Robust Color Correction for HDR Images with Colored Light Sources and Varying Textures

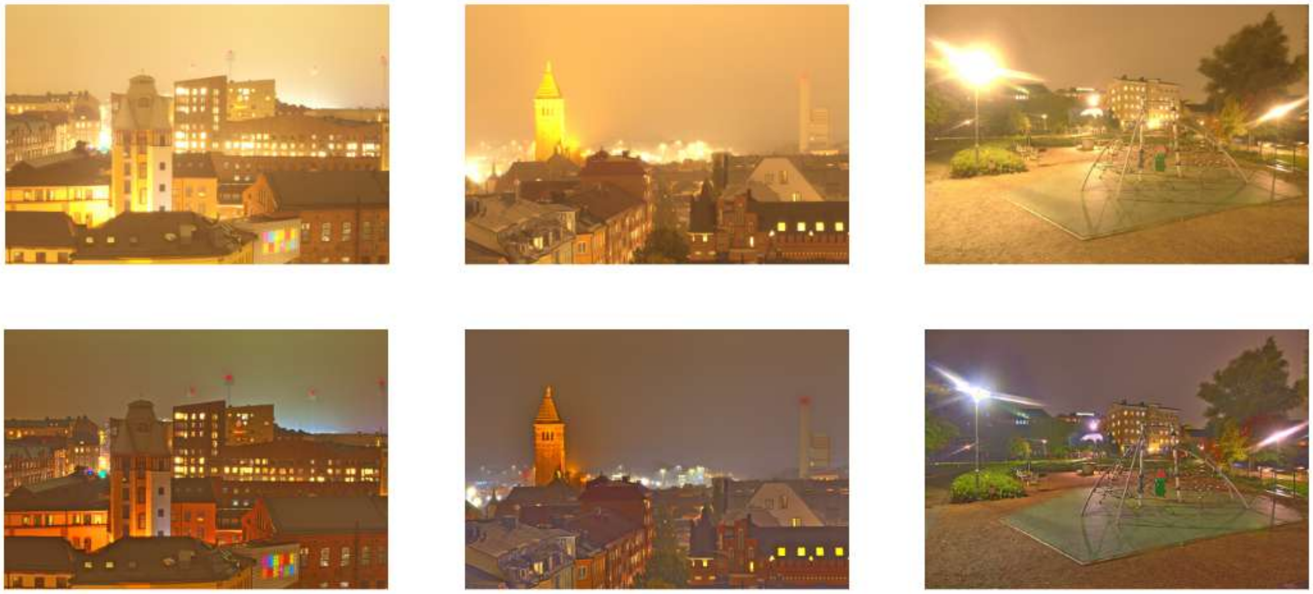


Figure 1: The tone mapping results of our color correction method; The first row represents the single low dynamic range (LDR) images provided in SIHDR dataset as references[9]. The second row represents the corresponding results achieved through our method. This approach enhances the realism in images with a dominant color, aligning the visual representation more closely with human perception. It can also handle the images with complex lighting conditions and varying textures.

ABSTRACT

Mapping from HDR to LDR is widely used for Computer Vision and Computer Graphics applications. State-of-the-art approaches focus on mapping intensity for each pixel. Generally, post-processing follows tone mapping; however, performing color correction before tone mapping is advantageous due to the broader range for correction available in HDR space. We developed a novel scheme to incorporate color constancy into the RGB values of the HDR images. It removes the negative effects caused by light source. For images with varying textures such as the ones with the simple scenes and complex scenes, the amount of correction also is dependent on the textures. We further improve the RGB values with the information that describes how much the chrominance varies. This approach improves the realism and naturalness of images, particular those with a dominant color. We quantify the quality of the LDR output using state-of-the-art objective evaluation metrics. Our method can be seamlessly integrated into the existing HDR to LDR conversion pipeline as a pre-processing method for the tone mapping.

CCS CONCEPTS

• **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

KEYWORDS

HDR, Tone mapping, Color Constancy, Color correction

ACM Reference Format:
. 2018. Robust Color Correction for HDR Images with Colored Light Sources and Varying Textures. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Image Signal Processing (ISP) has recently become a prominent subject in Computational Photography.Utilizing the ISP pipeline enhances visual effects, addressing the limitations of optical camera sensors. A typical ISP pipeline comprises processes such as demosaicing, white balance, denoising, sharpening, and tone mapping.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

Among those procedures, the white balance and tone mapping are particularly crucial because they significantly affect the final colors and mimic chromatic adaptation and luminance adaptation[7] in human visual system. In conventional ISP pipelines, these processes are treated as two individual parts. Additionally, each step in ISP represents a separate research area. The white balance is usually an early process in the ISP pipeline, applied to raw image formats outputted from a camera's sensor. Conversely, tone mapping often occurs towards the end of the pipeline, leading to minimal interaction between these two processes.

