

Correction of Color Saturation for Tone Mapping Operator

E.D. Birukov¹, M.S. Kopylov¹, A.A. Khlopina¹

birukov@gin.keldysh.ru|kopylov@gin.keldysh.ru|nastyak@gin.keldysh.ru

¹Keldysh Institute of Applied Mathematics of RAS, Moscow, Russia

This article reviews the problems of color saturation correction during dynamic range compression of images. Special attention is drawn to emulation of human vision and photo cameras effects during processing of the images which have very bright areas (with brightness more than an order of magnitude greater than average brightness of the rest of the image). A method of desaturation of the brightest parts of images is suggested. This method provides lowering of the color saturation of each pixel depending on the pixel brightness. The improved method lowers saturation only of the brightest parts of an image. The size of the desaturated part of the image should not exceed 25% of the size of an entire image.

Keywords: color saturation, tone mapping operator, high dynamic range images, desaturation.

1. Introduction

Dynamic range of an image is ratio between the pixel brightness with maximal and minimal value. More broadly, dynamic range is ratio between the maximal and minimal brightness which can be obtained from a photo or video camera, simulated with a computer graphics system or displayed at some device. The problem of range compression of images is very relevant in present time. Mostly this task is required during conversion of physical values such as luminance and illuminance to pixel brightness at the image display devices such as computer monitor or projector. Range of luminance and illuminance values which were obtained during a light propagation simulations in physically accurate computer graphics systems may be several orders of magnitude higher than dynamic range of mass-market computer monitors. Even many photo cameras already have high dynamic range [1].

A problem of color saturation often appears during solving this task. For example, if only brightness is compressed with keeping original color values, then image will be significantly oversaturated. Otherwise, if each color channel (in RGB model) is being compressed separately then most often it will seem that saturation of the image is very low, so that the entire image becomes nearer to grayscale. During setting color balance and saturation of images it is also reasonable to take specifics of human vision into account [6]. For example, for the darker parts of an image color saturation should be decreased to some extent because human color perception works worse in the darkness. But for the brightest highlights saturation also should be decreased because when eyes are adapted to low brightness then small bright part of the observed space will seem just white, without distinguishing of color hues. Typical example of such saturation lowering is a dark room with a small window leading out, while outside there is a sunny weather with bright sky.

2. Overview of the existing methods of color correction

In this article we describe compression of separate color channels for RGB color model because it is the most popular color model in computer graphics. It should be noticed that pixel brightness value which is used for calculations should be obtained either from some external source (for example, during the lighting simulation in the realistic computer graphics system), or using some conversion from RGB color model, for example, with SRGB conversion:

$$Y = 0.2126R + 0.7152G + 0.0722B$$

Basic formula of compression of separate color channels is as follows:

$$C_{out} = \frac{C_{in}}{L_{in}} L_{out} \quad (1)$$

where C_{in} - is one of the color channels of the original image (red, green or blue),

C_{out} - is one of the color channels of the transformed image,

L_{in} - is the pixel brightness of the original image,

L_{out} - is the pixel brightness of the transformed image obtained using some tone mapping (brightness compression) operator.

This formula doesn't correct color saturation, so output image often becomes oversaturated. There are different methods of correcting color saturation. Most of them allow tuning only the entire color/saturation balance over the entire dynamic range, without taking into account human vision specifics described above.

Tumblin and Turk [9] proposed the following formula of dynamic range compression for color correction:

$$C_{out} = \left(\frac{C_{in}}{L_{in}} \right)^s L_{out} \quad (2)$$

where s - parameter in this formula controls color saturation.

The main problem of this formula is that it may significantly modify image brightness in case if s is not equal to 1 and colors are different from grayscale. Taking this into account Mantiuk et al [4] proposed their own formula for compressing separate color channels:

$$C_{out} = \left(\left(\frac{C_{in}}{L_{in}} - 1 \right) s + 1 \right) L_{out}.$$

This formula keeps the brightness and tunes only the linear interpolation between chromatic and the corresponding achromatic colors. But its side effect is shift of color hues, especially for red and blue channels. So Mantiuk et al also suggested an alternative variant: usage of the single formula of dynamic range compression which is applied to all color channels. It should be equivalent to the formula (2) in case if color correction coefficient s is equal to dynamic range compression coefficient c and dynamic range compression function looks as follows:

$$L_{out} = (L_{in} b)^c$$

where b - is the brightness tuning coefficient which normalizes maximal value of the output brightness so that it will be equal to 1.

If the tone-curve is an arbitrary function, applying the same tone-curve to all color channels is not equivalent to (2) under $s = c$, but the results are very close. In case of local tone mapping operators, the three color channels usually cannot be modified

simultaneously. If c is less than s , it will lead to decreasing of saturation.

3. Usage of saturation lowering in case of high brightness

Methods described above are used for tuning color balance and saturation over the entire dynamic range of an image. Such tuning is indented for preserving color distinguishing of an original image after dynamic range compression. But under certain circumstances also additional lowering of color saturation is needed for the brightest parts of an image (in particular, for highlights). Many images which initially had a high dynamic range look unnaturally after compression because they contain oversaturated colors in the brightest zones which should seem just white. First of all it is caused by specifics of human vision. Besides, such “burnout” effect for brightest parts of an image occurs in the photographic equipment, including the modern digital devices. Usually it can appear if exposure time during taking photo is too great. But it should be noticed that in some cases it is necessary to set a long enough exposure time intentionally for providing sufficient brightness of the key parts of an image. And for all that some bright parts of the image inevitably will be overlighted. There are methods which decrease maximal brightness in such cases providing maximal distinction of all details, for example, [3, 7]. But contrary actions are often required for obtaining realistic looking simulated images. It causes simulated images to look as photo images of the real world taken with a usual photo camera.

In Keldysh Institute of Applied Mathematics we have developed the following original method which makes images

having very bright highlights to look more realistic after compressing dynamic range. With the help of this method one can simulate the effects described earlier for artificially rendered images. The method suggests lowering of color saturation with increasing of brightness.

The first implementation of the method had the following algorithm:

1. Find the average value of the color channels brightness in the given pixel. We'll call it $\overline{C_{in}}$;
2. Find the difference between the color channel value and the average value;
3. Divide the pixel brightness by the maximal brightness value of the image. It will be the individual coefficient of saturation lowering for this particular pixel;
4. Finally, difference between the brightness value of each color channel of the pixel and the average value should be multiplied by the coefficient described above. Then it should be subtracted from the color channel value itself. Due to this, values of the color channels will be nearer to each other and so color saturation of this pixel will be decreased.

So, formula (1) is being transformed to the following new formula:

$$C_{out} = \left(C_{in} - \left((C_{in} - \overline{C_{in}}) * \frac{L_{in}}{L_{in\ max}} \right) \right) * \frac{L_{out}}{L_{in}}$$

An example of the results of the tone mapping operator proposed in [1] and the same algorithm with saturation lowering proposed here is shown on fig. 1.

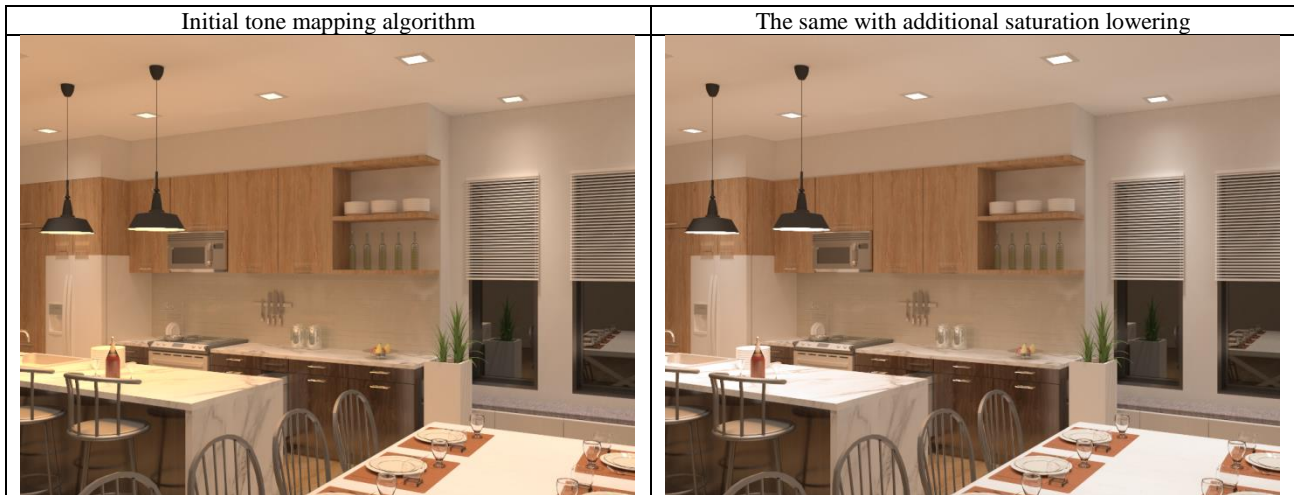


Fig. 1. Saturation lowering without additional parameters.

