Saturated-Pixel Enhancement for Color Images

Di Xu, Colin Doutre, and Panos Nasiopoulos

Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, B.C., Canada
Email: {dixu, colind, panos}@ece.ubc.ca

Abstract—**We propose an algorithm to correct both luma and chroma of the saturated pixels in an overexposed image. Our method is based on the strong chroma spatial correlation between saturated pixels and their surrounding unsaturated area. We first identify the saturated areas in the image. Then, we partition these areas into regions with similar chroma, and estimate the chroma of each saturated region based on the chroma of its surrounding unsaturated region. Next, we correct the saturated R, G, or B color channels according to the estimated chroma and the unsaturated color channel(s) of the pixel. The last step involves smoothing of the boundaries between regions of different saturation scenarios. Both objective and subjective experimental results show that our algorithm is very effective in restoring the color of saturated pixels.**

I. Introduction

Recently developed high dynamic range (HDR) displays [1] have greatly extended the limited dynamic range of conventional cathode ray tube (CRT), liquid crystal display (LCD), and projector-based displays. In order to effectively display legacy LDR images and videos on HDR displays, inverse tone mapping schemes have been developed to extend the dynamic range of LDR images and videos. Legacy images and videos store only a small dynamic range of information due to the limitations of the capturing and display devices. The very bright and dark parts of a scene are clipped to the upper and lower displayable limits, respectively. As a result, information is lost. Over the last few years, special attention has been paid to the restoration of the clipped pixels [3]-[7], so that the enhanced clipped regions have higher dynamic range and look more realistic on HDR displays. All of the above schemes enhance only the luma, while the chroma enhancement is not considered.

For a saturated pixel, often not all three of the red (R), green (G), and blue (B) channels are clipped, nor does the same amount of clipping occur in each channel. If clipping changes the R, G, B color ratios of a pixel, then the result is color distortion. In fact, color distortion occurs very often. Although people are accustomed to the clipping effect in highlights, where the distorted color is desaturated and close to white, the distorted colors near the midtone produce a very noticeable and disturbing effect.

Little work has been done on correcting color distortion caused by clipping, as most methods for correcting clipped pixels only modify the luma channel [3]-[7]. One method that does attempt to fix color distortion is the statistical approach proposed by Zhang and Brainard in [8]. Their method exploits the correlation between the responses of RGB color channels at each pixel, and estimates the clipped color channel(s) using a Bayesian algorithm based on the unsaturated color channel(s) and the prior statistical distribution parameters. The method is proven to be effective for enhancing saturated pixels for images with a small saturated area; however, it is not as effective for images with large saturated areas.

In this paper, we propose an effective saturated-pixel enhancing algorithm, which restores both the luma and chroma of the clipped pixels. We exploit the strong spatial correlation in chroma between saturated pixels and their surrounding unsaturated pixels. Experimental results show that our algorithm outperforms the Bayesian algorithm [8] in both objective and subjective quality evaluations.

The rest of the paper is structured as follows. Section II describes our proposed method. The experimental results are presented in Section III and conclusions are given in Section IV.

II. Our Proposed Method

In this paper, we aim at restoring the lost information in over-exposed color images. Our approach is based on the presence of the strong spatial correlation in the chroma channels. The *YCbCr* color space is designed so that the chroma channels will be smooth in local regions for most images. It has been shown that utilizing the smoothness property of chroma is better than assuming luma is smooth [6]. Fig. 1 shows the normalized autocorrelation of *R*, *G*, *B*, *Y*, *Cb*, and *Cr* at lags of 0 to 25 pixels. Each point in the figure is an average value over the 24 true-color Kodak images [9]. From the graph, we can see that there is stronger autocorrelation for the *Cb* and *Cr* channels than the *R*, *G*, *B*, and *Y* channels. Exploiting the strong spatial correlations in the *Cb* and *Cr* channels has more potential than exploiting the correlations in the *R*, *G*, *B*, or *Y* channels. For this reason, in our approach we apply a chroma interpolation for the clipped pixels other than directly correcting the *R*, *G*, and *B* signals.