The tone mapping is also used for processing the High Dynamic Range(HDR) image format to display colors effectively on standard screens.HDR images are typically created through post-processing, which is to combine Low Dynamic Range (LDR) images of different exposures from the sensor. If there is a color cast in the LDR image due to white balance or other factors like a dominant yellow tint around the image, it often persists in the HDR image ,primarily due to chrominance rather than luminance. This can result in a color cast in the LDR images following tone mapping. Consequently, some color correction modules are implemented post-tone mapping. There are also color-adaptive tone mapping techniques available. However, performing color correction prior to tone mapping might be more advantageous, as it allows for a broader scope of color correction within the HDR space.

We propose a perception-based pre-processing method that integrates color constancy by correcting both illumination and texture effects. his method can be easily incorporated into the HDR to LDR conversion pipeline. After applying our technique to HDR images, the tone mapping results are significantly improved, mitigating negative effects such as dominant colors caused by colored light sources. We evaluate it by using the state-of-the-art TMQI-II[13] evaluation metrics. It performs well in challenging scenarios, such as nighttime scenes with complex lighting conditions. In cases without a dominant color, our method does not alter the tone mapping results significantly.

2 PREVIOUS WORK

2.1 Tone mapping and color correction

Tone mapping is essential for displaying High Dynamic Range (HDR) images on standard displays and forms a module in modern Image Signal Processing (ISP) pipelines. There are subtle differences in application: the former involves post-processing, while the latter processes images directly from the camera sensor. Our focus is the first one. The tone mapping operators(TMOs) have been widely explored, including global tone mappers, local tone mappers, perceptive tone mappers, AI-based tone mappers, and so on.However, there's less emphasis on color adaptive tone mappers. Li et al. proposes a clustering-based content and color adaptive method[11], but the limitation is that it is a learning-based method and does not learn on the dataset containing the challenging scenes of complex lighting conditions. The method is not designed for those scenes and is not robust. There are also color correction methods for tone mapping. Furthermore, color correction methods for tone mapping, like the one proposed by Mantiuk et al. [14], often rely on simplistic empirical equations and struggle with complex scenes. These methods are applied after LDR tone mapping, but performing color correction beforehand in the HDR space offers more flexibility. Therefore, we propose a framework to do the color correction on HDR data while robustly handling the challenging scenes. (more literature should be added)

2.2 Color constancy

Color constancy is a feature of the human visual system that ensures the perceived color of objects remains relatively constant under varying illumination conditions. The white balance is a technique to achieve color constancy for the captured image by illumination correction. But it is not the only way to achieve color constancy. It is a complicated mechanism of our human visual system. This implies that we can simulate this mechanism to achieve color constancy when processing images. There is a lot of work on how to achieve color constancy. While there's extensive research on achieving color constancy in raw images, studies focusing on HDR and SRGB images are less common. Our work concentrates on incorporating color constancy as a pre-processing way before tone mapping. It's important to note the difference from Afifi et al.'s work [1], which corrects incorrectly white-balanced images in SRGB space. Our approach applies color constancy to HDR images.

Most color constancy methods involve illumination estimation[2][8][19]. Our model also involves illumination estimation. There are many ways to estimate the illumination color.

2.3 Color appearance model

The color appearance model aims to simulate how the human visual system perceives color, including chromatic and luminance adaptation processes. These processes can be correlated with color constancy and tone mapping at a technological level. Recently, there has been a focus in color science on developing color appearance models for HDR data [17][12]. However, these models are challenging to code due to their reliance on numerous parameters linked to real physics data, which causes potential errors. The color appearance model can also achieve the cross-media reproduction. An alternative model proposed by Kim et al. [10] from the graphics community, comes with source code, offering a more accessible approach to modeling perceptual color accurately. Notably, we disable the CAT (Chromatic Adaptation Transform) option and extract chromaticities, as our method aims to replace it with our own color constancy model.

3 BACKGROUND

3.1 Chromanaticities Computation

In this section, we will discuss Chromaticities, the domain in which our manipulations take place. There are various chromaticity spaces, such as CIELAB, XY-chromaticity, and CAM-based chromaticities. For our purposes, we have chosen to use the XLR-CAM based chromaticity as outlined in Kim et al. [10]. This model is particularly suitable for our task as it is an HDR-based CAM model. Additionally, its implementation is feasible for us since the source code is accessible.We will now explain how chromaticities are derived in this model.

Initially, the model transforms pixel values from the RGB color space into the XYZ space using the following equation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

Subsequently, the model converts the color space from XYZ to LMS:

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \underset{3\times 3}{\mathrm{HPE}} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{2}$$

The model then applies a biologically-inspired mapping curve to adjust the luminance to a lower range, This simulates the luminance adaptation of our eyes:

$$L' = \frac{L^{n_c}}{L^{n_c} + L_a^{n_c}}, M' = \frac{M^{n_c}}{M^{n_c} + L_a^{n_c}}, S' = \frac{S^{n_c}}{S^{n_c} + L_a^{n_c}} \tag{3}$$

where

$$L_a = \exp(\mathrm{mear}(\log Y)) \tag{4}$$

The L_a simulates the environment light intensity.

