Luminance GS improved

Cheng-Chi Chien[1], I-Ting Liu[1], and Pei-Ju Wu[1]

National Taiwan University, Taipei 106319, Taiwan

Abstract. 3D Gaussian Splatting (3DGS) enables real-time novel view synthesis with high visual quality but remains sensitive to complex lighting variations. Luminance-GS improves robustness by jointly optimizing Gaussian parameters and pseudo ground truth images while preserving rendering efficiency. In this work, we revisit Luminance-GS and identify limitations in its per-view color matrix mapping, which can lead to color inconsistencies. We propose an HSV-based luminance formulation that operates solely on the value channel and introduce a saturation compensation strategy to correct the resulting color distortions. We further analyze the use of Vision Transformers in the view-adaptive curve module. Additionally, our method allows user-controlled adjustment of luminance and color temperature, enhancing flexibility under diverse lighting conditions. The proposed improvements improve color fidelity and practical usability while maintaining the efficiency of the original 3DGS framework.

1 Introduction

In recent years, Gaussian Splatting (3DGS)[7] has been introduced for reconstructing realistic 3D scenes. The goal of 3DGS is to achieve high-quality novel view synthesis from multi-view images in real-time. It represents a scene using thousands to millions of 3D Gaussian primitives. During rasterization, the Gaussians are projected from 3D space to the 2D image plane and sorted by depth. For each pixel, the projected Gaussians that overlap the pixel are composited in a front-to-back order using alpha blending to produce the final image.

Recent studies have shown improvement in tackling various lighting conditions in real world scenarios. To leverage the efficiency of 3DGS's explicit rendering while also ensuring robust reconstruction quality under diverse and challenging lighting conditions, Luminance-GS[1] proposed a method effectively handling varying lighting challenges without imposing additional lighting assumptions.

Luminance-GS argues that applying post-processing to rendered images can introduce view inconsistency across different viewpoints. To address this issue, it proposes a joint optimization strategy in which the loss function simultaneously influences both the Gaussian Splatting model and the generation of pseudo ground truth images. This design allows the pseudo ground truth images to be continuously optimized during training, while ensuring view consistency since the outputs are generated by the jointly trained Gaussian Splatting model. An additional advantage of this approach is that Gaussian Splatting only requires

learning two extra color modulation parameters. These parameters are applied during rendering to adjust color appearance, enabling correct color reproduction for novel views. As a result, the inference speed remains unchanged compared to the original 3DGS model.

However, we have identified several areas where the methodology could be further refined. Specifically, we have reservations regarding the inaccuracies stemming from color discrepancies in per-view color matrix mapping and solve the problem with HSV method, which only apply luminance transform on V channel of HSV. We also discuss the atypical choice of utilizing Vit[2] to extract global image features within the view-adaptive curve module. Upon implementing the proposed method, we encountered a challenge regarding the inaccuracies in saturation. To address this, we incorporated the saturation compensation technique.

Building upon our performance enhancements, we further extend the system's capabilities by granting users precise control over specific luminance and color temperature settings. This provides a more versatile range of options for scene reconstruction, significantly improving its adaptability to diverse real-world lighting conditions and practical applications.

The contributions of our work can be summarized as follows:

- We discuss some points for improvements in Luminance-GS[1], including replacing per-view color matrix mapping with HSV method, using Vit[2] in view-adaptive curve module, trying to compensate saturation loss caused by HSV method.
- We enable user to target luminance and color temperature.

2 Related works

Luminance GS[1] In contrast to prior approaches that modify the explicit 3DGS representation or rely on additional assumptions regarding scene illumination, Luminance-GS adopts a lightweight alternative based solely on image-space curve adjustment. The method maps images captured under diverse lighting conditions into a consistent, well-lit domain while maintaining multi-view coherence. To achieve this goal, Luminance-GS introduces two key components: per-view color matrix mapping and view-adaptive curve adjustment. Specifically, for each viewpoint, the input image is first projected into an appropriate representation via a view-specific color transformation matrix, and is then refined through a curve adjustment module that is adaptively conditioned on the lighting characteristics of the current view. The system is further stabilized through several unsupervised losses, including a 3DGS regularization term $Loss_{reg}$, a structural similarity loss $Loss_{spa}$, a curve-shape constraint $Loss_{curve}$, and a smoothness prior $Loss_{tv}$, collectively ensuring consistent tone mapping and robust training.