Our proposed method can be broken down into several steps. First, we identify the clipped areas. Then we partition each clipped area into smaller regions according to the chroma. We estimate the chroma in each clipped region by

This work was supported in part by grants from the Natural Sciences and Engineering Research Council of Canada (NSERC).

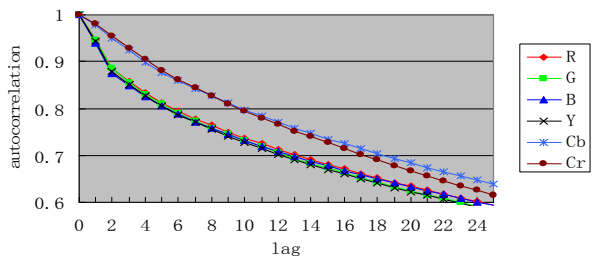

Figure 1. Normalized autocorrelation of *R, G, B, Y, Cb*, and *Cr* signals (average over 24 true-color Kodak images).

smoothly interpolating the chroma of the unclipped surrounding regions. Then we calculate *RGB* values for the clipped pixels based on the interpolated chroma. Last, we apply a smoothing process to the corrected *RGB* values. A detailed description of these steps is given in the following subsections.

A. Identify Clipped Areas

Before doing any enhancement, we first need to identify the clipped areas. One way of doing this is to simply select pixels that have the maximum value in any color channel (e.g., 255 for 8-bit per channel images). This simple approach often generates very small isolated clipped areas, and large clipped areas with small holes. This effect is caused by image noise. Therefore, we first apply a bilateral filter to remove the noise. Then, a threshold τ is applied to each color channel of a pixel to identify clipped pixels and channels. We experimentally selected τ to be 252.5 for 8-bit per channel images herein.

B. Partition Clipped Areas

The purpose of partitioning the clipped areas is to group the clipped pixels into regions with similar chroma before correcting the color for each region. We first partition the clipped areas into spatially disconnected regions, which probably belong to different surfaces and have different chroma. Each region may still contain multiple clipped objects that have different colors. Therefore the clipped regions are further segmented using the histogram-based multi-threshold algorithm presented in [10], applied to the chroma channels. After partitioning, we will have a number of clipped regions, and within each region the chroma should be similar for all pixels. This partitioning is essential for the subsequent color correction steps.

C. Correct Chroma

As explained earlier, we choose to indirectly correct the saturated *R*, *G*, or *B* values by first estimating the chroma of the clipped pixels. This is because the spatial correlation is higher in the chroma channels than in the *R*, *G*, or *B* channels. Therefore, a smooth chroma signal can be more accurately estimated based on its spatial neighbors. For this reason, we chose to estimate the chroma values in a clipped region by smoothly interpolating the chroma of neighboring unclipped pixels.

In order to correct the chroma of a clipped region, we first attempt to find an unclipped region with similar chroma. We select neighboring pixels with similar color to the clipped region as seed points. This is done by first choosing the unclipped or already corrected neighboring pixels with gradients of both chroma channels less than a threshold (experimentally, we determined 2.5 works well). Then, starting from each seed point, we apply a region growing algorithm shown in [10] to both *Cb* and *Cr*, and the intersection of the two obtained regions is a surrounding area with similar chroma to the clipped region. Since there may be small chroma variations within each clipped region, we take the union of all surrounding areas obtained from different seed points as the surrounding region for a clipped region.