The achromatic response is then obtained using the three cone responses:

$$A = \left(40L' + 20M' + S'\right)/61 \tag{5}$$

Finally, two chromaticities are derived, representing Redness-Greenness and Yellowness-Blueness respectively:

$$a = \frac{1}{11}\left(11 \cdot L' - 12 \cdot M' + S'\right) \quad \text{Redness - Greenness} \tag{6}$$

$$b = \frac{1}{9}\left(L' + M' - 2S'\right) \quad \text{Yellowness - Blueness} \tag{7}$$

It is important to note that in this model, the CAT (Chromatic Adaptation Transform) correction is skipped to obtain the chromaticities. This is possible due to the flexibility offered by the source code.

In the upcoming sections, our method primarily manipulates within this a-b chromatic space.

3.2 Color constancy

There are many ways to frame the color constancy problem. A widely used method is described in equation 8. In this equation, I represents the image color, L represents the illumination color and W represents the image color after color constancy correction. This method is an approximation[6] but has been proved as an effective way[4][5][21] to frame the color constancy problem.

$$I = W \times L \tag{8}$$

In section 4.3.1, we will modify this equation for our color correction to combine the texture effect into the illumination color correction.

In addition, there have been many ways to find the illumation color so far. In conclusion, they aim to find the representative point of the chrominance space as the reference illumination color, which is to find the most frequent chromaticity. For our method, we use a simple reference point selection, which is to find the cluster centroid of the chrominance space. The cluster centroid is computed by the following equations:

$$C = \left(\frac{\sum_{x\in X'} x}{|X'|}, \frac{\sum_{y\in Y'} y}{|Y'|}\right) \tag{9}$$

Where $|\cdot|$ represents the cardinality. X represents the set of x-coordinates, Y represents the set of y-coordinates, X' represents the subset of X such that each element x in X' satisfies $\mu_X - \sigma_X < x < \mu_X + \sigma_X$. Y' represents the subset of X such that each element y in Y' satisfies $\mu_Y - \sigma_Y < y < \mu_Y + \sigma_Y$. This approach is effective and also easy to compute.

To get the RGB color value of the illuminant, we combine this chromaticity with the average achromaticity A which is computed by averaging the A values of all the pixels of the original HDR image. Finally, we use the inverse transformation to transform [A, a,b] into [L, M, S], then to RGB. The transformation matrices are derived from experimental data and standardized by International Commission on Illumination (CIE).

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1.0000 & 0.3215 & 0.2053 \\ 1.0000 & -0.6351 & -0.1860 \\ 1.0000 & -0.1568 & -4.4904 \end{bmatrix} \begin{bmatrix} A \\ a \\ b \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{11}$$

4 APPROACH

In the human visual system, the perception of texture and color involves complex neural processes. When we observe a scene, our eyes and brain work together to filter Texture Information and adapt to the lighting conditions. The former one is to recognize and interpret texture details without significantly altering the perceived color information. This is akin to 'seeing through' the texture to understand the underlying color. It is worth to note that this minimization of the impact of texture on the perception of color is not to "smooth" the image. It is to minimize the texture effect on the chrominance to ensure a consistent perception of the surface color. The latter is to adjust to the color of the illuminant, effectively 'discounting' the color cast caused by the light source to perceive the true colors of objects. Since our task is image processing, the term "texture" here also includes multiple varying light sources and also multiple different colored objects. Those two mechanisms are both related to the color constancy which is the ability of our visual system, to perceive colors consistently under varying lighting conditions. Our model is designed to simulate and those two mechanisms and forms a two-step process, providing a realistic effect.

4.1 Chromaticity collection

our operational space is chrominance space, as tone mapping focuses on mapping luminance values. Then our color correction will manipulate the chrominance before the tone mapping. We first collected the chromaticities of images. We adopted the chromaticities of the xlrCAM[10] to utilize the a and b models from xlrCAM [10], It is derived from HDR data and thus more suitable for our task than other spaces like CIELAB.. We then divided the image into regular grids and get the average chromaticities of the grids. The advantage of it is that it is more efficient and reducing the feature size is beneficial for us to analyze and process. In our experiment, we used 1000 grids.

4.2 Texture Effect Removal

We then address texture correction. Texture analysis is conducted first. We calculate the texture effect based on the texture scale(vertical and horizontal) , color difference of the adjacent pixel neighbors and spread out ratio of the 2d chrominance space of the image. The texture scale and color difference for computing the texture measure is inspired by work[18]. The texture measure can be computed in the following equation.

$$T = w_1 \cdot S(C, s_i) + w_2 \cdot D(C, s_i) + w_3 \cdot R(C, s_i) \tag{12}$$

Where T denotes the total texture measure. w_1, w_2, and w_3 denote the weights for each term. c denotes the original chrominance of the image. s_i denotes each superpixel i. S indicates the texture scale, D represents the color difference and R represents the spread out ratio. In our experiment, w1 and w3 are set by 1 and w2 is set by 150 related to their number scales.

