Fast algorithm for HDR color conversion

Andrey Norkin

Netflix Inc.
Winchester Circle 100
Los Gatos, CA 95032, USA
anorkin@netflix.com

***Abstract**:* The paper addresses a problem of perceptual artifacts that appear in Y'CbCr non-linear luminance 4:2:0 HDR video. A computationally inexpensive method is proposed for converting the 4:4:4 HDR video to Y'CbCr 4:2:0 non-constant luminance format. The method removes artifacts in areas with saturated colors. The approach obtains results in one step, improving the average linear light PSNR by 2.15 dB and tPSNR by 2.00 dB on the investigated videos.

1. Introduction

High Dynamic Range (HDR) is considered one of the major coming improvements in the quality of the television pictures. To support luminance levels in the region of 0 to 10 000 cd/m^2, ST.2084 defines a highly non-linear transfer function [1] to achieve quantization that is perceptually unnoticable. To facilitate transport and broadcasting of HDR videos, a Digital Entertaiment Content Ecosystems (DECE) has adopted a specification, called HDR10 [2] that specifies HEVC Main10 encoding, ST.2084 transfer function [1], BT.2020 color space [3], Y'CbCr 4:2:0 non-constant luminance color format [3], and some optional supplemental enhancement information (SEI) messages.

Several MPEG proposals [4], [5], [6] identified a subjective quality problem with HDR *Y'*CbCr non-constant luminance 4:2:0 color format. The problem is caused by a steep slope of the opto-electrical transfer function (OETF) in the low-luminance range and the color transform, which make color components with low values have significant impact on *Y'*, Cb, and Cr values. This causes artifacts in colors at the color gamut boundaries .

A solution to the problem described above was proposed in [7]. The solution was to downsample and upsample chroma components and then iterate over different values of luma to choose the value of the luma sample that results in the linear luminance closest to the one of the original signal. A bisection method was applied, which enabled getting the result in at most ten iterations for each luma sample for a 10-bit signal. The iterations require computing a transfer function and applying color transform and can therefore be rather slow even if a transfer function is represented as a look-up table.

This paper proposes a closed form solution that calculates the value of a luma sample in one step. The algorithm removes artifacts in saturated colors, improving subjective and objective video quality. The method achieves higher average PSNR and tPSNR [8] compared to straighforward downsampling of chroma. Compared to [7], the proposed method has somewhat higher average PSNR and lower tPSNR. Subjectively, the results from both methods look very similar, removing artifacts caused by chroma subsampling.

The rest of the paper is organized as follows. Section 2 gives the overview of the problem with color subsampling. Section 3 describes the proposed method for color conversion. Section 4 presents the subjective and objective results, and Section 5 estimates the computational complexity. Finally, Section 6 concludes the paper.

2. Problem with HDR 4:2:0 color conversion

2.1. Y'CbCr 4:2:0 color conversion as in HDR10

A non-constant luminance approach has been adopted in the recent HDR10 standard proposed by DECE [2] and is described in BT.2020 [1]. HDR10 standard specifies operations in the receiver side but the encoding process is partly determined by it.

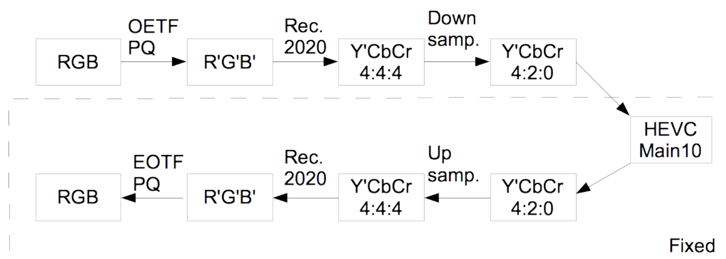


Figure 1. Flowchart of HDR10 processing chain. The dashed box marks part defined by HDR10 specification.