Attention should be drawn to the lamps and table surfaces. They are shown at figs. 2 and 3. One can see that at the original image they are colored with yellow while after color correction they become almost white. At the same time other parts of the image have near the same color saturation.



Fig. 2. Fragment of the image (table top). Above is the original tone mapping, below the desaturation is added.

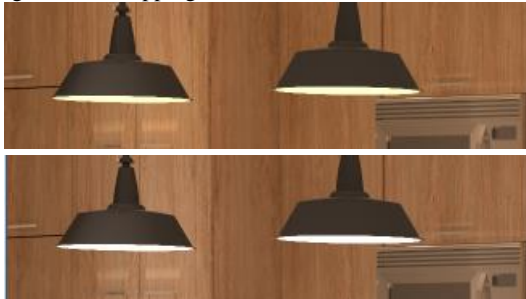


Fig. 3. Fragment of the image (lamp). Above is the original tone mapping, below the desaturation is added.

It should be noticed that desaturation itself in this algorithm is done in RGB color model by adding (subtracting) some value to each color channel in order to make the value of this channel nearer to the average one. In our opinion it is much easier than preliminary conversion of the image to a color model with the special saturation parameter. For example, IPT color model [2] is used in the method of Pouli et al [5]. Saturation parameter of the transformed image (which, in its turn is calculated for HCL color model) in this method is multiplied by relation of the original image saturation to the compressed one, and the saturation is being calculated in the IPT color model.

Unfortunately, using the same formula for desaturation in all cases makes no sense. Under these circumstances images which have many pixels with high brightness will have inadequately low color saturation over most of the image. Example of such image is shown at fig. 4. At this image the apple is unnecessary desaturated.

Taking this into account, there should be found some ways of tuning saturation lowering. The easiest way of such tuning is adding the desaturation coefficient which should be set manually by a user. This coefficient may have values from 0 to 1. Individual desaturation coefficient of each particular pixel (which was obtained at the step 3 of the described algorithm) should be multiplied by this additional coefficient. For images with very bright highlights and the low-brightness background the value of this coefficient should be increased. For images which have near

equal high enough brightness over most part of the image it should be decreased.

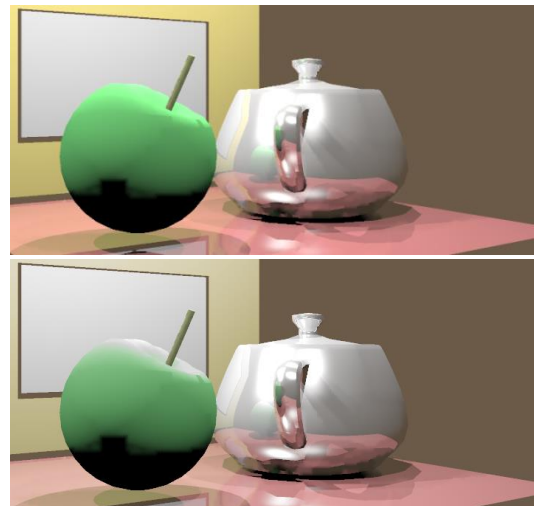


Fig. 4. Above is the original image, below is desaturated image.

But forcing user to tune the desaturation coefficient value manually is not a user-friendly solution. Some universal algorithm is required. It should allow automatic tuning of desaturation depending on the specifics of each particular image.

4. Methods of automatic parameters choosing

Desaturation should be applied only to the brightest parts of an image with brightness several times greater than the rest of it. So, the most optimal method which showed good results in most cases was setting some threshold value. Desaturation will not be applied at all for values below this value. Individual desaturation coefficient in this case increases linearly from 0 to 1 for brightness value from threshold to maximum correspondingly. Empirically it was established that half of the maximal brightness value of the image as the threshold value will provide acceptable result for most images. This is because dynamic range is usually so great that most part of the image has brightness several times lower than the maximal one.

But setting threshold value by the statistical parameters of the image revealed best results, including cases where dynamic range of the image doesn't require desaturation of any fragments. This method is based on the preposition that threshold value should be significantly higher than brightness of the most part of the image. Otherwise the desaturation just makes no sense because such images can be found neither while observing real world objects with human eyes, nor at the photos with a little bit correct exposure parameters. For example, some kind of average or matrix metering (in several points) is generally used for exposure setting in modern digital photo cameras instead of a single lighting measurement in some point [8]. It allows choosing such exposure parameters that most part of the image will have normal saturation. After several experiments there was elaborated the following method for calculating threshold value. Such brightness value should be used that certain number of image pixels would have brightness below this value. In our algorithm the pixel quantity had been defined as 75% from the total. Then this value is doubled and the result is used as the threshold. If the threshold value is higher than the maximum then desaturation is not applied at all.