We then modulate chromaticities a and b using the texture measure to minimize texture impact on chrominance. The modulation factor γ is calculated using a sigmoid function, defined as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{13}$$

Then the modulation factor is calculated as below:

$$\gamma = \gamma_0 + k * (1 - \sigma(T)) \tag{14}$$

Where the γ_0 is the base γ, which is set by 0.7 in our experiment. The k is a scale factor which is set by 0.4 in our experiment.
Then, we can get the modulated chromaticities a_m and b_m:

$$\begin{aligned} a_m &= a \times \gamma \\ b_m &= b \times \gamma \end{aligned} \tag{15}$$

Then we will illustrate how we compute the texture measure T.

4.2.1 Texture Scale Computation.

Texture Scale Computation. We calculated the texture scale by doing the spatial frequency analysis. To align with the observation in work[18], we calculated the horizontal and vertical frequencies of the 2d chromaticities feature data. We use a fast fourier transform to get all the frequency information of the data and then get the horizontal and vertical components.

$$\begin{aligned} F(u, v) &= \text{FFT2}(f(a, b)) \\ F_s(u, v) &= \text{FFTShift}(F(u, v)) \\ M(u, v) &= 20 \log(|F_s(u, v)| + 1) \end{aligned} \tag{16}$$

Here, FFT2 represents the 2D fast fourier transform. $f(a, b)$ represents the 2D data input, $F(u, v)$ is the 2D Fourier transform of f, F_s is the shifted Fourier transform, and M is the magnitude spectrum on a logarithmic scale.
Next, we calculate the horizontal and vertical components of the magnitude spectrum M, whcih can be denoted as H and V, respectively.

$$H(u, v) = \begin{cases} M(u, v) & \text{if } v = \frac{\text{width of } M}{2} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

$$V(u, v) = \begin{cases} M(u, v) & \text{if } u = \frac{\text{height of } M}{2} \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

.
We then compute the scale measure by applying a weighted sum of the mean values of the horizontal and vertical component scales:

$$S = w_1 \times \text{mean}(H) + w_2 \times \text{mean}(V) \tag{19}$$

where w_1 and w_2 are both 0.5 in our experiments.

4.2.2 Color Difference Computation.

Color Difference Computation. Then we calculate the local color difference for every grid. We will illustrate this algorithm by making the regular grids at the beginning, We calculate the mean color difference for every regular grid by using the total chrominance differences calculated by Euclidean distance of 8 neighbors. Algorithm.1 shows our color difference computation.

Algorithm 1 Grid-Based Color Difference Calculation Using Boundary Pixels

```
Require: $grids$, $average\_a\_$, $average\_b\_$
Ensure: Average local color difference for regular grids
  Initialize $color\_differences$ as an empty list
  for each grid label $L$ in $grids$ do
    Compute $mask$ for $L$
    Compute average chromaticities $a_L$ and $b_L$ for $mask$
    Find boundary pixels of $L$
    for each boundary pixel $(i, j)$ of $L$ do
      for each neighbor position $(i + dy, j + dx)$ in a $3 \times 3$ grid around $(i, j)$ do
        if $(i + dy, j + dx)$ is a valid position and belongs to a different grid $N$ then
          Compute average chromaticities $a_N$ and $b_N$ for $N$
          $color\_diff \leftarrow \sqrt{(a_L - a_N)^2 + (b_L - b_N)^2}$
          Add $color\_diff$ to $color\_differences$
        end if
      end for
    end for
  end for
  if $color\_differences$ is not empty then
    return mean($color\_differences$)
  else
    return 0
  end if
```

4.2.3 Spread out Ratio Computation.

Spread out Ratio Computation. Additionally, we compute the spread-out ratio of the a and b chrominance values across all grids of the image. This ratio intuitively reflects the texture level of the image.
Working with two arrays, a and b, the function determines a refined spread-out ratio through the following procedures:

$$\sigma_a = \text{std}(a),\ \sigma_b = \text{std}(b) \tag{20}$$

$$R_a = \max(a) - \min(a),\ R_b = \max(b) - \min(b) \tag{21}$$

$$C_a = \frac{\sigma_a}{R_a} \text{ if } R_a \neq 0 \text{ else } 0,\ C_b = \frac{\sigma_b}{R_b} \text{ if } R_b \neq 0 \text{ else } 0 \tag{22}$$

$$\alpha_{\text{normalized}} = \text{clip}\left(\frac{C_a + C_b}{2} \times s, 0, 1\right) \tag{23}$$

Where σ_a and σ_b represent the standard deviations of the values in a and b in (20). In (21), R_a and R_b are their respective ranges. In (22), C_a and C_b are the combined measures for each array, computed as the ratio of the standard deviation to the range of the values. The final output, $\alpha_{\text{normalized}}$ described in (23), is the normalized average of these combined measures, scaled by a factor (denoted as s) and clipped to the range [0, 1].

At the end of this section, we show the images before removing the rexture effect and after in Figure **2**. We also show the chromaticity change after this operation.