3 Improve method

3.1 Replacing learnable M_k with HSV method

HSV method The original enhancing function in [1] is as below:

$$C_k^{out} = L_k \left(C_k^{in} \cdot M_k \right) \cdot M_k^{-1}$$

The input original image and the pseudo-enhanced image are denoted as C_k^{out} and C_k^{in}, respectively. Initially, the RGB channels of C_k^{in} are projected into a transformed space via a learnable matrix M_k for per-view color matrix mapping. Within this projected domain, the luminance component undergoes a non-linear transformation using the luminance curve L_k generated from view-adaptive curve adjustment module. Subsequently, the channels are mapped back to the RGB space by applying the inverse matrix M_k^{-1}, yielding the enhanced output image C_k^{out}.

However, we argue that the inaccuracies in color transformation in M_k in the beginning stage cause the color discrepancies refered in [1]. Consequently, instead of projecting the RGB channels into an appropriate space space, we propose a direct and simpler approach: adjusting the Value ($\mathbf{V}$) channel of the HSV color space directly with the luminance curve $\mathbf{L}_k$. This strategy effectively resolves the color discrepancies, which is illustrated in Fig 1. This simplified process leads to the following formulation:

$$C_k^{out} = Transform_{HSV \to RGB} \left(\left[C_k^{in}(H) \quad C_k^{in}(S) \quad L_k \left(C_k^{in}(V) \right) \right] \right)$$

Fig. 1. Color discrepancies and our improvement. From left to right: the rendered image produced by our HSV-based method, the result of the original method in [1], and the ground truth. The original design in [1] exhibits color discrepancies in certain regions and introduces pixelation artifacts. In contrast, our HSV method effectively alleviates these issues.

3.2 Combination with ViT

In the view-adaptive curve adjustment module of [1], image features are first extracted using two convolutional layers to capture local visual patterns. These local features are subsequently used as the Key and Value inputs to an attention block, while the Query is derived from the camera-to-world transformation matrix.

Motivated by the hypothesis that incorporating global contextual information may further get overall luminance features, we explore replacing the convolutional feature extractor with a Vision Transformer (ViT) [2]. This design enables the model to encode global image context and allows us to empirically evaluate whether such global representations offer advantages over the original purely local feature extraction strategy.

3.3 Target Luminance for 3DGS

On top of the original capability of the model, we aim to enable users to set their ideal luminance for the reconstructed scene. The model was originally designed to automatically learn the most suitable luminance through a luminance curve for each image and the scene to achieve optimal results. It utilized a luminance curve, which histogram equalization plays a significant role in, to process input images into luminance-adjusted pseudo-enhanced images. These pseudo-enhanced images are used to train the additional parameters a,b for each 3DGS, adjusting the color of the 3DGS to become $c = a * c + b$. These enhanced 3DGS are then utilized to render novel views during inference.

In order to achieve our objective, we examined various approaches with promising results. We will introduce our methods in the following sections.

Histogram Equalization with Bounded Output Range (l, r) In standard histogram equalization, the transformation function is the cumulative distribution function CDF.

In this case, where we want to enable the reconstructed scene to have a certain luminance, we can achieve this by limiting the output intensity values of the histogram to fall within a restricted range (l, r), where $0 \leq l < r \leq 1$. Such l, r are set by users, defining the lower and upper bounds of the distribution.

We therefore define a modified transformation function:

$$Y = T(X) = l + (r - l)F_X(X).$$

Where $F_X(X)$ is the CDF. See appendix A.1 for our formal proof of this transformation formula. With this approach, we obtain promising results with various scenes and range settings.

However, since we are basically forcing each pixel of the image to have luminance greater than the set lower bound, pixels in the original image that are nearly black are no longer close to black. To address this, we propose another method that reduces the occurrence of this phenomenon.

Histogram Transform Into a Normal Distribution Instead of transforming the distribution into a uniform distribution with a hard-capped lower bound, we decided to transform the distribution into a normal distribution.

By setting the distribution's mean to a user-defined target with a standard deviation of 0.15, we can achieve our goal while allowing some pixels to remain at relatively low luminance. In our implementation, the user-defined target are calculated as $(l + r)/2$.

We therefore define a modified transformation function:

$$Y = T(X) = \mu + \sigma\Phi^{-1}(F_X(X)),$$

where Φ is the CDF of the standard normal distribution and Φ^{-1} is its inverse (the probit function). Such Y follows a normal distribution $\mathcal{N}(\mu, \sigma^2)$. See appendix A.1 for our formal proof of this transformation formula.

With this approach, we successfully reduce the number of artifacts visible in the scene. It also makes the average luminance of the scene closer to the target in our experiments. However, we still consider that there is some room for improvement in making the average luminance of the scene closer to the target value. Hence, we will introduce yet another approach.

