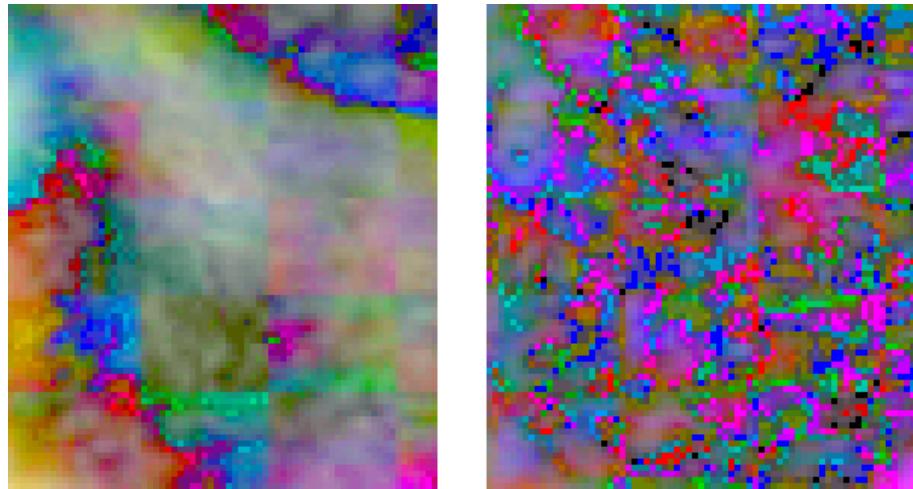


Figure 7: The images in figure 2 (left) and figure 3 (right), auto-leveled, saturated, cropped from (901, 444), and zoomed to 400%. In the unstabilized video (left), the compression artifact blocks are clearly visible after extreme saturation and brightness/contrast adjustments. The block size, 21.3x16 pixels, is the result of upscaling the 16x16 pixel blocks in the input video from 4:3 to 16:9. In the stabilized video (right), the blocks are still there, but barely visible, because the compression artifact patterns in the two overlaid frames are no longer aligned.

4 Conclusions

- Stabilizing video before compressing it can significantly reduce file size for medium and low quality video. This translates into less bandwidth usage, ultimately increasing network performance.
- The initial study shows size reduction reliable above 5% in the range of standard encoding qualities, and typically in the range of 5-20%.
- Stabilization requires the camera sensor to have a resolution that is slightly higher than the resulting video.
- Stabilization may result in increased file size at very high quality encoding settings for video input that already contains compression artifacts.
- The tests have been made with the compressed video that could be obtained as output from the camera. It is a reasonable assumption that the size reduction could be even greater if implemented earlier in the chain.
- The benefit of stabilizing uncompressed video is assumed to be great enough to motivate further investigation, and experiments should also be extended to include more types of video content in order to reach more conclusive and detailed results.