The processing with non-constant luminance [3] can be summarized as follows. The opto-electrical transfer function (OETF), which is the inverse of electro-optical transfer function (EOTF), is applied separately to each of R, G, and B components of the original linear light signal. Then, Y'CbCr signal is obtained by applying the color transformation. Chroma components Cb and Cr are downsampled by two vertically and horizontally. The decoding and display is the inverse of this process. The inverse of the ST.2048 [1] transfer function that transforms linear light to transfer function domain is shown below.

$$PQ\_TF(L) = \left(\frac{c_1 + c_2 L^{m_1}}{1 + c_3 L^{m_1}} \right)^{m_2}; \quad m_1 = 0.1593017578125, \tag{1}$$

$$m_2 = 78.84375, \quad c_1 = 0.8359375, \quad c_2 = 18.8515625, \quad c_3 = 18.6875.$$

The shape of the OETF function reflects the fact that the human visual system is more sensitive to changes in luminance when the luminance level is low. Therefore, an OETF, such as the inverse of ST.2084, allocates more codewords (and uses smaller quantization steps) for low luminance samples.

Components R', G' and B' are obtained by applying OETF to the linear light R, G, and B components separately. Then, the non-constant luminance Y′CbCr values according to BT.2020 are obtained as follows.

$$\mathrm{Y}' = 0.2627\ \mathrm{R}' + 0.6780\ \mathrm{G}' + 0.0593\ \mathrm{B}' \tag{2}$$

$$\mathrm{Cb} = (\mathrm{B}' - \mathrm{Y}') / 1.8814; \ \mathrm{Cr} = (\mathrm{R}' - \mathrm{Y}') / 1.4746;$$

2.2. Subjective quality problems

It has been reported in [4], [5], [6] that subsampling chroma components in the non-constant luminance color format can cause significant variations in colors that are close to color gamut boundaries. This shift looks like artifacts (or details) that were not present in the original linear light signal. Figures 2 and 3 demonstrate these subjective quality problems. The artifacts are intensity variations that are not in the original image (see the areas around the sparks in the FireEater sequence (Fig. 2(b)) and the blue shirt and red tent in the Market sequence (Fig. 3 (b)).

(a) Original (b) Obtained from Y′CbCr 4:2:0

Figure 2. FireEater sequence. (a) Original, (b) Obtained from Y'CbCr 4:2:0

(a) Original (b) Obtained from Y'CbCr 4:2:0

Figure 3. Market sequence. (a) Original, (b) Obtained from Y'CbCr 4:2:0

FireEater and Market are HDR sequences with luminance up to 4000 nits. To produce Figures 2 and 3, "exposures" of video were obtained by clipping the values above and below chosen thresholds. In Figures 2 and 3, pictures on the left are the original RGB pictures and pictures on the right are reconstructed from Y'CbCr 4:2:0, non-constant luminance. It should be noted that FireEater and Market are BT.709 sequences [9]. BT.709 color primaries and color transform coefficients were used to model the case when colors are close to color gamut boundaries using the same conditions as in MPEG HDR and WCG Call for Evidence [8].

It was suggested in [5] that the artifacts are caused by large derivative of the OETF function in the range close to the zero (see Fig. 4) and the fact that OETF is applied to each color component separately. If a component has values close to zero, while other components have higher values (which is true for colors close to gamut boundaries), a small intensity component has disproportionally high contribution to the resulting Y'CbCr signal (derived as in [9] or [3]). Hence, small variations in this component values result in significantly different values of Y', Cb, and Cr, although linear light RGB values are similar. When 4:2:0 subsampling is applied, chroma values are averaged but luma values remain significantly different. After the inverse transform and EOTF, these colors are reconstructed to significantly different values (see Fig. 2(b) and Fig. 3(b)).

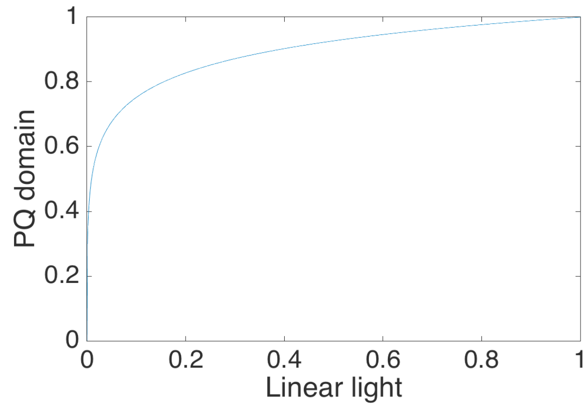


Figure 4. Inverse of ST.2084. Function has steep slope at values close to 0.

