Verification of Normalization Processing for VQEG Video Test Data

18May98

Norman Franzen (norman.franzen@tek.com)
Jerry Lu (jerry.lu@tek.com)
Mihir Ravel (mihir.ravel@tek.com)

Information Technologies Group,
Tektronix Measurement Div.,
Beaverton, OR USA

1. Purpose

The Video Quality Experts Group (VQEG) will be characterizing the performance of various Objective Video Quality Models as described in the VQEG OBJECTIVE VIDEO QUALITY MODEL TEST. Below, we refer to this document as simply the Test Plan. The plan calls for the normalization of the Source and Processed video test sequences prior to subjective viewer testing and objective model testing. This note summarizes various algorithms that can be used to verify the results of the normalization processing.

2. Un-normalized parameters

In section 2.0 of the Test Plan, several categories of differences between Source and Processed sequences were **disallowed**:

- picture cropping > 10 pixels
- chroma/luma differential timing
- picture jitter
- spatial scaling

It is understood that in the domain of mixed analogue and digital video processing some of these conditions may occur, but that for this experiment these conditions will not be detected and normalized.

3. Normalized parameters and procedure

The Test Plan calls for the following conditions that **will be detected and normalized**:

- temporal misalignment, (i.e. frame offset between Source and Processed sequences)
- horz./vert. spatial shift
- chroma/luma gain and level

We interpret this in more detail as follows:

- chroma and luma spatial realignment will be applied to the Y, Cb, Cr channels independently. The spatial realignment step will be done first. Details will be given below.
- chroma/luma gain and level will be corrected in a second step using a cross-correlation process, as specified below, but other changes in saturation or hue will not be corrected.
- cropping and spatial misalignments will be assumed to be global, i.e. constant throughout the sequence. Dropped frames will not be allowed. Any remaining misalignments will be ignored.

The Test Plan specifies that the first and last second of each video sequence will consist of a specific alignment pattern described in the APPENDIX below. This pattern is constructed to make alignment easy and robust and is largely transparent to any HRC. The required amount of normalization will be estimated from the frames containing this alignment pattern, and the normalizations will be applied uniformly across the full length of the sequences.

4.Verification of normalization

The results of the normalization processing can be verified using the following algorithms. Other algorithms exist that may be more efficient, but the following methods are relatively simple to understand and use well known image processing techniques.

4.1 Detection of Temporal Alignment

Using the initial alignment stripe segment, the frame offset can be easily determined by locating the sequence transition at the end of the striped segment. This will occur at either frame 30 (525/60), or frame 25 (625/50). Frame drops can be detected by counting frames between the end of the first stripe and the beginning of the final stripe. A more sophisticated approach using cross-correlation of the two sequences can also be used, but we find that this added complexity is usually not necessary.

4.2 Detection of Spatial Alignment

The magnitude of spatial misalignment (Δx, Δy) can be determined by successively applying phase correlation methods to the stripe area to determine integer pixel shifts and then remaining sub-pixel shifts. The general technique is described in reference [1]. Below we summarize a specific implementation of this approach that is robust and conceptually simple, but not necessarily the most efficient method.

Detection of spatial alignment between a Source and Processed sequences is done on the 1-second length of striped frames after the two sequences are aligned temporally. The verification of spatial alignment can be generally described as a three-step process:

1. First, the nearest whole-pixel offset is computed by using a phase correlation method over the region of the frame containing the alignment stripe pattern(Reference 1, Digital Video Processing, by A. M. Tekalp, Prentice-Hall, 1995).
2. Then, in the region of the frame containing the alignment pattern, both the Source and Processed frames are upsampled or interpolated by a factor of 16 in each dimension. (i.e. the width and height of the resulting image regions are 16 times as large as the originals).
3. The last step is a refinement step, where the same phase correlation method as in step 1 is applied again over the upsampled Source and Processed frame regions, which brings the estimated spatial offset to the required subpixel accuracy (1/16 of a pixel).

Details of the Method

The phase correlation method uses the linear phase characteristic of the Source-Processed correlation function in the frequency domain to detect the translation vector.

$$\delta(n_x, n_y) = \mathcal{F}^{-1}\left(\frac{F_{tst}(\omega_x, \omega_y) F^*_{src}(\omega_x, \omega_y)}{\left| F_{tst}(\omega_x, \omega_y) F^*_{src}(\omega_x, \omega_y) \right|} \right)$$

where F_{src} and F_{tst} are the discrete time Fourier transforms of the source and distorted frames. $\mathbf{F}^{-1}$ is the inverse Fourier transform. The Fourier transform functions are represented by DFT (FFT) coefficients in implementation. Therefore, offset can only be estimated up to the nearest pixel. The real function $\delta(n_x, n_y)$, where n_x and n_y are integer offset values, is searched for the location of the peak. The coordinates of the peak is the estimated translation vector of the Processed image with respect to the Source image.