4.3 Color correction using Illumination Estimation

4.3.1 Texture based blending function. Next, we further modify the equation 8 and propose a blending function taking the texture information into consideration. The idea behind modulating color correction with texture information is based on the assumption that areas of an image with more texture might be less influenced by the color of the light source and hence might require less color correction for color constancy. In other words, the hypothesis is that in areas with more texture, the perception of color is less dominated by the color of the light source. This could be due to the fact that textured areas inherently have more color variation, which makes the color cast from the light source less noticeable or significant. So We propose a blend model:

$$I' = \frac{I_0}{1 + k \times ((\text{R}F \times s - 1) \times (1 - \alpha))} \tag{24}$$

In this equation, the alpha represents the spread out ratio we calculated in 4.2.3, which represents the texture level. I' and I_0 represent the image color after and before the illumination color correction, respectively. If alpha is close to 1, I' is close to I_0, indicating that it is less affected by the illumination color if it is more textured. RF represents the RGB values of the reference illuminant. s denotes and the scale factor. It is used to amplify the influence of the ratio since the RF is range in 0-1 in our setting by normalization.k is a control factor which can control the ratio of color correction. the For some images, if there is no obvious color diviation like daytime image, the k can be small to decrease the ratio of the color correction.

The intuitive understanding under this blend function is that in practical terms, think of it as if the areas with rich textures are less "washed out" or "biased" by the light source color, and thus need less correction. Uniform areas, like a plain wall, are more likely to visibly show the color influence of the light source and hence need more correction.Note that this operation can be seen as complementary to the previous texture removal method. One is used to ensure chromaticity consistency across textures, while the other can be applied to tailor illumination correction to the characteristics of the texture in the image.

4.3.2 Dynamic range based scaling factor. Here we calculate the scaling factor and make it smart. Since our color constancy correction is on HDR images, we make this scaling factor adapt to the HDR dynamic range. It ensures that regions with higher dynamic ranges receive proportionally more correction, aligning with the greater need for adjustment in these areas. This idea is also inspired by the paper[18]. In this article, the perceived color amount is in a linear relationship with the texture scale. Both texture scale and dynamic range deal with spatial variations within an image. While texture scale looks at spatial variation in terms of patterns and their frequency, dynamic range looks at the variation in luminance levels.

Here is the calculation for the scaling factor:

$$L = 0.2225 \cdot I_R + 0.7169 \cdot I_G + 0.0606 \cdot I_B \tag{25}$$

This equation is to calculate the luminance, which is aligned with calculating the Y from RGB to XYZ space in 3.1.

$$DR = P_{98}(L) - P_2(L) \tag{26}$$

Where DR means the dynamic range of a region. The P representation means the percentile of the luminance, aiming for a more robust way to calculate the dynamic range to avoid outliers and abrupt changes in dynamic range.

$$s = c \cdot \log(d \cdot DR + 1) \tag{27}$$

In this equation, c is set by 6, and d is set by 8 in our experiment. The addition of 1 is to avoid the negative value.

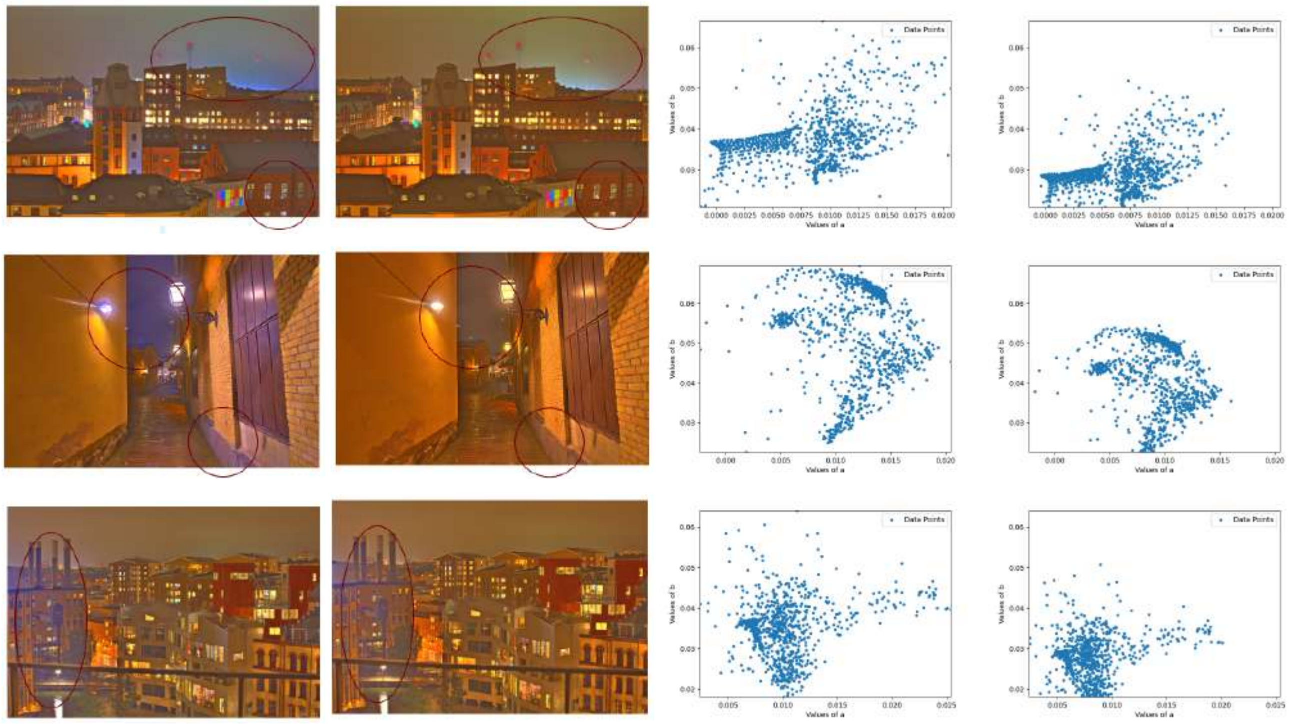
5 EXPERIMENT

In this section, we will introduce the datasets we used, explain our evaluation metrics and show our results for the evaluation metric sores and comparison with other color adaptive tone mapping methods.

5.1 Dataset

5.1.1 SI-HDR dataset. First, we use the SI-HDR dataset[9] for evaluation. This dataset contains 181 RAW exposure stacks selected to cover a wide range of image content and lighting conditions. Each scene is composed of 5 RAW exposures and merged into an HDR image using the estimator that accounts photon noise. A simple color correction was applied using a reference white point and all merged HDR images were resized to 1920×1280 pixels. This means that the white balance is not set well for the images, thus this dataset is very suitable for our tasks. Furthermore, the dataset contains 66 nighttime images and 120 daytime images. We found that the nighttime HDR images are really useful for our tasks since they contain complicated lighting conditions and varying textures.

5.1.2 LVZ-HDR dataset. Next,we used the LVZ-HDR dataset[15]. Each HDR image in the proposed LVZ-HDR Benchmark dataset is formed by fusing 5 LDR images of the same scene taken at multiple exposure brackets, -2EV, -1EV, 0EV, +1EV, +2EV, using Photoshop HDR tools. The images were taken using an Olympus E-P1 camera. The LVZ-HDR benchmark dataset consists of 456 High Dynamic Range images taken from different states, cities, time of the day and most importantly covering a wide range of sceneries. Images in the LVZ dataset are grouped into four broad scene categories: Indoors (71 images), Nature (173 images), Nighttime (80 images), and River-side-Sunset (133) images. However, this dataset has less



(a) (b) (c) (d)

Figure 2: (a): The tone mapping results with the same illumination correction and setting except the chromaticity modulation. (b): The tone mapping results with the same illumination correction and setting with the chromaticity modulation. (c): The scatterplot of the a,b chrominaticities space with the chromaticity modulation. (d): The scatterplot of the a,b chrominaticities space without the chromaticity modulation. From the differences of each image annotated by the red circle, our method makes the color more consistent by removing the texture effect. For the high textured(the chrominance variation is higher by looking at the scatterplots) images of the first row and second row, the color correction by removing the texture effect is obvious. For the low textured(the chrominance variation is lower by looking at the scatterplots) image of the third row, the color correction by removing the texture effect is fewer.

quality since there are many images containing considerably visible noise.

5.2 Evaluation Metrics

We used the state-of-the-art evaluation metrics TMQI[20] and TMQI-2[13] for evaluation final tone mapping results. Those two evaluation metrics evaluate the structural fidelity and statistic naturalness. Among them, the parameters of the statistic naturalness is derived by user study thus this term is especially useful for our color evaluation.

We also make a comparison for the naturalness part of the TMQI and TMQI-II individually. The naturalness portion is a psychophysical metric for the color, it is suitable for our task since our method is to mainly change the chrominance of the images.

5.3 Result Comparison

We further did the experiment to plug our color correction model into different tone mapping methods. We plugged it into two simple tone mapping methods, Storm model[3] and Reinhard model[16] . We selected those two models because they are simple, The storm only maps the intensity and the so they will not affect the chrominance correction of our method.The scores show that although we use simple tone mapping, the results out perform the state-of-the-art color adaptive tone mapping methods in challenging scenes.

We also compare our results with two color adaptive tone mapping models, The clustering based content and color adaptive tone mapping[11] results,(c): The xlrCAM tone mapping results of [10]. The result for the metric scores are shown in the Table.**1** and Table.**2**

5.4 Results analysis

From the Table.**1** and Table.**2**, we see that our method outperforms the methods we compare with by easily plug in our module controling the k in the 24 . For some challenging scenes like nighttime, indoors, and river side sunset, our method shows an obvious naturalness improvement.

5.5 Results before adding module and after

There should be image results here.

	TMQI-2 total score					
	Li2018[11]	xlrCAM[10]	Storm only	Strom+ours	Reinhard	Reinhard+ours
SIHDR(daytime)			60.16	60.23		
SIHDR(nighttime)			40.52	40.67	47.83	46.46
LVZ-HDR(nighttime)			56.50	57.28	52.85	51.30
LVZ-HDR(nature)			77.36	77.43	60.15	60.15
LVZHDR(indoors)			76.99	76.95	65.64	65.79
LVZHDR(riverside-sunset)			72.00	71.74	55.87	55.80

Table 1: Comparison among methods on TMQI-2 total score across datasets.

	TMQI-2 naturalness					
	Li2018[11]	xlrCAM[10]	Storm only	Strom+ours	Reinhard	Reinhard+ours
SIHDR(daytime)			38.7	39.1		
SIHDR(nighttime)			0.97	3.90	23.67	21.71
LVZ-HDR(nighttime)			60.99	62.96	29.63	25.81
LVZ-HDR(nature)			79.87	80.02	37.40	37.40
LVZHDR(indoors)			88.85	88.87	49.29	49.58
LVZHDR(riverside-sunset)			72.2	71.7	28.94	28.81

Table 2: Comparison among methods on TMQI-2 naturalness metric across datasets.

6 LIMITATION AND FUTURE WORK

7 CONCLUSTION

ACKNOWLEDGMENTS

To Robert, for the bagels and explaining CMYK and color spaces.

REFERENCES

[1] Mahmoud Afifi, Brian Price, Scott Cohen, and Michael S Brown. 2019. When color constancy goes wrong: Correcting improperly white-balanced images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1535–1544.

[2] Vivek Agarwal, Besma R Abidi, Andreas Koschan, and Mongi A Abidi. 2006. An overview of color constancy algorithms. *Journal of Pattern Recognition Research* 1, 1 (2006), 42–54.

[3] Nikola Banic and Sven Loncaric. 2018. Flash and Storm: Fast and Highly Practical Tone Mapping based on Naka-Rushton Equation.. In *VISIGRAPP (4: VISAPP)*. 47–53.

[4] Jonathan T Barron. 2015. Convolutional color constancy. In *Proceedings of the IEEE International Conference on Computer Vision*. 379–387.

[5] Jonathan T Barron and Yun-Ta Tsai. 2017. Fast fourier color constancy. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 886–894.

[6] David H Brainard and Brian A Wandell. 1986. Analysis of the retinex theory of color vision. *JOSA A* 3, 10 (1986), 1651–1661.

[7] Mark D Fairchild and Lisa Reniff. 1995. Time course of chromatic adaptation for color-appearance judgments. *JOSA A* 12, 5 (1995), 824–833.

[8] Arjan Gijsenij, Rui Lu, and Theo Gevers. 2011. Color constancy for multiple light sources. *IEEE Transactions on image processing* 21, 2 (2011), 697–707.

[9] Param Hanji, Rafal Mantiuk, Gabriel Eilertsen, Saghi Hajisharif, and Jonas Unger. 2022. Comparison of single image HDR reconstruction methods—the caveats of quality assessment. In *ACM SIGGRAPH 2022 conference proceedings*. 1–8.

[10] Min H Kim, Tim Weyrich, and Jan Kautz. 2009. Modeling human color perception under extended luminance levels. In *ACM SIGGRAPH 2009 papers*. 1–9.

[11] Hui Li, Xixi Jia, and Lei Zhang. 2018. Clustering based content and color adaptive tone mapping. *Computer Vision and Image Understanding* 168 (2018), 37–49.

[12] Xi Lv and Ming Ronnier Luo. 2021. Newcolour Appearance Scales Under High Dynamic Range Conditions. In *Advances in Graphic Communication, Printing and Packaging Technology and Materials: Proceedings of 2020 11th China Academic Conference on Printing and Packaging*. Springer, 60–65.

[13] Kede Ma, Hojatollah Yeganeh, Kai Zeng, and Zhou Wang. 2015. High dynamic range image compression by optimizing tone mapped image quality index. *IEEE Transactions on Image Processing* 24, 10 (2015), 3086–3097.

[14] Radoslaw Mantiuk, Rafal Mantiuk, Anna Tomaszewska, and Wolfgang Heidrich. 2009. Color correction for tone mapping. In *Computer graphics forum*, Vol. 28. Wiley Online Library, 193–202.

[15] Karen Panetta, Landry Kezebou, Victor Oludare, Sos Agaian, and Zehua Xia. 2021. Tmo-net: A parameter-free tone mapping operator using generative adversarial network, and performance benchmarking on large scale hdr dataset. *IEEE Access* 9 (2021), 39500–39517.

[16] Erik Reinhard and Kate Devlin. 2005. Dynamic range reduction inspired by photoreceptor physiology. *IEEE transactions on visualization and computer graphics* 11, 1 (2005), 13–24.

[17] Muhammad Safdar, Jon Yngve Hardeberg, and Ming Ronnier Luo. 2021. ZCAM, a colour appearance model based on a high dynamic range uniform colour space. *Optics Express* 29, 4 (2021), 6036–6052.

[18] Jing Wang, Jana Zujovic, June Choi, Basabdutta Chakraborty, Rene van Egmond, Huib de Ridder, and Thrasyvoulos N Pappas. 2020. Influence of Texture Structure on the Perception of Color Composition. *J. Percept. Imaging* 3, 1 (2020), 10401–1.