3. Algorithm description

The following algorithm has been proposed to mitigate the problem described in Section 2.2. The proposed solution enables fast calculations done in one step.

First, downsampled chroma is obtained. The downsampled chroma can be obtained directly as shown in the flowchart in Fig. 1. Alternatively, an approach described in [7] may be applied based on performing downsampling in linear RGB domain followed by the OETF and color transform. Then, chroma is upsampled back by applying a chosen upsampling filter. The algorithm estimates Y', Cb, and Cr values such that reconstruction to the linear light RGB produces values similar to those of the original image or video. To reduce the dimensionality of the problem, the value of Y is obtained while keeping Cb and Cr the same as after the downscaling step.

The algorithm estimates luma value $Y'(x, y)$ that results in the $RGB_{new}(x, y)$ pixel closest to the original linear light $RGB_{org}(x, y)$ pixel in the Euclidean distance sense, where x and y are horizontal and vertical positions of the sample respectively. The distance between two RGB is measured as follows:

$$D = (R_{new}(x,\mathrm{y}) - R_{org}(x,\mathrm{y}))^2 + (G_{new}(x,\mathrm{y}) - G_{org}(x,\mathrm{y}))^2 + (B_{new}(x,\mathrm{y}) - B_{org}(x,\mathrm{y}))^2 \quad (3)$$

In a more general case, we can also weight the importance of each color component R, G and B with a weighting factor w_X, where X corresponds to a color component, i.e. w_R, w_G and w_B. After omitting pixel coordinates for simpler notation and denoting EOTF as f, the cost function is as follows:

$$D = w_R\, (R_{new} - R_{org})^2 + w_G\, (G_{new} - G_{org})^2 + w_B\, (B_{new} - B_{org})^2. \quad (4)$$

or

$$D = w_R\, (f(R'_{new}) - f(R'_{org}))^2 + w_G\, (f(G'_{new}) - f(G'_{org}))^2 + w_B\, (f(B'_{new}) - f(B'_{org}))^2. \quad (5)$$

The transfer function domain values R', G' and B' can be obtained from Y'CbCr by applying an inverse color transform, which depends on the color gamut and in case of Y'CbCr and BT.709 and BT.2020 has the following form:

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{pmatrix} \begin{pmatrix} Y' \\ Cb \\ Cr \end{pmatrix} \quad (6)$$

For an EOTF with a somewhat complex formula (such as ST.2084) finding a closed form solution minimizing the cost function D (5) may be difficult. It will be shown in Section 3.1 that if EOTF is approximated with truncated Taylor series (the first or second degree polynomials), a closed form solution to the cost function minimization can be found.

3.1. Linear approximation of EOTF

In order to obtain a closed form solution for estimating Y', the EOTF $f(\mathrm{X})$ is approximated with a first degree polynomial, i.e.

$$f(\mathrm{X}_i + \Delta) = f(\mathrm{X}_i) + f'(\mathrm{X}_i)\,\Delta\,, \tag{7}$$

where $f'(\mathrm{X}_i)$ is the value of the derivative of the $f(\mathrm{X})$ with respect to X at point X_i. Substituting (7) into (5), the cost function is approximated as follows:

$$D = \mathrm{w_R}\,(f'(R'_{\mathrm{org}})\,\Delta_\mathrm{R})^2 + \mathrm{w_G}\,(f'(G'_{\mathrm{org}})\,\Delta_\mathrm{G})^2 + \mathrm{w_B}\,(f'(B'_{\mathrm{org}})\,\Delta_\mathrm{B})^2. \tag{8}$$

Then, we substitute Δ_R in (9) with $(a_{1,1}\,\mathrm{Y}'_{\mathrm{new}} + e_\mathrm{R})$, do similar substitutions for Δ_G and Δ_B, differentiate cost function D with respect to Y', set it equal to zero, and solve the resulting equation with respect to Y'. The resulting solution is as follows. First, we calculate:

$$\begin{aligned} e_\mathrm{R} &= -\mathrm{Y}'_{\mathrm{org}}\,a_{1,1} + (\mathrm{Cb_{new}} - \mathrm{Cb_{org}})\,a_{1,2} + (\mathrm{Cr_{new}} - \mathrm{Cr_{org}})\,a_{1,3}\,, \\ e_\mathrm{G} &= -\mathrm{Y}'_{\mathrm{org}}\,a_{2,1} + (\mathrm{Cb_{new}} - \mathrm{Cb_{org}})\,a_{2,2} + (\mathrm{Cr_{new}} - \mathrm{Cr_{org}})\,a_{2,3}\,, \\ e_\mathrm{B} &= -\mathrm{Y}'_{\mathrm{org}}\,a_{3,1} + (\mathrm{Cb_{new}} - \mathrm{Cb_{org}})\,a_{3,2} + (\mathrm{Cr_{new}} - \mathrm{Cr_{org}})\,a_{3,3}\,. \end{aligned} \tag{9}$$

The value of Y' is equal to:

$$Y'_{new} = -\frac{w_R f'(R'_{org})^2 e_R a_{1,1} + w_G f'(G'_{org})^2 e_G a_{2,1} + w_B f'(B'_{org})^2 e_B a_{3,1}}{w_R f'(R'_{org})^2 a_{1,1}^2 + w_G f'(G'_{org})^2 a_{2,1}^2 + w_B f'(B'_{org})^2 a_{3,1}^2} \tag{10}$$

Provided $a_{1,1} = a_{2,1} = a_{3,1} = 1$, as in BT.709 and BT.2020, the expression simplifies to

$$Y'_{new} = -\frac{w_R f'(R'_{org})^2 e_R + w_G f'(G'_{org})^2 e_G + w_B f'(B'_{org})^2 e_B}{w_R f'(R'_{org})^2 + w_G f'(G'_{org})^2 + w_B f'(B'_{org})^2} \tag{11}$$

or, if all the weights are also set equal to 1, the Y' can be found as follows:

$$Y'_{new} = -\frac{f'(R'_{org})^2 e_R + f'(G'_{org})^2 e_G + f'(B'_{org})^2 e_B}{f'(R'_{org})^2 + f'(G'_{org})^2 + f'(B'_{org})^2}\,. \tag{12}$$

Note that values of EOTF derivative in power of two $f'(\mathrm{X})^2$ can be pre-computed and stored in a look-up table. Video with bit depth of 10 requires a table with 1024 entries.

One can notice that the method can work with various transfer functions, including ST.2084 or BT.1886 [10]. The EOTF derivative can be obtained either by differentiating the EOTF function or approximating it numerically, for example using the definition of a derivative (i.e. dividing the change in the EOTF value by the change in the argument). The method can also be applied to different color spaces. Weights $\mathrm{w_R}$, $\mathrm{w_G}$, and $\mathrm{w_B}$ in expressions (4)-(16) can be set equal to one or chosen based on the desired precision or importance of each component.

3.2. Second degree approximation of EOTF

The EOTF can also be approximated using a second degree polynomial

$$f(\mathrm{X}_i + \Delta) = f(\mathrm{X}_i) + f'(\mathrm{X}_i)\,\Delta + f''(\mathrm{X}_i)\,\Delta^2 / 2. \tag{13}$$

The derivations are done in a similar way to the linear approximation case and are omitted for brevity. The solution, assuming $a_{1,1} = a_{2,1} = a_{3,1} = 1$, looks as follows.

$$\begin{aligned} e_{\mathrm{R}} &= -\,\mathrm{Y'}_{\mathrm{org}} + (\mathrm{Cb}_{\mathrm{new}} - \mathrm{Cb}_{\mathrm{org}})\, a_{1,2} + (\mathrm{Cr}_{\mathrm{new}} - \mathrm{Cr}_{\mathrm{org}})\, a_{1,3}\,, \\ e_{\mathrm{G}} &= -\,\mathrm{Y'}_{\mathrm{org}} + (\mathrm{Cb}_{\mathrm{new}} - \mathrm{Cb}_{\mathrm{org}})\, a_{2,2} + (\mathrm{Cr}_{\mathrm{new}} - \mathrm{Cr}_{\mathrm{org}})\, a_{2,3}\,, \\ e_{\mathrm{B}} &= -\,\mathrm{Y'}_{\mathrm{org}} + (\mathrm{Cb}_{\mathrm{new}} - \mathrm{Cb}_{\mathrm{org}})\, a_{3,2} + (\mathrm{Cr}_{\mathrm{new}} - \mathrm{Cr}_{\mathrm{org}})\, a_{3,3}\,. \end{aligned} \tag{14}$$

