https://doi.org/10.2352/ISSN.2470-1173.2021.16.COLOR-221


# 3D-LUT Optimization for High Dynamic Range and Wide Color Gamut Color Processing

*S. Andriani, A. Zabot, G. Calvagno, JD Vandenberg*

## Abstract

*3D-LUTs are widely used in cinematography to map one gamut into another or to provide different moods to the images via artistic color transformations. Most of the time, these transformations are computed off-line and their sparse representations stored as 3D-LUTs into digital cameras or on-set devices. In this way, the director and the on-set crew can see a preview of the final results of the color processing while shooting. Unfortunately, these kind of devices have strong hardware constraints, so the 3D-LUTs shall be as small as possible, but always generating artefact-free images. While for the SDR viewing devices this condition is guaranteed by the dimension 33×33×33, for the new HDR and WCG displays much larger and not feasible 3D-LUTs are needed to generate acceptable images. In this work, the uniform lattice constrain of the 3D-LUT has been removed. Therefore, the position of the vertices can be optimized by minimizing the color error introduced by the sparse representation. The proposed approach has shown to be very effective in reducing the color error for a given 3D-LUT size, or the size for a given error.*

## Introduction

In motion picture and television industries images are always heavily color processed to represent the artistic intent of the director and to adapt them to the viewing conditions. The final look of the footage is usually so different from what it is captured by the camera that the on-set crew needs some simple and fast solution to process the images in real-time to verify that everything captured on the set looks as desired. Three-dimensional Look-Up-Tables (3D-LUTs) have been the viable solution. Professional cameras and on-set devices can upload 3D-LUTs and run them in real-time to show a preview on external monitor, but the hardware implementation has always limited the dimension of those 3D-LUTs. Therefore, a trade-off between memory consumption and accuracy in the color representation is a major concern during the system design.

In the past years, almost all the displays or televisions in commerce were Standard Dynamic Range (SDR) with the maximum luminance of around 100 nits and BT.709 gamut [1]. Cinema theatres had half maximum luminance and a slighter bigger gamut i.e., the DCI-P3 [2]. In these viewing conditions, it was proved by experience and several visual tests that 3D-LUTs having size $33 \times 33 \times 33$ were large enough to ensure artefact-free images in almost all situations and a reasonable memory requirement. For this reason, the $33 \times 33 \times 33$ size has become the golden-number in SDR. Sometimes smaller 3D-LUTs have been used to reduce the hardware complexity in entry-level on-set devices or for secondary monitor outputs of the camera, but this solution was accepted only when the images were intended for a preview rather than a distribution purpose.

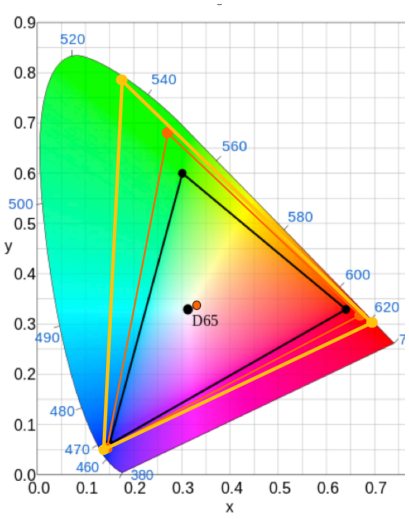



***Figure 1.*** *Gamut comparison between ITU-R BT.709 (black line), BT.2020 (yellow line) and DCI-P3 (orange line).*

Nowadays, the newest television and cinema projector can do much more than the SDR. The maximum luminance can go easily up to 500-600 nits for consumer and even up to 4000 nits for professional displays, the gamut has become much wider and the new BT.2100 standard has been introduced [3]. Fig. 1 shows the difference between the BT.709 and BT.2100 standard gamuts, it is clear that what was sufficient for the SDR may no longer be enough for the HDR standard. Vandenberg *et al* in [4] proved that even with the most accurate 3D-LUT interpolation (i.e., tetrahedral) on average a $49 \times 49 \times 49$ size should be used to avoid visual artefacts, i.e., in HDR the memory requirement for an error-free 3D-LUT should be more than three times larger than an equivalent 3D-LUT in SDR. Unfortunately, at the moment, no camera or on-set device can afford this complexity.

Another problem of uniform 3D-LUTs is shown in Fig. 2 where smaller 3D-LUTs sometimes provide better results than larger ones. This is clearly a huge disadvantage because it could happen that the cost and the effort to increase the size of the 3D-LUT is not reflected into a more accurate color representation.

The idea of using non-uniform 3D-LUTs has been initially proposed and used in printing to convert the image gamut to the printer gamut in an efficient way. In [5], Kang explained how to construct an unequal spaced 3D-LUT for printing by using the sequential linear interpolation approach, then he proved the effectiveness of the method even if the characteristics of the function

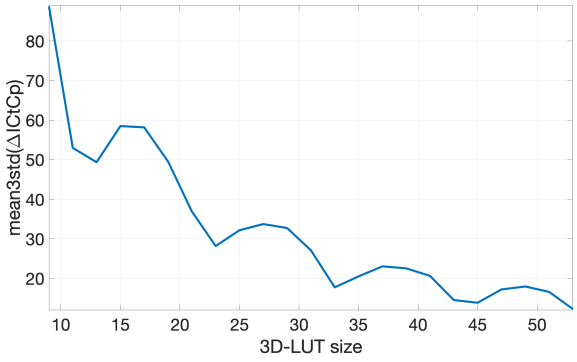



***Figure 2.*** *Uniform 3D-LUTs results for dark images.*

are not known in advance. Monga and Bala in [6] proposed two ways to optimize a 3D-LUT, i.e., by removing the uniform lattice constrain and by allowing small errors on the values stored on the vertices if this reduces the overall errors on the adjacent cubes. Furthermore, in [5] it is explained a simplification on how to simplify interpolation in a non-uniform lattice, and the approach is the same as the non-uniform quantization, i.e., instead of dealing with non-uniform cube sizes and linear input, a distortion is applied to the image before the interpolation and the 3D-LUT is maintained uniform. This approach requires three additional 1D-LUT (one for each color channel) but makes the interpolation much easier.

The aim of this research is to find a way to optimize 3D-LUTs by distributing non-uniformly its vertices to reduce the overall error at a given LUT size or to reduce the LUT size under a maximum error constrain. Unfortunately, the equation behind the color transformation is, most of the time, unknown and the optimization process must use, as reference, a large 3D-LUT having dimension $65 \times 65 \times 65$ or larger. Because of that, the resulting 3D-LUT is only a sub-optimal solution of the problem, but even under this constrain non-uniform optimized 3D-LUTs are the solution for fast and efficient color transformation in HDR.

## Color difference measure in high dynamic range

For High Dynamic Range and the BT.2100 viewing conditions the color difference metrics developed for the BT.709 strongly deviate from the human perception and they are no more reliable. For this reason, new color spaces and metrics have been proposed in the last years, and the most promising combination is the ICtCp color space [7] and the $DeltaIC_tC_p$ color differences [8].

The color transformation from RGB to ICtCp color space is computed as follows:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \frac{1}{4096} \begin{bmatrix} 1688 & 2146 & 262 \\ 683 & 2951 & 462 \\ 99 & 309 & 3688 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

Then, PQ-non-linearity [9] is applied in order to map the optical signal on a display:

$$L'M'S' = EOTF_{PQ}^{-1}(LMS) \tag{2}$$

Finally, $IC_tCp$ coordinates are derived from the $L'M'S'$ values:

$$\begin{bmatrix} I \\ Ct \\ Cp \end{bmatrix} = \frac{1}{4096} \begin{bmatrix} 2048 & 2048 & 0 \\ 6610 & -13613 & 7003 \\ 17933 & -17390 & 543 \end{bmatrix} \begin{bmatrix} L' \\ M' \\ S' \end{bmatrix} \tag{3}$$

If the Hybrid Log Gamma (HLG) curve [10] is used in place of the PQ-curve the procedure remains valid and only the matrix coefficients of the transformation are different.

Once the images are in the ICtCp color space, the $\Delta IC_tC_p$ metric is calculated as follows:

$$\Delta IC_tC_p = 720 \cdot \sqrt{(\Delta I)^2 + 0.25 \cdot (\Delta C_T)^2 + (\Delta C_P)^2} \tag{4}$$

where 720 is the normalization factor to ensure that the unity of the metric is the JND.

If the metric is used for monitor or printer calibration purpose, the knowledge of the $\Delta IC_tC_p$ value is enough; however to compare color transformation on real images a meaningful way to compress the color metric calculated for every pixel of the image into a unique value should be found. The use of the average of the $\Delta IC_tC_p$ is not the best solution because many of the pixels are above the average values and the spread of the error is unknown. A better way is to use the *mean3std* values [4]. It is calculated as follows:

$$\text{mean3std}(\Delta IC_tC_p) = mean(\Delta IC_tC_p) + 3\sigma(\Delta IC_tC_p) \tag{5}$$

where $\sigma(\cdot)$ is the standard deviation. In this way, if the *mean3std* is a certain value, this ensures that most of the pixels of the under-test image have a color distortion below it.

## Optimization Method

The proposed optimization method evaluates the distribution of the error inside the 3D-LUT and shifts the vertices of the cube to obtain the overall minimal error inside each sub-cube. In this section, the method is explained only for one channel (the optimization of the other two works in the same way).

The process starts by uniformly distributing the vertices of the cube, then for each vertex position $x$ between the second and the second last (the first and the last vertices are always 0 and 1), the following operations are performed:

1. the area between $x-1$ and $x+1$ is divided into $n_s$ sub-cubes, where $n_s$ is the number of the sub-cubes tested during the optimization. The higher the value the more precise is the optimization;
2. a test image (like the one shown in Fig. 3) is created to evaluate the interpolation errors in the desired area. This image has been developed to be sure that every sub-cube is tested evenly;
3. the test image is interpolated by using the reference 3D-LUT;
4. the test image is distorted by applying the three 1D-LUTs, one for each channel. Note, in the beginning of the optimization, when the vertices are uniformly distributed these 1D-LUTs contain the identity transformation;
5. the interpolation error is calculated for each sub-cube;
6. the sub-cube which contains the minimum error is selected as suitable position for the vertex in the next position;

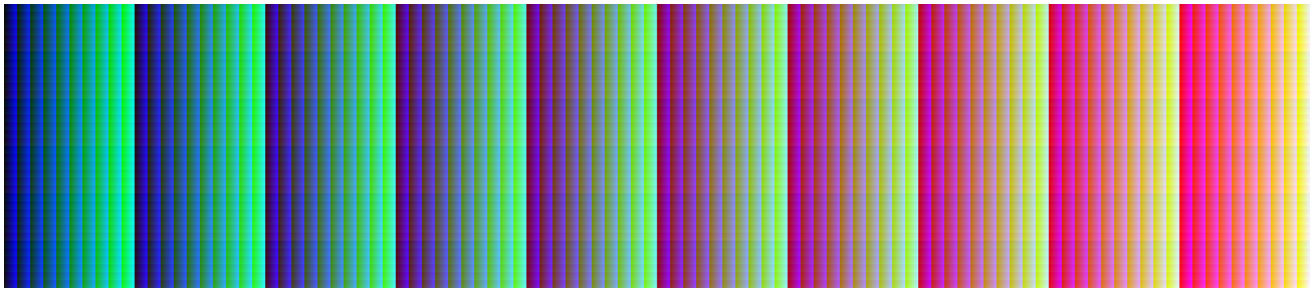


***Figure 3.*** *Starting test image used during the optimization process.*

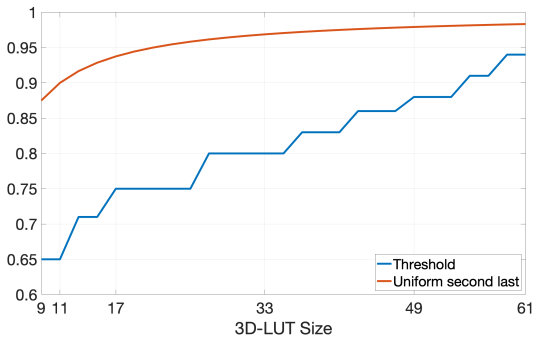



***Figure 4.*** *Optimization thresholds according to the 3D-LUT size.*

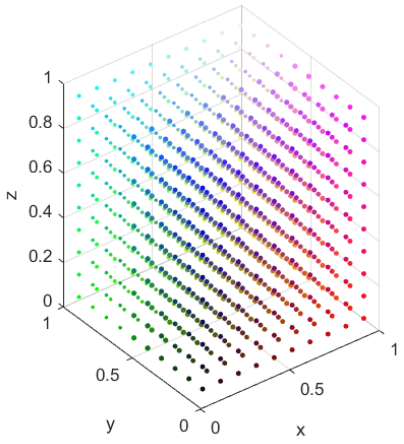



***Figure 5.*** *Positions and colors of the vertices of the* $11 \times 11 \times 11$ *uniform 3D-LUT. The x, y and z-axis are the red, the green and the blue channels, respectively.*

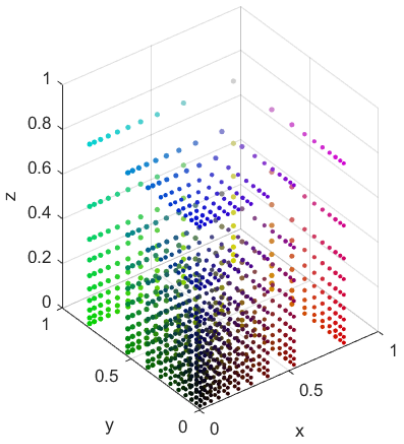


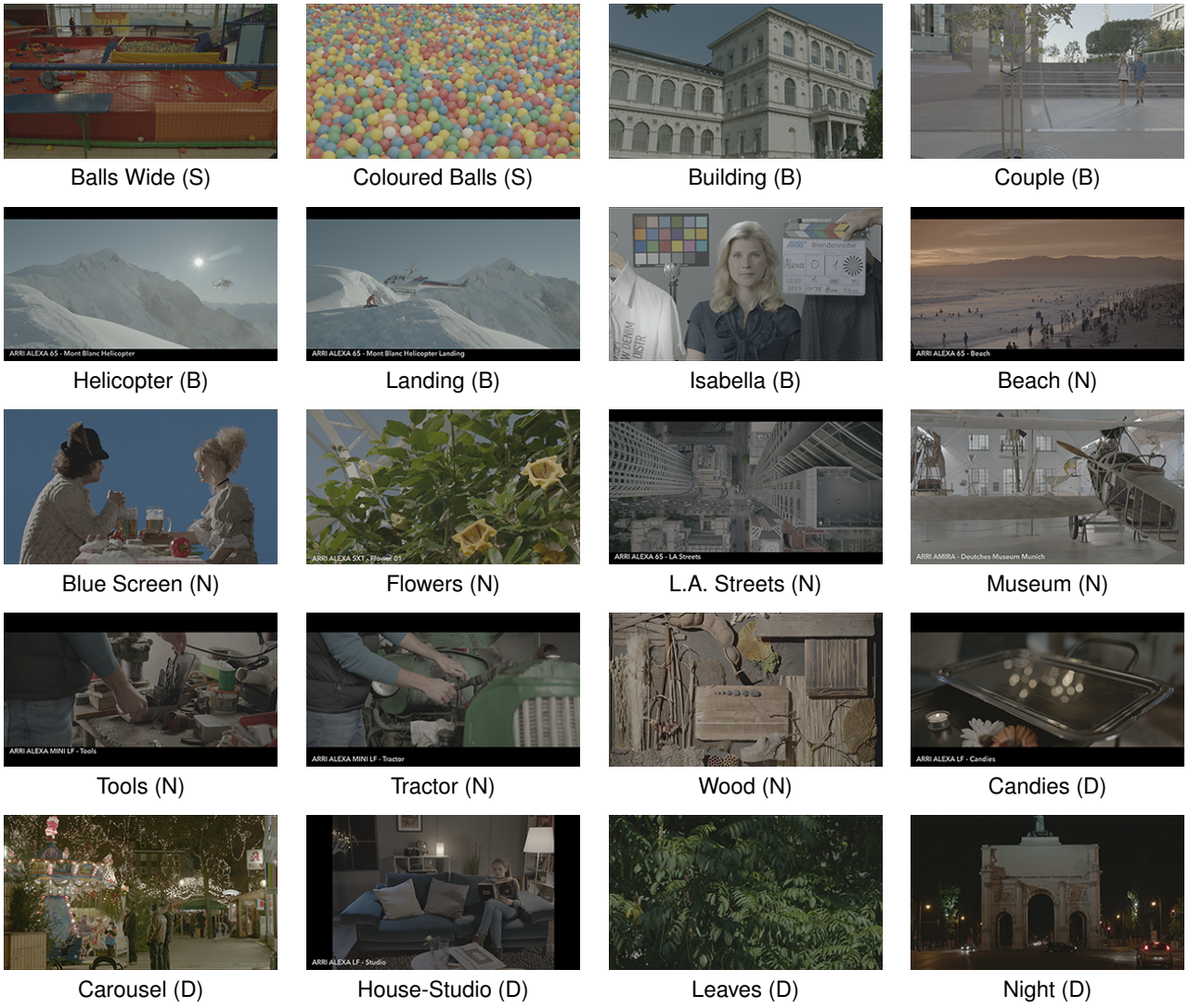

***Figure 6.*** *Positions and colors of the vertices of the* $11 \times 11 \times 11$ *optimized 3D-LUT. The x, y and z-axis are the red, the green and the blue channels, respectively.*

The process continues and the position $x+1$ of the next vertex is optimized. When the process reaches the last vertex, it starts again from the beginning. Initially, the stop condition was the convergence of the error between two iterations. However, even if it worked, the computational time was too high, and the optimized 3D-LUTs were too dense in the dark areas creating visual artefacts in the bright ones. The problem was annoying especially for HDR 3D-LUTs due to the instability of the $\Delta IC_tC_p$ metric in the dark and saturated areas.