The *Cb* and *Cr* values of the clipped region could be interpolated from its surrounding region. A problem arises from the fact that the surrounding region is irregularly shaped with some "missing" pixels which cannot be used in the interpolation because they are either clipped pixels, or non-clipped pixels with different chroma to the current clipped region. A common interpolation approach is to use convolution (filtering). However, traditional convolution does not work when there are missing samples within the convolution mask. For this reason, we use normalized convolution [11] instead, which allows for missing samples by adjusting the filter weights to use only the valid samples that fall within the mask. The normalized convolution can be expressed as follows:

$$\tilde{c}(x,y) = \frac{[c(x,y)\cdot m(x,y)] * h(x,y)}{m(x,y) * h(x,y)},$$

where the certainty map $m(x,y)$ is

$$m(x,y) = \begin{cases} 1, & \text{for unsaturated pixels in the surrounding region}, \\ 0, & \text{otherwise}, \end{cases}$$

The $c(x,y)$ and $\tilde{c}(x,y)$ represent the chroma channel signal (*Cb* or *Cr*) before and after the convolution, and $h(x, y)$ denotes the filter for performing the convolution. Here, a Gaussian filter with a standard deviation 5 is used as the function $h(x, y)$.

D. Correct RGB Values

We calculate the missing *R*, *G*, or *B* values in each clipped region based on the estimated *Cb* and *Cr* values (calculated in the previous step) and the unsaturated *R*, *G*, or *B* values in that region. We elaborate this correction process for the following three different scenarios.

1) Correct 2-channel saturated pixels: This is the most straightforward scenario. We know the conversion from RGB to YCbCr, introduced in the ITU-R BT.601, is:

$$Y = 0.2568R + 0.5041G + 0.0979B + 16 \quad (1)$$

$$Cb = -0.1482R - 0.2910G + 0.4392B + 128 \quad (2)$$

$$Cr = 0.4392R - 0.3678G - 0.0714B + 128. \quad (3)$$

When two channels are clipped, then one of the *R*, *G*, and *B* values, say *U* (which stands for the unsaturated channel), is known and the two clipped channels, say S_1 and S_2 (which stands for the saturated channels), are unknown and need to be solved for. The corrected values of the two saturated channels can be uniquely solved for using the two equations (2) and (3). Therefore, we have:

$$\widetilde{S}_1 = f_1(U, Cb, Cr),$$
$$\widetilde{S}_2 = f_2(U, Cb, Cr),$$

where f_1 and f_2 are linear functions of U, Cb, and Cr that can be derived from (2) and (3), and $\widetilde{S}$ denotes the corrected value of color channel S.

2) Correct 1-channel saturated pixels: This is similar to correcting 2-channel saturated pixels. Since there is only one unknown value S, and two equations, (2) and (3), the value can be estimated twice by using the corrected Cb and Cr, respectively, as well as the two unsaturated channel values U_1 and U_2. Then, we simply take the average of the two estimations as the corrected value of the saturated channel. The estimation process can be described as:

$$\widetilde{S} = \frac{S_1 + S_2}{2}, \qquad \text{where}$$
$$S_1 = f_3(U_1, U_2, Cb),$$
$$S_2 = f_4(U_1, U_2, Cr),$$

and the functions f_3 and f_4 are derived from (2) and (3), respectively.

3) Correct 3-channel saturated pixels: There are three unknown variables in this scenario. Hence, three equations are needed to solve the corrected R, G, and B values. We first estimate the luma Y value of the 3-channel saturated pixels based on the surrounding region. We fit the clipped region and its surrounding area with a 2D Gaussian function. Unlike many other surface-fitting methods, we do not enforce any assumptions on the location or rotation of the 2D Gaussian function. By not assuming the center of the Gaussian function as the centroid of the clipped region, we are able to handle more general and sophisticated clipping cases. For example, our model works well for the situation where the brightest spot is not located near the center of the clipped region and the surrounding region only partially encloses the clipped area. In general, a 2D Gaussian function is of the form:

$$f(x,y) = Ae^{-[a(x-x_0)^2+2b(x-x_0)(y-y_0)+c(y-y_0)^2]} + B,$$

where A, B, a, b, c, x_0, and y_0 are the parameters, and $\begin{bmatrix} a & b \\ b & c \end{bmatrix}$ is positive definite. The optimization can be solved with a standard least squares fitting algorithm. Once the parameters are estimated, the luma Y at the 3-channel saturated pixels can be computed by evaluating the Gaussian function. In the end, the corrected RGB values $\widetilde{R}$, $\widetilde{G}$, and $\widetilde{B}$ are solved using (1), (2), and (3).

E. Smooth Enhancement

Often, there are small jumps of enhanced values between adjacent 1-, 2-, and 3-channel saturated regions. A smoothing process near the region boundaries is needed to reduce disturbing contours and obtain natural looking enhanced images. Among the pixels near the region boundary, we choose to adjust those with more saturated channels (i.e., the pixels that have relatively higher estimation errors). For each pixel P that needs to be adjusted, a linear combination of the estimated values (from the previous steps) at pixel P and its nearby area in the less saturated region is computed in order to replace the previously estimated value at P. The weight of value at P in the linear combination is proportional to the distance between P and its nearest pixel in the less saturated region.

III. EXPERIMENTAL RESULTS

In this section, we present some experimental results to show the effectiveness of our proposed method for enhancing the clipped pixels. In our experiment, we use conventional 24 bits per pixel LDR color images for our tests.

We generate the clipped images by clipping the R, G, B values that are greater than a threshold (e.g., 255*0.8). Then, we enhance the clipped images using our proposed method as well as the Bayesian algorithm [8], the only color correction algorithm for clipped pixels that we are aware of. We compute the PSNR values (averaged over R, G, and B channels) and the S-CIELAB metric [12](averaged over the saturated pixels) of each test image for: 1) the clipped image with no correction, 2) the enhanced image generated by the Bayesian algorithm, and 3) the enhanced image produced by our proposed algorithm. The quality comparison for a set of representative images is listed in Table I. Note that the S-CIELAB metric is a distance measure; a lower value means better quality. From the table, we can see that while both the Bayesian algorithm and our proposed algorithm correct the color and enhance the clipped pixels, our proposed method outperforms the Bayesian algorithm by an average of 4.10 dB in PSNR and 0.72 in S-CIELAB over the test images. Our method performs well especially for images with large portion of clipped areas.

TABLE I. QUALITY COMPARISON

Image	PSNR (in dB)			S-CIELAB		
	Clipped	*Bayesian*	*Ours*	*Clipped*	*Bayesian*	*Ours*
girl	42.2	39.7	50.7	0.98	1.51	0.33
landscape	25.7	28.7	29.5	1.87	1.41	1.29
baby_girl	32.2	29.1	36.4	1.24	1.86	0.68
mountain	30.0	34.3	37.0	1.65	0.91	0.53
shoes	25.3	32.9	32.4	2.00	1.02	0.77
sunset	21.7	20.8	27.2	3.04	3.47	1.72
kodim03	34.3	35.2	39.7	2.30	2.25	1.07
kodim05	33.6	35.7	37.1	2.25	1.83	1.17
kodim06	25.2	25.1	32.5	3.44	1.97	1.56
kodim12	28.4	33.7	33.2	2.08	0.81	1.08
kodim16	35.1	35.9	41.7	2.16	1.98	0.89
kodim21	32.4	33.6	36.8	3.03	2.51	1.49
kodim23	29.6	31.2	34.9	1.70	1.40	0.91
Average	**30.4**	**32.0**	**36.1**	**2.14**	**1.76**	**1.04**

Figure 2. Results of clipped pixel enhancement for images baby_girl, kodim03, and kodim16. For each row, we show (from left to right) the original image, clipped image, clipped areas superimposed on the image luma, enhanced image using Bayesian algorithm, and enhanced image using our proposed algorithm.

Due to limited space, Fig. 2 shows the representative resulting images and is used for evaluating the subjective quality of the enhanced clipped pixels and in turn the overall image. For each image, we show (in reading order) the original image, clipped image, clipped areas superimposed on the image luma, enhanced image using the Bayesian algorithm, and enhanced image using our proposed algorithm. Pixel values of images in each group are linearly scaled using the same scaling factor to realize the maximum display contrast. From Fig. 2 we can see that all clipped images have color distortions due to over exposure. The Bayesian algorithm corrects color for most clipped regions. However, it over-corrects the color in some clipped regions and results in further color distortion. An over-correction example can be seen in the background area of the "baby_girl" image. These artifacts happen when the color properties of the clipped region are different from the statistical properties of the unclipped regions in the image. Distortion usually occurs when the images do not possess much color variety or a large portion of clipped pixels exist. Compared to the Bayesian algorithm, our algorithm gives comparable or better subjective quality for images in Fig. 2 and other test images, without notable artifacts.

IV. CONCLUSIONS

In this paper, we have proposed an effective method for enhancing saturated pixels in color images. We take advantage of the strong correlation between the chroma of the saturated pixels and their surrounding unsaturated pixels. Our method greatly reduces the color distortion caused by clipping. Our proposed method outperforms the Bayesian algorithm by an average of 4.10 dB in PSNR and 0.72 in S-CIELAB over a set of 13 test images. Subjective results also show that the enhanced images generated by our method are visually closer to the unclipped ground truth images than both the clipped images and the enhanced images produced by the Bayesian algorithm.

REFERENCES

[1] H. Seetzen, W. Heidrich, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs, "High dynamic range display systems," ACM Transactions on Graphics, vol. 23, no. 3, pp. 760-768, August 2004.

[2] O.A. Akyüz, R. Fleming, B.E. Riecke, E. Reinhard, and H.H. Bülthoff, "Do HDR displays support LDR content? A psychophysical evaluation," ACM Transactions on Graphics, vol. 26, no. 3, Jul. 2007.

[3] L. Meylan, S. Daly, and S. Süsstrunk, "The reproduction of specular highlights on high dynamic range displays," In Proc. of the 14th Color Imaging Conference, 2006.

[4] L. Meylan, S. Daly, and S. Süsstrunk, "Tone mapping for high dynamic range displays," In Proc. IS&T/SPIE Electronic Imaging: Human Vision and Electronic Imaging XII, vol. 6492, 2007.

[5] F. Banterle, P. Ledda, K. Debattista, and A. Chalmers, "Inverse tone mapping," In Proc. of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia (Kuala Lumpur, Malaysia, Nov. 29 – Dec. 02, 2006). GRAPHITE '06. ACM, New York, NY, pp. 349-356.

[6] A.G. Rempel, M. Trentacoste, H. Seetzen, H.D. Young, W. Heidrich, L. Whitehead, and G. Ward, "LDR2HDR: On-the-fly reverse tone mapping of legacy video and photographs," ACM Transactions on Graphics (Proc. SIGGRAPH), vol. 26, no. 3, 2007.

[7] P. Didyk, R. Mantiuk, M. Hein, and H.-P. Seidel, "Enhancement of bright video features for HDR displays," Comput. Graph. Forum, vol.27, no.4, pp. 1265-1274, 2008.

[8] X. Zhang and D.H. Brainard, "Estimation of saturated pixel values in digital color imaging," Journal of the Optical Society of America A, Optical Society of America, 2004, vol. 21, no. 12, pp. 2301-2310.

[9] Kodak Lossless True color image suite, available at http://r0k.us/graphics/kodak/.

[10] R.C. Gonzalez and R.E. Woods, Digital Image Processing. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1992.

[11] H. Knutsson and C.-F. Westin, "Normalized and differential convolution: methods for interpolation and filtering of incomplete and uncertain data," Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 515-523, New York City, USA, Jun. 1993.

[12] X. Zhang and B.A. Wandell, "A spatial extension of CIELAB for digital color-image reproduction," Journal of the Society for Information Display, vol 5, no. 1, pp. 61-63, Mar. 1997, doi:10.1889/1.1985127.