Then

$$\begin{aligned} \mathrm{T}_{3,\mathrm{X}} &= f''(\mathrm{X})^2, \\ \mathrm{T}_{2,\mathrm{X}} &= 3 f'(\mathrm{X}) f''(\mathrm{X}) + 3\, f''(\mathrm{X})^2\, e_{\mathrm{X}}, \\ \mathrm{T}_{1,\mathrm{X}} &= 2 f'(\mathrm{X})^2 + 6 f'(\mathrm{X}) f''(\mathrm{X})\, e_{\mathrm{X}} + 3\, f''(\mathrm{X})^2 (e_{\mathrm{X}})^2\,, \\ \mathrm{T}_{0,\mathrm{X}} &= 2\, f'(\mathrm{X})^2\, e_{\mathrm{X}} + 3\, f'(\mathrm{X})\, f''(\mathrm{X})\, (e_{\mathrm{X}})^2 + f''(\mathrm{X})^2 (e_{\mathrm{X}})^3, \end{aligned} \tag{15}$$

where X stands for R, G, and B, and $f'(\mathrm{X})$ stands for the first derivative $f'(R'_{\mathrm{org}})$, $f'(G'_{\mathrm{org}})$ or $f'(B'_{\mathrm{org}})$ and $f''(\mathrm{X})$ stands for the second derivative $f''(R'_{\mathrm{org}})$, $f''(G'_{\mathrm{org}})$ or $f''(B'_{\mathrm{org}})$. In order to minimize the cost function D, the cubic equation below needs to be solved with respect to $\mathrm{Y'}_{\mathrm{new}}$

$$\begin{aligned} &(w_{\mathrm{R}}\, \mathrm{T}_{3,\mathrm{R}} + w_{\mathrm{G}}\, \mathrm{T}_{3,\mathrm{G}} + w_{\mathrm{B}}\, \mathrm{T}_{3,\mathrm{B}})\, (\mathrm{Y}_{\mathrm{new}})^3 + (w_{\mathrm{R}}\, \mathrm{T}_{2,\mathrm{R}} + w_{\mathrm{G}}\, \mathrm{T}_{2,\mathrm{G}} + w_{\mathrm{B}}\, \mathrm{T}_{2,\mathrm{B}})\, (\mathrm{Y}_{\mathrm{new}})^2 + \\ &(w_{\mathrm{R}}\, \mathrm{T}_{1,\mathrm{R}} + w_{\mathrm{G}}\, \mathrm{T}_{1,\mathrm{G}} + w_{\mathrm{B}}\, \mathrm{T}_{1,\mathrm{B}})\, \mathrm{Y}_{\mathrm{new}} + (w_{\mathrm{R}}\, \mathrm{T}_{0,\mathrm{R}} + w_{\mathrm{G}}\, \mathrm{T}_{0,\mathrm{G}} + w_{\mathrm{B}}\, \mathrm{T}_{0,\mathrm{B}}) = 0. \end{aligned} \tag{16}$$

The cubic equation has either one or three real roots. In case of three real roots, the minimum is achieved in either the root having the largest or the smallest value (since cost function D is quadratic with a positive coefficient at the quadratic term). The values of cost function D are then calculated for both roots and the root resulting in a smaller value is chosen as $\mathrm{Y'}_{\mathrm{new}}$. If equation (16) has only one real root, it is worth considering the real part of the complex roots pair as a solution. Small variations and inaccuracies in approximation of the EOTF derivative may result in (16) turning to having only one root whereas the real minimum is located close to the real part of the complex roots pair.