The solution to this problem was to introduce a threshold to stop the iteration. However, this threshold does not work on the error, but on the position of the second-last vertex. As soon as this vertex is smaller than the threshold the iteration is stopped and the 3D-LUT is optimized. This exit strategy has the advantage of ensuring that the vertices are not too sparse in the bright areas. The threshold value depends only on the size of the optimized 3D-LUT and, at the moment, it has been empirically estimated. In Fig. 4, the stopping threshold is compared with the second-last vertex position in a uniform distributed 3D-LUTs. It is clear that, after the optimization, the second-last vertex is much darker than before, but, at the same time, the threshold ensures that it is not too dark.

Figure 5 shows the color of the vertices of a $11 \times 11 \times 11$ uniform 3D-LUT, while Figure 6 shows the 3D-LUT after the optimization. As explained above, the vertices are now more dense in the dark area, leaving the bright colors with a relative less precise representation. Figure 7 shows the one dimensional distortion LUT applied to the three color channels of a RGB image before the use of the optimized $11 \times 11 \times 11$ 3D-LUT to perform the color interpolation in a uniform lattice into a not-uniform domain. From its initial steepness it is easy to understand how it compresses the values in the dark areas. It is a piece-wise linear

**Data-set used to test the optimized 3D-LUTs. Where (S) is for saturated images, (B) is for the bright images, (N) is for natural images and (D) is for dark images.**



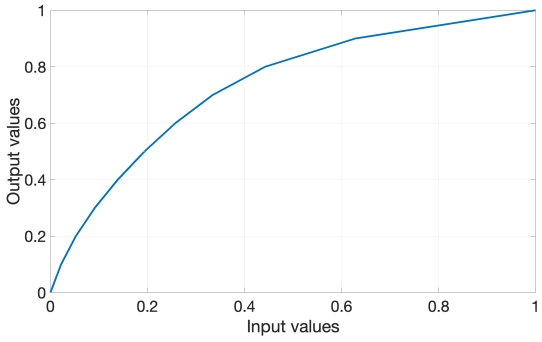




***Figure 7.*** *Example of distortion 1D-LUT applied to the green channel of the image before the optimized 3D-LUT.*

curve, and therefore very easy to implement in hardware without too expensive additional costs. The final memory requirement for an optimized 3D-LUT is then $3 \times N_o + N_o^3$, while the one for the uniform 3D-LUT is $N_u^3$. Notice that the color processing of an image employs several 1D-LUT transformations and the distortion 1D-LUT could be combined with one of them. In this case, the complexity of the optimized 3D-LUT is $N_o^3$. In the following of the paper, this simplification is not considered as it cannot always be used.

## Results

The proposed optimization method has been tested on different gamut conversion 3D-LUTs from the ARRI Wide Gamut LogC domain to the BT.2100 with different peak luminance and dark thresholds. The reference 3D-LUTs have dimension of either $65 \times 65 \times 65$ or $127 \times 127 \times 127$, and they have been reduced to the under-test size by a resize on a regular lattice for the uni-

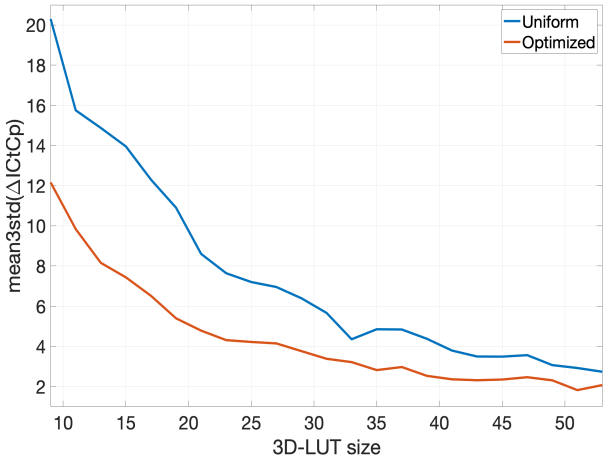



***Figure 8.*** *Optimized vs uniform 3D-LUTs $\Delta IC_tC_p$ results for natural images.*

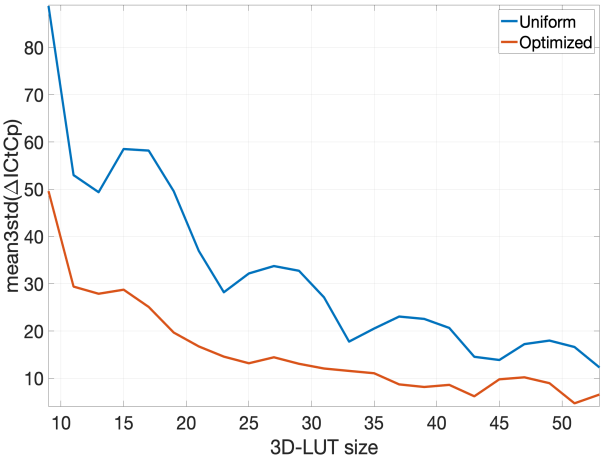



***Figure 10.*** *Optimized vs uniform 3D-LUTs $\Delta IC_tC_p$ results for dark images.*

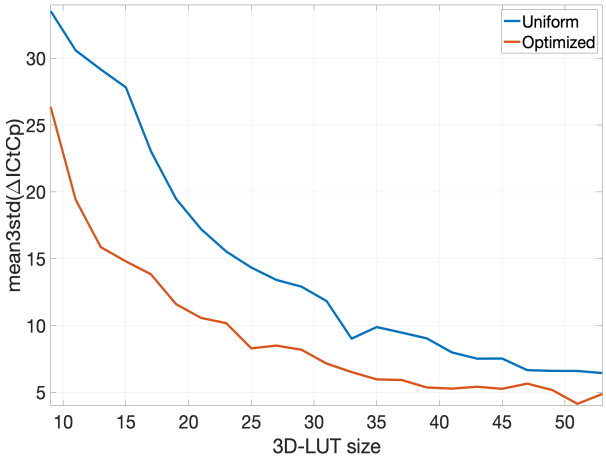



***Figure 9.*** *Optimized vs uniform 3D-LUTs $\Delta IC_tC_p$ results for saturated images.*

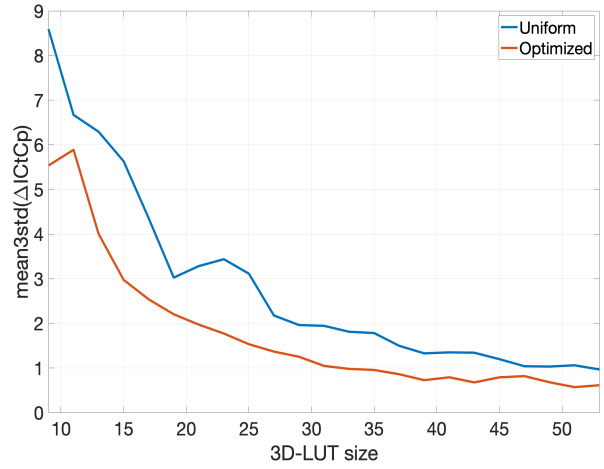



***Figure 11.*** *Optimized vs uniform 3D-LUTs $\Delta IC_tC_p$ results for bright images.*

form 3D-LUT, and by applying the proposed method to obtain the optimized one. Then these smaller 3D-LUTs have been applied to different kinds of images to verify the effectiveness of the proposed optimization. Table 1 presents the set of images used for the tests, they have been captured with the ARRI ALEXA family cameras in RAW format and then processed till the ALEXA Wide Gamut-LogC domain. Notice that, they look de-saturated because they are displayed without any gamut transformation. Those images can be divided into four big categories:

- **Natural**: images featuring the correct illumination and natural colors;
- **Saturated**: images with saturated colors, selected to test the optimized 3D-LUT where the $\Delta IC_tC_p$ metric has problems;
- **Dark**: night or under-exposed scenes where the optimized 3D-LUTs should be very effective considering that the optimization process usually shifts the vertex positions towards the dark;
- **Bright**: sunny or over-exposed scenes where the optimized 3D-LUTs should be less effective due to their sparser representation nearby the saturation.

The $\Delta IC_tC_p$ and then the *mean3std* values have been calculated between the reference, the uniform and the optimized images, i.e., the images interpolated by using the reference, the evenly resized and the optimized 3D-LUTs.

The results for the natural and saturated images are shown in Fig. 8 and Fig. 9, respectively. In these kinds of images, the optimization works well and an optimized 3D-LUTs having dimension $33 \times 33 \times 33$ is able, in average, to give the same color distortion of a $43 \times 43 \times 43$ uniform 3D-LUT.

Fig. 10 shows the results of the optimized 3D-LUT applied to the dark images. Also in this situation the optimization is able to strongly reduce the color distortion for a given 3D-LUT size, but this is not the only advantage of the optimization. On this kind of images, the uniform 3D-LUTs (blue curve) present a bumpy

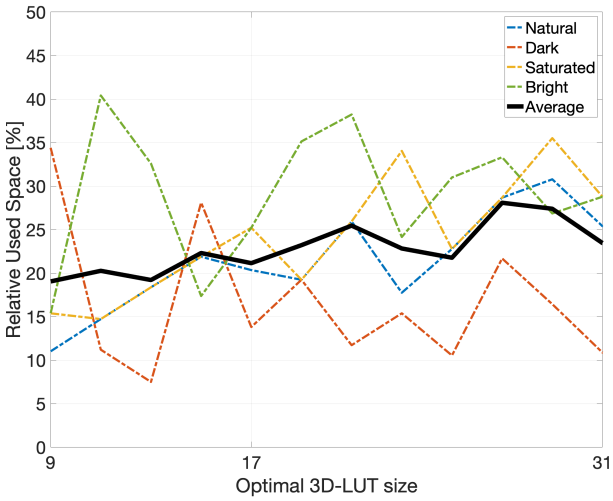



***Figure 12.*** *Relative used space.*

trend, where a smaller 3D-LUT presents sometimes less color distortion than bigger ones. For the optimized 3D-LUTs (red curve) this behaviour is much less noticeable and larger sizes almost always have better results.

Fig. 11 shows the results for bright images. For this kind of images the optimization gain is smaller, but this was predictable because the optimization process tends to reduce the color distortion in the dark regions at the expense of the bright ones.

Another way to look at the results is to calculate the memory space saved by using the optimization for a given color distortion. In fact, in secondary images paths, such as monitor pre-viewing or on-set devices, the image quality could be sacrificed for the benefit of a simpler hardware implementation, so the optimization could be seen as a way to save memory. This gain is shows as a percentage of the memory space required for storage of the uniform 3D-LUT that provides the same distortion error:

$$\text{relative used space} = \left(\frac{3N_o + N_o^3}{N_u^3}\right) * 100 \tag{6}$$

where $N_u$ and $N_o$ are the sizes of the uniform and optimized 3D-LUTs, respectively, that ensure the same color difference.

Fig. 12 shows that the optimized 3D-LUTs need, on average, four times less space than the uniform ones. This is a huge advantage in hardware, where all the resources are heavily constrained in term of FPGA space or power consumption.

## Conclusions

With the recent introduction of the HDR and WCG requirements, memory constrain and acceptable color reproduction constraints are becoming an issue that could not be solved by simply using a better interpolation, e.g., tetrahedral instead of trilinear, as shown in [4]. This paper proves that a non-uniform 3D-LUTs could be a solution to this problem. The proposed method showed to be effective in all kind of images and the optimized 3D-LUT are able to achieve the same color distortion of the uniform ones with four times less memory space (on average). These results were obtained performing the optimization on a computer generated image where all the colors have the same weight. Better results could be obtained by considering the type of images on which the 3D-LUT will be applied during the optimization process.

## References

[1] ITU-R:BT.709-2015, Parameter values for the HDTV standards for production and international programme exchange.

[2] SMPTE:RP 431-2:2011 - SMPTE Recommended Practice - D-Cinema Quality — Reference Projector and Environment, 1-14, 2, 2011.

[3] ITU-R:BT.2100-2.2018, Image parameter values for high dynamic range television for use in production and international programme exchange.

[4] JD. Vandenberg and S. Andriani, A Review of 3D-LUT Performance in 10-Bit and 12-Bit HDR BT.2100 PQ, SMPTE Motion Imaging Journal, 2, 129 (2020).

[5] H R Kang, Color technology for electronic imaging devices, Washington: SPIE Optical Engineering Press, 1997: 55-63

[6] V. Monga and R. Bala, Algorithms for color Look-Up-Table (LUT) design via joint optimization of node location and output values, Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 998-1001, 2010

[7] Dolby, ICtCp Dolby White Paper, 2016, Dolby Laboratories, Inc.

[8] , E. Pieri and J. Pytlarz, Hitting the Mark - A new color difference metric for HDR and WCG Imagery, SMPTE Motion Imaging Journal, 4, 127 (2018).

[9] SMPTE, High Dynamic Range Electro-Optical Transfer Function of Mastering Reference Displays, SMPTE ST 2084:2014, August 2014, 1-14.

[10] T. Borer and A. Cotton, A display independent High Dynamic Range Television system, BBC WHP 309, September 2015.

## Author Biography

*Stefano Andriani received his MS in telecommunication engineering from the University of Padua (2003) and his PhD in telecommunication and electronic engineering from the same university (2007). Since then he has worked in the R&D Department at Arnold & Richter Cine Technik (ARRI) in Munich, Germany. His work has focused on color reconstruction, sensor error correction and compression.*

*Aurora Zabot received her MS in ICT for Internet and Multimedia from the University of Padua in 2020. Her study path has been focused mainly on topics related to Computer Vision, Machine/Deep Learning and Multimedia Processing. Currently, she is attending an intensive course in Cloud Engineering.*

*Giancarlo Calvagno received the MS in Electronics Engineering (1986) and the PhD in Electronics and Information Engineering (1990) from the University of Padova. From 1988 to 1990 he was at the University of Illinois at Urbana-Champaign as Visiting Scholar. Since 1990 he has been with the Department of Information Engineering of the University of Padova, presently as Associate Professor. His research interests are in the area of digital signal processing, reconstruction and coding.*

*JD Vandenberg is an electrical engineer and mathematician working as an imaging and color science manager for the Walt Disney and Marvel Studios, Burbank, CA. In 2014, he joined Technicolor, Los Angeles, CA, as the product manager of the on-set color correction system called DP Lights. In 2018, he joined the Walt Disney Studios where he supports cross studio productions and color workflow.*

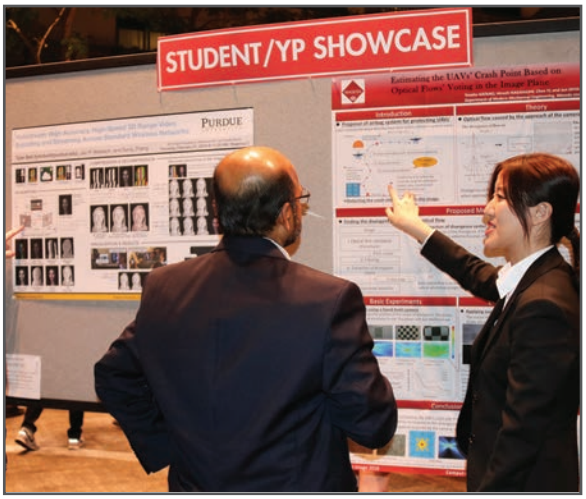
# JOIN US AT THE NEXT EI!

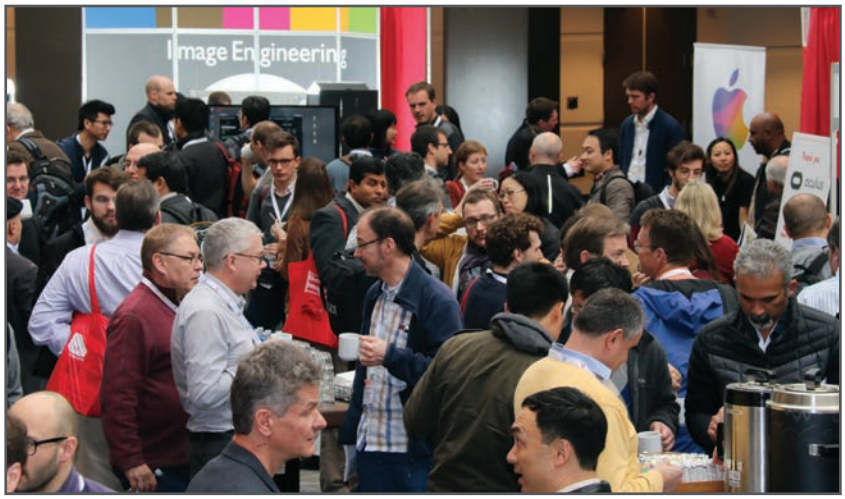
IS&T International Symposium on

# Electronic Imaging

SCIENCE AND TECHNOLOGY

*Imaging across applications . . . Where industry and academia meet!*





**• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •**
**• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •**

**www.electronicimaging.org**

IS&T
imaging.org