After applying this method to the first image one can see almost no difference from the initial desaturation method. Only the table top became a bit more saturated at the edge which is far from the lamp. So it looks even more realistic than after applying the original algorithm, as it is seen at fig. 5:



Fig. 5. Desaturation with threshold based on statistical parameters of the image.

In fig. 6 there are shown the results of the applying the algorithm to another image. The tray and the teapot have desaturated highlights.

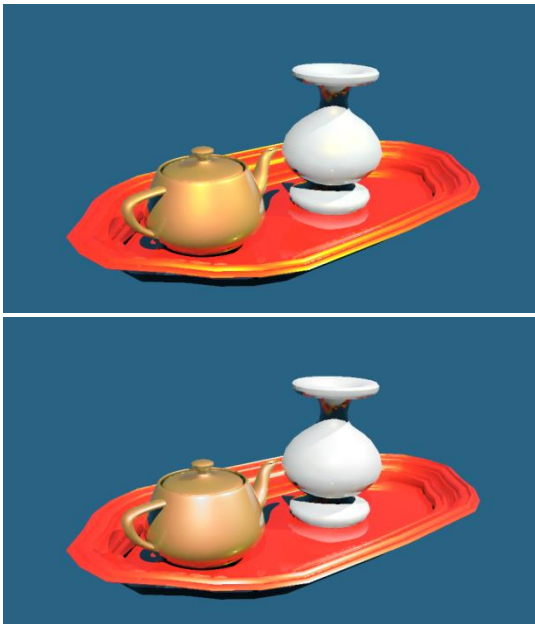


Fig. 6. Desaturated highlights on the tray and teapot.

As for images where original algorithm caused superfluous desaturation, like the apple from fig. 4, most often threshold becomes greater than the maximal brightness value, so desaturation is not applied at all.

During testing there was added one more improvement of the method. Only those pixels which have nonzero brightness are taken into account during calculating of the threshold value by the pixel. This is due to the fact that absolutely zero brightness at the images usually means that the image represent itself simulated rendering results and the background is black. So, this background contains no significant information and in fact is not related to the image itself.

5. Conclusion

Newly developed algorithm allows processing of high dynamic range images in so way that color saturation of the brightest highlights will be lowered as it is present in real life with taking into account specifics of human vision and photo technics. At the same time most images where such highlights are absent also looks realistic after processing with this algorithm.

6. Acknowledgements

This work was supported by RFBR, grants No 17-01-00363 and 19-01-00435.

7. References

- [1] Barladian, B.K., Voloboi, A.G., Galaktionov, V.A. and Kopylov, E.A., 2004. An effective tone mapping operator for high dynamic range images. *Programming and Computer Software*, 30(5), pp.266-272.
- [2] F. Ebner and M. D. Fairchild, 1998. Development and testing of a color space (IPT) with improved hue uniformity. In *Sixth Color Imaging Conference: Color Science, Systems and Applications*, pages 8–13.
- [3] D. Guo, Y. Cheng, S. Zhuo, T. Sim, 2010. Correcting over-exposure in photographs. *Correcting over-exposure in photographs*. IEEE Computer Society, 2010
- [4] Mantiuk, R., Mantiuk, R., Tomaszewska, A. and Heidrich, W., 2009, April. Color correction for tone mapping. In *Computer Graphics Forum* (Vol. 28, No. 2, pp. 193-202). Oxford, UK: Blackwell Publishing Ltd.
- [5] Tania Pouli, Alessandro Artusi, Francesco Banterle, Ahmet Oguz Akyuz, Hans-Peter Seidel and Erik Reinhard, 2013. Color Correction for Tone Reproduction. *Color and Imaging Conference, 21st Color and Imaging Conference Final Program and Proceedings*, pp. 215-220(6)
- [6] Reinhard, E., Stark, M., Shirley, P. and Ferwerda, J., 2002, July. Photographic tone reproduction for digital images. In *ACM transactions on graphics (TOG)* (Vol. 21, No. 3, pp. 267-276). ACM.
- [7] Alessandro Rizzi, Carlo Gatta, Daniele Marini, 2003. A new algorithm for unsupervised global and local color correction. *Pattern Recognition Letters*, Volume 24, Issue 11, July 2003, Pages 1663-1677
- [8] Sean T. McHugh, 2018. *Understanding Photography*. Cambridge University press.
- [9] Tumblin, J., 2000. Three methods of detail-preserving contrast reduction for displayed images (Doctoral dissertation, PhD thesis, Georgia Institute of Technology, 1999. <http://www.cc.gatech.edu/gvu/people/jack.tumblin>).