[19] Kai-Fu Yang, Shao-Bing Gao, and Yong-Jie Li. 2015. Efficient illuminant estimation for color constancy using grey pixels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2254–2263.

[20] Hojatollah Yeganeh and Zhou Wang. 2012. Objective quality assessment of tone-mapped images. *IEEE Transactions on Image processing* 22, 2 (2012), 657–667.

[21] Zichen Zhao, Hai-Miao Hu, Hongda Zhang, Fei Chen, and Qiang Guo. 2022. Improving color constancy using chromaticity-line prior. *IEEE Transactions on Multimedia* (2022).

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009

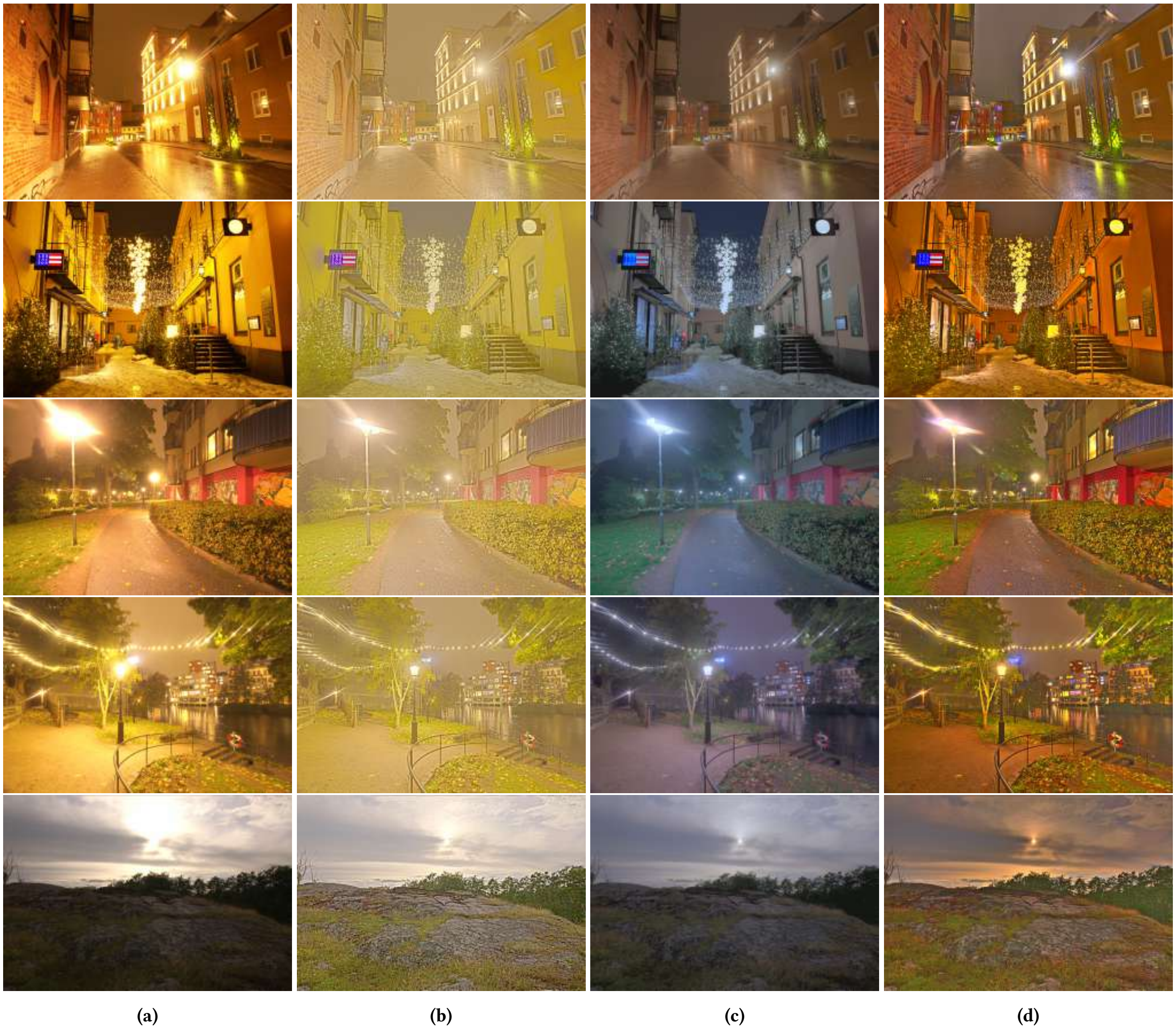


Figure 3: We show the image results comparison for different color adaptive tone mapping methods.We selected five scenes, including simple city, complex city, simple nature, complex nature of the nighttime scenes and a daytime scene.(a): Single ldr image of SIHDR dataset,(b): The clustering based content and color adaptive tone mapping[11] results,(c): The xlrCAM tone mapping results of [10](d): our results. The results show that the results of our methods are more realistic for the nighttime image and can provide a clear and reasonable results for the daytime image although it is a different style from the captured image.