4. Experimental results

Two algorithms described above have been implemented in the HDRTools software package [11] used in the MPEG HDR and WCG Call for Evidence [8]. The algorithms have also been compared to the luma micro-grading algorithm [7], which was implemented by the authors of [7] in the same software package. The objective results have also been obtained. The algorithms were run on sequences FireEater, Market, and Tibul in a BT.709 container. These sequences are 1920x1080p sequences with peak luminance of 4000 cd/m^2 that were used in MPEG HDR and WCG CfE [8]. The conditions from the CfE were used (except compression part). The experiments have used an updated version of the HDRTools software. In simulations, [-2 16 54 -4]/64 and [-4 36 36 -4]/64 filters were used for vertical and horizontal upsampling, respectively.

Figures 5 and 6 demonstrate the effect of the linear approximation algorithm, with weights w_{R}, w_{G} and w_{B} equal to 1, on the artifacts in FireEater and Market respectively. One can see that the picture resulting from the proposed algorithm is much more closer visually to the original than the results of the straightforward chroma downsampling. Figure 7 compares the proposed algorithm to the micro-grading [7] on a zoomed part of Market sequence. One can see some small residual artifacts in pink color on the output of [7], whereas the proposed algorithm closely resembles the original. The artifacts in [7] can be attributed to the fact that the algorithm minimizes error in linear luminance, which may in some (rare) cases cause shift in color. On the contrary, the proposed algorithm minimizes error in all three color components, which helps avoiding color artifacts. One should notice, however, that on most of the observed content the results of micro-grading [7] and the proposed algorithm are nearly identical visually.

The objective results are provided in Tables 1 - 4 below. The values in the tables represent PSNR in the linear light domain and tPSNR metric (as in MPEG HDR CfP [8]). tPSNR involves transforming RGB input to XYZ color space and averaging the output of two transfer functions, ST.2084 and Philips, and then calculating PSNR for X, Y, and Z components. It can be observed from Tables 1-4 that the average PSNR increases with more than 2 dB compared to straightforward downsampling of chroma components. A 2 dB improvement is also seen on tPSNR measure. Compared to the luma micro-grading approach from [7] the proposed method yields a 0.34dB higher average PSNR for linear approximation and 0.78dB higher for the second degree polynomial approximation of the EOTF. The tPSNR metric is 1.8 dB lower than for approach in [7] on average. One can see that the second degree approximation method results in a slightly better PSNR (65.96 dB vs 65.52 dB) than the linear approximation method with smaller difference in tPSNR.

What is more important, both proposed methods significantly improve the subjective quality of the videos removing the perceptual artifacts. Another observation is that both linear and square approximation approaches, as well as approach from [7] have slight low-pass filtering effect in saturated colors, likely due to fitting luma to the low-pass filtered chroma. However, this effect is minor compared to subjective improvement when removing the artifacts. Moreover, since the algorithms produce smoother luma than the direct chroma downsampling, gains in the subsequent compression of the resulting video can be expected. This is, however, a topic of further investigation.

5. Computational complexity

One can see from (9), (11), (15), and (16) that the linear approximation of the proposed method is less complex than the second degree approximation, in which roots of a cubic equation have to be found. Considering the performance/complexity trade-off, linear approximation is more suitable for real-time systems. Calculations of $(f\,'(X))^2$ can be approximated as a look-up table with 1024 entries for values of R', G', or B' in a 10-bit video. The approach is applicable to any differentiable EOTF including ST.2084.

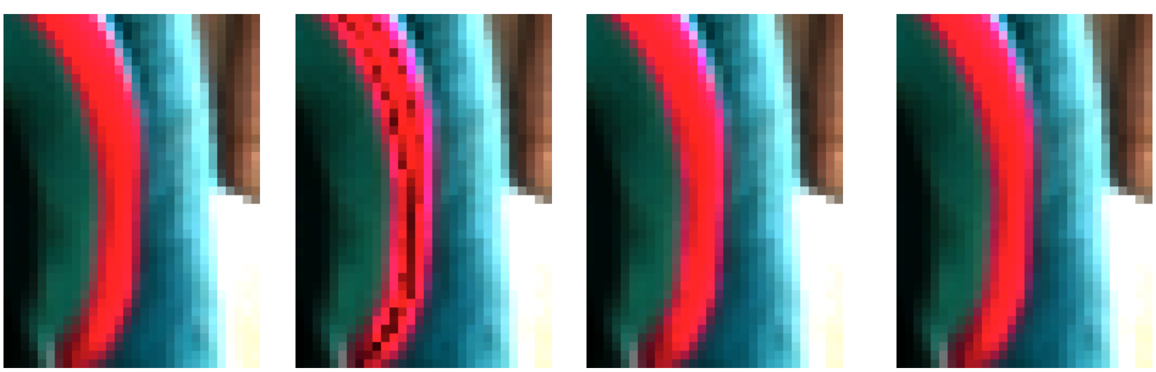
To compare the proposed linear approximation method with the worst-case complexity of the micro-grading approach [7], a rough number of operations after obtaining the upsampled chroma was estimated (see Table 5). The number of operations required for color space conversion, down- and up-sampling of chroma components is not included because it depends on the choice of up- and donwsampling filters and chroma sample positions. For example, when chroma is collocated with (0,0) luma position, as in HDR10 [2], the upsampling step can be omitted. Instead, chroma is low-pass filtered, the proposed algorithm is applied, and then chroma is decimated.

Original | Direct subsampling | Proposed (linear approx.)

Figure 5. Sequence FireEater. Comparison of the proposed algorithm with direct subsampling of chroma

Original | Direct subsampling | Proposed (linear approx.)

Figure 6. Sequence Market. Comparison of the proposed algorithm with direct subsampling of chroma



Original | Direct subsampling | Micro-grading [7] | Proposed (linear approx.)

Figure 7. Sequence Market (zoomed in). Comparison of the proposed algorithm with direct subsampling of chroma and micro-grading [7]

Table 1. Direct downsampling of chroma components

Sequence	PSNR-R	PSNR-G	PSNR-B	PSNR	tPSNR-X	tPSNR-Y	tPSNR-Z	tPSNR -XYZ
FireEater	50.80	71.20	67.45	63.15	54.46	57.10	54.73	55.28
Market	46.35	58.65	48.98	51.33	47.21	49.31	44.59	46.61
Tibul	62.51	82.86	81.56	75.64	50.25	52.20	59.69	52.55
Total	**53.22**	**70.90**	**66.00**	**63.37**	**50.64**	**52.87**	**53.00**	**51.48**

Table 2. Results of luminance micro-grading [7]

Sequence	PSNR-R	PSNR-G	PSNR-B	PSNR	tPSNR-X	tPSNR-Y	tPSNR-Z	tPSNR -XYZ
FireEater	57.92	68.95	66.79	64.55	64.34	70.91	53.20	57.58
Market	48.29	59.87	49.81	52.66	55.43	69.38	45.69	50.00
Tibul	71.36	82.69	80.94	78.33	62.43	66.94	54.45	58.33
Total	**59.19**	**70.50**	**65.85**	**65.18**	**60.73**	**69.08**	**51.11**	**55.31**

Table 3. Proposed algorithm (linear approximation of EOTF)

Sequence	PSNR-R	PSNR-G	PSNR-B	PSNR	tPSNR-X	tPSNR-Y	tPSNR-Z	tPSNR -XYZ
FireEater	65.89	64.76	66.72	65.79	68.24	61.81	51.74	56.01
Market	49.15	55.33	50.82	51.77	55.57	52.56	44.55	48.39
Tibul	77.17	78.94	80.91	79.01	64.28	61.04	52.03	56.04
Total	**64.07**	**66.34**	**66.15**	**65.52**	**62.70**	**58.47**	**49.44**	**53.48**

Table 4. Proposed algorithm (second degree polynomial approximation of EOTF)

Sequence	PSNR-R	PSNR-G	PSNR-B	PSNR	tPSNR-X	tPSNR-Y	tPSNR-Z	tPSNR -XYZ
FireEater	70.83	63.86	66.11	66.93	67.80	61.45	51.64	55.88
Market	49.34	55.40	51.07	51.94	55.67	53.24	44.75	48.65
Tibul	77.28	78.84	80.87	79.00	64.38	61.10	52.06	56.07
Total	**65.81**	**66.04**	**66.02**	**65.96**	**62.61**	**58.60**	**49.48**	**53.54**

Table 5. Comparison of number of operations per luma sample in micro-grading [7] worst case and proposed linear approximation approach (for luma bit-depth of 10).