In the refinement step (the last step), the above phase correlation is computed again, except all the Fourier transforms and the δ function in the equation are computed in the upsampled grid points, that is

$$\delta_{(16)}(n_{x(16)}, n_{y(16)}) = \mathcal{F}^{-1}\left(\frac{F_{tst(16)}(\omega_x, \omega_y) F^{*}_{src(16)}(\omega_x, \omega_y)}{\left|F_{tst(16)}(\omega_x, \omega_y) F^{*}_{src(16)}(\omega_x, \omega_y)\right|}\right)$$

In principle, the upsampling in the spatial domain can be performed by reconstruction of a continuous 2-D signal (surface) from the discrete frame images, and then by re-sampling the continuous signal with a sample rate 16 times as dense as the original signal in each dimension. A detailed mathematical description of this operation can be found in Fundamentals of Digital Image Processing, by A. K. Jain, Prentice-Hall, 1989 (p 89, Equation 4.16).

To avoid the complexity of computing the full frame, a window is applied in the striped area for evaluating phase correlation function. The window is 256x64 for the integer-pixel accuracy first step, and 256x32 for the sub-pixel refinement step. The two corners of the 256x64 window are at pixel locations (122, 76) and (377, 139). The 256x32 window is located at (122, 88) and (377, 119). The Hamming window (see Digital Signal Processing, by Oppenheim and Schafer) is used to weight the windows in both the integer-pixel and subpixel cases. This is to eliminate the border effect in estimating the shift vector.

This detection method can be applied independently to each Y, Cb, Cr channel.

4.3 Detection of Gain/Level

Gain and level can be computed for each Y, Cb, Cr channel independently using the following least squares based method. The calculations are to be performed in the region of the frame containing alignment pattern. In simplified form the defining equations are: (X = Source, Y = Processed, Z = Corrected)

Gain = (Rxy - Rx•Ry)/(Rxx - Rx•Rx), where Rxy = (1/N) Σ X(i)•Y(i), Rx = (1/N) Σ X(i), etc.

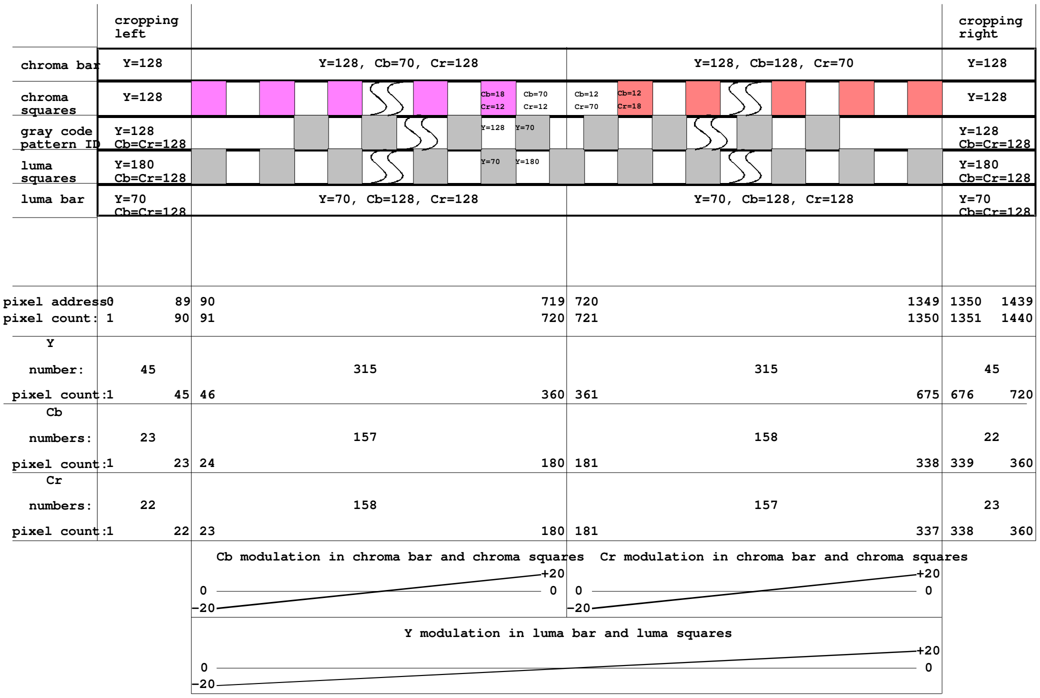
Level = Ry - Gain • Rx

5. References

[1] "Digital Video Processing", by A. Murat Tekalp, Prentice Hall 1995

APPENDIX: Structure of Alignment Stripe Pattern

Detailed description of Alignment bar for Pattern A:



	cropping left			cropping right
chroma bar	Y=128	Y=128, Cb=70, Cr=128	Y=128, Cb=128, Cr=70	Y=128
chroma squares	Y=128	Cb=18 Cr=12 Cb=70 Cr=12	Cb=12 Cr=70 Cb=12 Cr=18	Y=128
gray code pattern ID	Y=128 Cb=Cr=128	Y=128 Y=70		Y=128 Cb=Cr=128
luma squares	Y=180 Cb=Cr=128	Y=70 Y=180		Y=180 Cb=Cr=128
luma bar	Y=70 Cb=Cr=128	Y=70, Cb=128, Cr=128	Y=70, Cb=128, Cr=128	Y=70 Cb=Cr=128
pixel address	0 89	90 719	720 1349	1350 1439
pixel count:	1 90	91 720	721 1350	1351 1440
Y number:	45	315	315	45
pixel count:	1 45	46 360	361 675	676 720
Cb numbers:	23	157	158	22
pixel count:	1 23	24 180	181 338	339 360
Cr numbers:	22	158	157	23
pixel count:	1 22	23 180	181 337	338 360

Typical image:

Further details of test pattern: