LOG-ENCODING ESTIMATION FOR COLOR STABILIZATION OF CINEMATIC FOOTAGE

Javier Vazquez-Corral and Marcelo Bertalmío

Dept. de Tecnologies de la Informació i les Comunicacions, Universitat Pompeu Fabra, Spain

ABSTRACT

We propose a method for the color stabilization of cinema shots coming from different cameras that use unknown logarithmic encoding curves. The log-encoding curves are approximated by a concatenation of gamma-curves, whose values are accurately computed using image matches. The color stabilization procedure, based on the generic color processing pipeline of a digital camera, can be performed after the estimation of the encoding curves, and it also requires the existence of image matches. Our work can be applied in different scenarios such as multi-camera shoots, native-3D cinema, or color grading in post-production.

***Index Terms*—** Color stabilisation, color image analysis, color matching, non-linearity estimation

1. INTRODUCTION

Two cameras capturing the same scene, at exactly the same moment, will produce two pictures with colors that do not exactly match. This is also true when using the same camera with different user-defined settings or in automatic mode. This difference can cause problems for a wide range of applications where a multi-camera set-up is common (like professional movie shooting), or mandatory (e.g. some TV broadcasts and 3D cinema). Recently this problem, known as color stabilization, has gained special attention for amateur, gamma-corrected (i.e. the encoding curve follows a power law) images [1, 2, 3, 4, 5]. In general, color stabilization methods solve the problem by considering one of the images as the reference image and correcting the colors of the other images (known as target images) to match the colors presented in the reference one.

In the cinema industry it is common to encode images using a logarithmic-based curve (known as log-encoding), instead of performing gamma-correction. Logarithmic encoding reduces quantization errors and avoids the loss of detail in the dark regions of the images. This is important at post-production stages as colorists may want to enhance those details. Different manufacturers slightly differ on the formulation of the log-encoding curve, but a standard formula used by two of the major cinema camera manufacturers [6, 7] is the following:

$$I_{output} = c \cdot \log_{10}(a \cdot I_{linear} + b) + d, \tag{1}$$

where a, b, c, d are parameters that depend on the exposure.

In this paper our goal is to color stabilize a pair of images when at least one of them has been log-encoded and neither of the encoding curves used for the input pair are known. To our knowledge, this is the first work that addresses color stabilization for cinema footage, where log-encoding is predominant. From Eq. 1 we can see that working with such images encompasses a higher degree of difficulty than the case of gamma-corrected images, as there are four parameters to be estimated. We base our work in the following observation: log-encoding curves can be locally approximated by gamma curves, whose gamma-values can be accurately estimated when a set of corresponding achromatic matches is present in the images.

We test our method in two different scenarios. The first scenario is related to simultaneous multi-camera situations, where the images have some shared content. The second scenario is related to post-production modifications, where the color-grading process requires to perform color stabilization among shots of different scenes. We will see that our technique is based on matching image values among the pair, therefore taking the input images with a calibration checker card present in the images improves the quality of the results (and is required when the images do not share content because they come from different scenes).

2. RELATED WORK

As mentioned above, we believe this is the first work to deal with the problem of color stabilization for log-encoded footage. There is, however, a vast literature on addressing the same problem for gamma-corrected images both in terms of video and still images.

The more general approach to the problem is that of global color transfer methods [8, 9, 10]. These methods do not require any shared content among the scenes. They apply a single color transformation learned from the statistics of the image pair.

One of the main works on color stabilization is the one by Hacohen *et al.* [2, 3]. It is based on obtaining a dense set of correspondences between the pair of images, then the fitting

This work was supported by the European Research Council, Starting Grant ref. 306337, by the Spanish government, grant ref. TIN2015-71537-P, and by the Icrea Academia Award.

of a color model to these correspondences, and the application of this color model to the whole image. The drawbacks of this method stem from the difficulty of finding reliable correspondences in very large smooth regions and in scenes with strong lighting changes.

Kim *et al.* [11] characterize a set of gamma-corrected cameras and get back to the RAW information, which allowed them to obtain impressive results in color transfer applications. However, their method relies on the previous obtention of RAW-JPEG pairs, and it is designed specifically for gamma-corrected images. Chakrabarty *et al.* [12] improved this model by introducing an uncertainty criteria, and therefore not considering all the pixels as equally important.

For the video stabilization problem, Farbman and Lichinski [4] presented a method where some frames are designated as anchors and a set of correspondences to them are found from the remaining frames. From these correspondences, a very simple color model (not considering cross-channel talk) is learned. The main drawback of this method is the need of temporal coherence among frames. Recently, Frigo *et al.* [5] have presented a way of reducing this drawback by considering the motion speed as a cue for guiding the tonal stabilization process. Wang *et al.* [13] also handled the video stabilization problem by defining 'color states' that represent the exposure and white balance of a frame.

Finally, Vazquez-Corral and Bertalmío in [1] obtain a single color transform by following the color processing pipeline of digital cameras. Our method is based upon this one, so we give a detailed explanation of [1] in Section 2.1.

2.1. Color processing pipeline in digital cameras

In [14] it is proposed that the color processing pipeline of digital cameras can be summarized as

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix}_{out} = \left(A \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}_{in} \right)^{\frac{1}{\gamma}} \tag{2}$$

where A is a 3×3 matrix comprising white balance and color encoding, RGB_{in} is the camera raw triplet at a given pixel location, and a pixel-based non-linear function defined as a power law of exponent $\frac{1}{\gamma}$ is applied to each pixel value.

Let us suppose we have some shared content RGB_{in} viewed under two different cameras, so we have

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix}_{1} = \left(A_1 \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}_{in} \right)^{\frac{1}{\gamma_1}} ; \begin{bmatrix} R \\ G \\ B \end{bmatrix}_{2} = \left(A_2 \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}_{in} \right)^{\frac{1}{\gamma_2}} \tag{3}$$

We know that in this case the values of RGB_{in} should be equal in both cameras and, therefore, we obtain that for these corresponding pixels

$$\left(\begin{bmatrix} R \\ G \\ B \end{bmatrix}_{1} \right)^{\gamma_1} - H \cdot \left(\begin{bmatrix} R \\ G \\ B \end{bmatrix}_{2} \right)^{\gamma_2} = 0 \tag{4}$$

where $H = A_1 A_2^{-1}$. Let us note that $([RGB]_i^T)^{\gamma_i}$ represents the linearized value of image i, and therefore, Eq.(4) shows that matrix H performs the color stabilization between the linearized version of both images.

Consenquently, in [1] authors look for γ_1,γ_2 and H that minimize Eq.(4) by minimizing the error in a least-squares sense for the set of corresponding pixels. Then, the whole second image can be color corrected to match the first one by applying

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix}_{2}^{\prime} = \left(H \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}_{2}^{\gamma_2} \right)^{\frac{1}{\gamma_1}} \tag{5}$$

3. COLOR STABILIZATION

In this section we explain the main contributions of our work. In Section 3.1 we show how a log-encoding curve can be approximated by a concatenation of gammas depending on the grey-level value of the pixel. This observation allows us to use the stabilization method of Section 2.1 in achromatic pixels to obtain the non-linearities of the image pair in Section 3.2.

3.1. Approximating a log-encoding curve by a set of gamma curves

Let us consider a vector x of ordered values from 0 to 1. Let us now define I_1 and I_2, two non-linear versions of it

$$I_1(x) = (x)^{\frac{1}{\gamma}} \tag{6}$$

$$I_2(x) = c \log_{10}(a \cdot (x) + b) + d \tag{7}$$

Our goal is to find γ values (under known a,b,c, and d) so that in a particular interval $I_1(x) \approx I_2(x)$. To prove the plausibility of the approximation, we run the following experiment. We divide the range $[0, 1]$ into 40 different intervals, and compute at each interval the value of γ making I_1 best approximate I_2 (via a least-squares minimization). This has been done for four different log-encoding curves (Arri320, Arri640, Arri1280, ans Sony S-Log). We have then computed the error difference between the real curve and its approximation in terms of the mean of the percentage error at each point. Results are presented in Table 1, where we can see than in all cases the error is below 0.5%, confirming that it is possible to approximate log-encoding curves via gamma curves.

3.2. Characterization through achromatic matches

In the previous section we demonstrated that a log-encoding curve can be estimated by a concatenation of gamma curves. However, this is not the end of our problem, since given a non-linear color pixel value (R, G, B) the estimation of the linear values will lead to $(R^{\gamma_r}, G^{\gamma_g}, B^{\gamma_b})$, where $\gamma_r, \gamma_g, \gamma_b$ depend on the grey-level value of the pixel in each channel.

Algorithm 1 Stabilization from a set of achromatic matches

Given a pair of log-encoded images I_1,I_2
Obtain a set of achromatic matches $\{P, Q\}$ such that $I_1^p = I_1(P) \approx I_2(Q) = I_2^q$
Randomly initialize $NL_{I_1}(x) = c_1 \log_{10}(a_1 \cdot (x) + b_1) + d_1$
while NL_{I_1}, NL_{I_2} do not converge **do**
  Compute $I_{1,linear}^p = NL_{I_1}^{-1}(I_1^p) = pow(10, \frac{I_1^p - d_1}{c_1} - b_1)/a_1$
  for each interval k of achromatic matches **do**
    Obtain γ_2^k (the gamma value for I_2 at interval k) by applying Eq.(5) to the matches of $I_{1,linear}^p$ that belong to the interval and their corresponding I_2^q
  end for
  Estimate $NL_{I_2}(x) = c_2 \log_{10}(a_2 \cdot (x) + b_2) + d_2$ from the values γ_2^k
  Interchange I_1 and I_2
end while
Obtain $I_{1,linear}$ and $I_{2,linear}$ from NL_{I_1} and NL_{I_2}
Obtain the matrix H between $I_{1,linear}$ and $I_{2,linear}$ as in Eq.(4)
Stabilize the images using Eq.(8)

Therefore, in order to use the approximation suggested in Section 2.1 we should work on the pixels where the three color channels have similar values, i.e. the achromatic pixels. Let us also note that at different gray-level intensities the gamma values would also be different. Accordingly, we apply an iterative process to obtain the non-linearities of the image pair by spliting the different achromatic pixels into intervals depending on their color values, as we explain just below.

Let us suppose we have two images I_1 and I_2, encoded by a logarithmic function. First, we select the corresponding matches $\{P, Q\}$ such that $I_1(P) \approx I_2(Q)$ with the condition that the matches are achromatic. Let us call I_1^p and I_2^q the $n \times 3$ matrices containing all these matches, where n is equal to the number of matches and columns represent the R, G, and B value for each pixel. We start our algorithm by initializing the non-linearity of I_1: $NL_{I_1}(x) = c_1 \log_{10}(a_1 \cdot (x) + b_1) + d_1$ to some random values a_1, b_1, c_1, d_1. We apply the inverse of this non-linearity to I_1^p obtaining the linear version of the matches: $I_{1,linear}^p = NL_{I_1}^{-1}(I_1^p) = pow(10, \frac{I_1^p - d_1}{c_1} - b_1)/a_1$. Then, we split the linear matches of $I_{1,linear}^p$ into different intervals based on their grey-level values. For each interval k we apply Eq.(5) to the matches of $I_{1,linear}^p$ falling in the interval and their corresponding I_2^q matches to obtain γ_2^k (i.e. the gamma value for I_2 in the interval k). Later on, we fit a logarithmic curve to the set of γ_2^k values obtaining the non-linearity of I_2: $NL_{I_2}(x) = c_2 \log_{10}(a_2 \cdot (x) + b_2) + d_2$. Then, we start the process again by obtaining $I_{2,linear}^q$ from NL_{I_2} and looking for NL_{I_1}. This process is repeated until both non-linearities converge, i.e. the differences between the current and previous NL_{I_1} and the current and previous NL_{I_2} are below some threshold. Then, we undo both non-linearities and find the matrix H converting one linear image to the other as in Eq.(5). Matrix H is computed using the full set of correspondences (both chromatic and achromatic). Finally, the whole I_2 is matched to I_1 by applying

$$I_2' = NL_{I_1}(H \cdot NL_{I_2}^{-1}(I_2)) \tag{8}$$

A more detailed explanation of the algorithm can be found in Algorithm 1. Let us note that in the case where one of the image pair is log-encoded and the other is gamma-corrected, the same algorithm applies, we just need to use a power-law instead of a log-curve as the model for the NL function.

	Arri320	Arri640	Arri1280	S-log
L2-approx	0.46%	0.41%	0.35%	0.35%

Table 1. Average error percentage at each point between the different curves and our approximations via local gammas

Method	Median	Mean	RMS
Ours	**0.0097**	**0.0167**	**0.0240**
Random	0.0157	0.0241	0.0359

Table 2. Mean, median and RMS of the distance between the real and the estimated curve for the set of 48 images.

4. RESULTS

Let us start by showing that given a set of achromatic matches between a pair of images it is possible to find the non-linearity present in each image. To this end, we designed the following experiment: We considered a set of 48 different RAW images obtained by a Nikon5100 camera. For each RAW we obtained 2 different images by first multiplying the RAW image by a 3×3 matrix A that varies in every case and then applying two different log-encoding curves with typical parameters. We found the achromatic matches for each image pair and applied our method to estimate the non-linear encoding curves for both images. The error between our estimation and the real curves is computed as the difference in area between them, and summarized (for the 48 images) in Table 2. We compare our method versus a random paradigm that chooses one solution among the set of 11 possible curves used in the experiment (the curves given in [6]). We can see that we greatly outperform the results of the random paradigm by more than a 33% percent.

4.1. Color-stabilization results

We have collected a dataset of images in the following manner. We have captured different scenes twice, one with both a grey and a Macbeth checker and one without them. Some

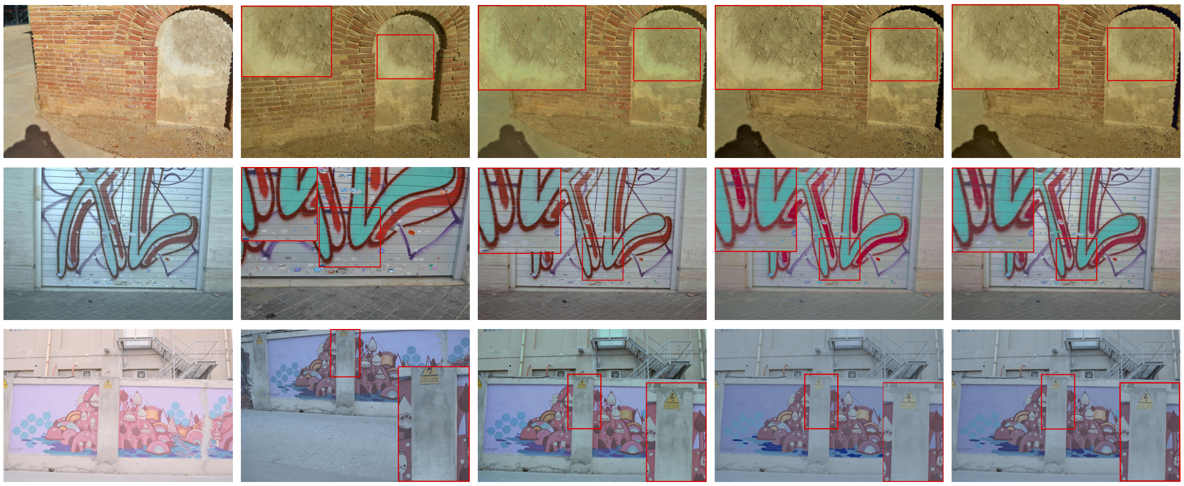

Fig. 1. Results of our method for the case where there is shared content. From left to right: Source image, reference image, result of Hacohen *et al.* [2], our result without considering the grey-checker, our result considering the grey checker.

of the scenes are directly the JPEGs of a NikonD3100 camera, while others have been captured in RAW and then log-encoded by first multiplying the image by a 3×3 matrix A that varies in every case and then applying a log-encoding curve that also varies in every case. Our idea is to take either the JPEG or log-encoded version of one image and color-match it to some other image with a different non-linearity. To this end, we will use the method explained in section 3.2, that is, we will follow the procedure outlined in Algorithm 1.

Our first experiment studies the case where there is shared content between the two images. In this case, we have run our method in two ways: computing the achromatic matches from the images without checkers, or instead from the gray-checkers only. Results for this experiment are shown in Fig.(1), where we compare our method to the one of Hacohen *et al.* [2]. In the first row of the figure the source image is a JPEG and the reference is a log-encoded image, in the second row the source is a log-encoded image and the reference is a JPEG, and in the third row both are log-encoded images. We want the reader to focus in the cropped regions, to perceive that a greenish cast is introduced by Hacohen *et al.* in the top image, that the green character in the graffiti is better solved by our method in the middle image, and that the grayish column in the bottom image is also better corrected by our method. The addition of the gray-checker for obtaining the non-linearities slightly improves our results. Note that for display reasons we present log-encoded images in sRGB.

Our second experiment overcomes the restriction of working with images presenting some shared content. To this end, we will consider also the information coming from the Macbeth color checker, whose matches will be used to obtain the color stabilization matrix H. Results of this second experiment are presented in Fig.(2). The left column of the Figure represents the source image (a gamma-corrected image), the second column represents the reference image, in this case is a log-encoded one, and the third column shows our color stabilization result. Let us note that we tried to run the method of Hacohen *et al.* [2] in the color checkers of these images, but the method was not able to find any reliable matches due to the large color differences among them.

Fig. 2. Results for the case of not shared content. From left to right: Source image, reference image, our result.

5. CONCLUSIONS

We have presented a method to color stabilize a pair of images with shared content specially focusing on log-encoded footage. Our method builds upon the fact that log-encoding curves can be estimated by a concatenation of gamma curves, leading us to present an iterative method based on the set of achromatic matches among the pair. Our work has many applications in the TV and cinema industries (e.g. multi-camera broadcasts or 3D cinema). Future work will deal with the computational speed of our approach, the manual selection of image matches, and the extension of the method for images presenting no achromatic matches.

6. REFERENCES

[1] J. Vazquez-Corral and M. Bertalmío, "Color Stabilization Along Time and Across Shots of the Same Scene, for One or Several Cameras of Unknown Specifications," *Image Processing, IEEE Transactions on*, vol. 23, no. 10, pp. 4564–4575, Oct 2014.

[2] Y. HaCohen, E. Shechtman, D.B. Goldman, and D. Lischinski, "Non-rigid dense correspondence with applications for image enhancement," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4, pp. 70, 2011.

[3] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski, "Optimizing color consistency in photo collections," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, pp. 38, 2013.

[4] Z. Farbman and D. Lischinski, "Tonal stabilization of video," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4, pp. 89, 2011.

[5] J. Delon O. Frigo, N. Sabater and P.Hellier, "Motion driven tonal stabilization," in *ICIP*. IEEE, 2015.

[6] H. Brendel, "Alexa loc-c curve-usage in vfx," `https://www.arri.com/camera/alexa/downloads/`, 2012, [Online; accessed January-2016].

[7] H. Gaggioni, P. Dhanendra, J. Yamashita, N. Kawada, K. Endo, and C. Clark, "S-log: A new lut for digital production mastering and interchange application," `https://pro.sony.com/bbsccms/assets/files/mkt/cinema/solutions/slog_manual.pdf`, [Online; accessed January-2016].

[8] F. Pitié, A.C. Kokaram, and R. Dahyot, "Automated colour grading using colour distribution transfer," *Computer Vision and Image Understanding*, vol. 107, no. 1, pp. 123–137, 2007.

[9] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *Computer Graphics and Applications, IEEE*, vol. 21, no. 5, pp. 34–41, 2001.

[10] H. Kotera, "A scene-referred color transfer for pleasant imaging on display," in *Proceedings of IEEE ICIP 2005*, 2005, pp. 5–8.

[11] S. Kim, H. Lin, Z. Lu, S. Susstrunk, S. Lin, and M. Brown, "A new in-camera imaging model for color computer vision and its application," *IEEE Trans PAMI*, 2012.

[12] A. Chakrabarti, Ying Xiong, Baochen Sun, T. Darrell, D. Scharstein, T. Zickler, and K. Saenko, "Modeling radiometric uncertainty for vision with tone-mapped color images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 11, pp. 2185–2198, Nov 2014.

[13] Y. Wang, D. Tao, X. Li, M. Song, J. Bu, and P. Tan, "Video tonal stabilization via color states smoothing," *Image Processing, IEEE Transactions on*, vol. 23, no. 11, pp. 4838–4849, Nov 2014.

[14] S. Bianco, A. Bruna, F. Naccari, and R. Schettini, "Color space transformations for digital photography exploiting information about the illuminant estimation process," *JOSA A*, vol. 29, no. 3, pp. 374–384, 2012.