Algorithm	Adds	Mults	Divs	Table look-ups, TF or TF deriv. sq.	Compari-sons	Shifts (divs by 2)
Micro-grading [7]	65	39	0	30	10	10
Proposed (lin. approx.)	16	9	1	3	0	0

The proposed method uses a closed-form solution to find the value of Y' in one iteration. On the contrary, the micro-grading approach [7] requires up to 10 iterations for a 10-bit video, which includes obtaining R'G'B' values, applying EOTF (can be approximated with a look-up table), and calculating linear light luminance. Average number of operations per sample in micro-grading [7] can be decreased relative to the worst case complexity estimated in Table 5 by computing tighter initial bounds to get the result in fewer iterations, which, however, would add additional operations to the worst-case complexity. Proposed linear approximation algorithm spends fixed number of operations for each sample irrespective of the input. If the proposed algorithm uses weights for color components as in (11), the number of multiplications per sample in Table 5 should be increased by 6. The proposed linear approximation approach has good complexity/quality trade-off and a constant number of operations per sample, which makes it well suited for real-time systems.

6. Conclusions

An approach has been proposed that efficiently removes color artifacts in saturated colors of HDR video that appear in non-constant luminance Y'CbCr 4:2:0 color subsampling. The approach approximating EOTF with linear function performs better than direct downsampling of chroma, resulting in 2.15 dB improvement in average PSNR calculated on linear light R, G, and B components. The improvement in tPSNR measure is 2.00 dB. The proposed approach yields slightly higher average PSNR values than the luma micro-grading approach proposed in [7], while showing smaller improvement on the tPSNR metric. The proposed approach performs calculations in one iteration unlike the approach in [7], which requires up to 10 iterations to yeld the value of Y'.

These features of the proposed approach, namely good perceptual quality and objective metrics performance at low computational complexity, make it useful for 4:2:0 color subsampling in non-constant luminance HDR systems, such as the HDR10 format [2].

7. Acknowledgements

The author thanks Technicolor for providing sequences FireEater, Market, and Tibul, Alexis Tourapis (Apple) for providing the HDRTools software package, and authors of [7] for implementing their approach in HDRTools, which helped obtaining their results.

References

[1] SMPTE ST.2084: "High Dynamic Range Electro-Optical Transfer Function of Mastering Reference Displays"

[2] DECE, "Common File Format & Media Formats Specification version 2.1", Digital Entertainment Content Ecosystem (DECE).

[3] Recommendation ITU-R BT.2020: "Parameter values for ultra-high definition television systems for production and international programme exchange".

[4] E. François, P. Lopez, Y. Olivier, "About using a BT.2020 container for BT.709 content", ISO/IEC (MPEG) document m35255, Sapporo, July 2014

[5] J. Stessen, R. Nijland, R. Brondijk, R. Goris, "Chromaticity Based Color Signals for Wide Color Gamut and High Dynamic Range", ISO/IEC (MPEG) document m35065, Strasbourg, France, Oct. 2014.

[6] J. Ström, "Investigation of HDR color subsampling", ISO/IEC (MPEG) document m35841, Geneva, Switzerland, Feb. 2015.

[7] J. Ström, J. Samuelsson, M. Pettersson, K. Andersson, P. Wennersten, R. Sjöberg, "Ericsson's response to CfE for HDR and WCG", ISO/IEC (MPEG) document m36184, Warsaw, Poland, June 2015.

[8] Call for Evidence (CfE) for HDR and WCG Video Coding, ISO/IEC (MPEG), http://mpeg.chiariglione.org/standards/exploration/high-dynamic-range-and-wide-colour-gamut-content-distribution/call-evidence.

[9] Recommendation ITU-R BT.709: "Parameter values for the HDTV standards for production and international programme exchange".

[10] Recommendation ITU-R BT.1886: "Reference electro-optical transfer function for flat panel displays used in HDTV studio production".

[11] HDRTools software package (Apple). Alexis M. Tourapis.