Adjust luminance for pseudo-enhanced images To further improve, we retain all the benefits of the luminance curve after adopting our normal distribution approach mentioned above, and additionally adjust the pseudo-enhanced images to match the target luminance. We utilized linear scaling with ACES tone mapping to achieve that. Our approach avoids brutally clipping values larger than 1 with the help of ACES, a non-linear transform used to convert values greater than 1 back into the [0,1] range while preserving contrast relative to the original values after transformation. This approach gives us the best results across multiple target luminance settings and scenes, achieving our goal of training scenes with specific luminance while maintaining the quality and details of the reconstructed scene.

Fig. 2. Rendering results of garden scene in Mip-NeRF 360 dataset with target luminance set to 0.2, 0.4, 0.6.

3.4 Saturation compensation

To address the saturation problem in HSV mentioned that will discuss in experiment, we investigate two approaches. The first approach initializes saturation using a predefined value and adjusts it using a learnable gamma correction. The second approach adopts CoTF [5] to generate the initial saturation, followed by the same gamma-based adjustment.

Specifically, the saturation value is corrected using a gamma-based function defined as

$$S_{\text{gain}} = a \cdot V_{\text{in}}^{10}$$
$$S_{\text{insert}} = \begin{cases} b, & \text{if } V_{\text{in}} > 0.75 \\ 0, & \text{otherwise} \end{cases}$$
$$S_{\text{out}} = S_{\text{gain}} \cdot S_{\text{in}} + S_{\text{insert}}$$

where S_{in} and S_{out} denote the input and corrected saturation values, respectively, V_{in} denotes the average brightness of the original image, and a and b are learnable parameters optimized during training.

3.5 Target Correlated Color Temperature

Gray World Algorithm Originally, our goal is to address the white balance problem, especially in scenarios where the input image contains multiple correlated color temperatures (CCTs). Such conditions prevent the 3DGS model from converging to a stable color distribution, resulting in over-smoothed and averaged color appearances in the rendered results. To mitigate this issue, we adopt a white balance strategy based on the Gray World assumption.

The Gray World algorithm assumes that a properly white-balanced image has equal average intensities across the RGB channels. Accordingly, white balance is achieved by applying channel-wise gains to the input image, which can be formulated as

$$\begin{bmatrix} R_{\text{out}} \\ G_{\text{out}} \\ B_{\text{out}} \end{bmatrix} = \begin{bmatrix} G_R & 0 & 0 \\ 0 & G_G & 0 \\ 0 & 0 & G_B \end{bmatrix} \begin{bmatrix} R_{\text{in}} \\ G_{\text{in}} \\ B_{\text{in}} \end{bmatrix}$$

where $R_{\text{in}}, G_{\text{in}}, B_{\text{in}}$ and $R_{\text{out}}, G_{\text{out}}, B_{\text{out}}$ denote the input and white-balanced RGB channels, respectively. The gains G_R, G_G, and G_B are learnable parameters optimized during training.

To enforce the Gray World assumption, we encourage the mean values of the output RGB channels to be equal, i.e.,

$$R_{\text{out}} = G_{\text{out}} = B_{\text{out}}$$

which is achieved by minimizing the ℓ_1 loss between each pair of channel means.

Target Correlated Color Temperature While the previous method achieves satisfactory results under white balance conditions, it may fail when the desired output corresponds to a different illumination, such as during sunset lighting. To address this, we extend the original method by learning the ratios between each pair of color channels conditioned on a target reference image. All other components of the method remain unchanged.

4 Experiments results

We do experiments on a single RTX 4090. Similar to [1], we evaluate our model under three challenging lighting conditions: (a) low-light, (b) overexposure using the LOM dataset [3], and (c) varying exposure using the Mip-NeRF 360 dataset [4]. In addition, we adopt the same hyper-parameter settings as [1] to ensure a fair comparison.

4.1 Improved Method

We first adopt the original design in [1] as our baseline and compare it against two variants: (1) HSV method and (2) Vit method. Results on the low-light, overexposure, and varying-exposure datasets are reported in Table 1, Table 2, and Table 3, respectively.

HSV method From the first and second columns of Tables 1, 2, and 3, we observe that the HSV-based method successfully deal the color discrepancies problem, which is shown in Fig 1. The method consistently improves performance under low-light and varying-exposure conditions, but exhibits noticeable degradation in overexposed scenarios. To further investigate this performance drop, we analyze the characteristics of overexposed images and find that overexposure leads to substantial color information loss. As a result, adjusting only the V channel introduces inaccuracies in saturation, causing the enhanced images to appear desaturated and visually gray, which is shown in Fig 4. These observations suggest that while low-light and variable lighting conditions can be effectively addressed by adjusting only the V channel in the HSV channels, overexposed scenarios require more comprehensive color adjustments across the RGB channels to achieve robust performance gains.

Vit method From the first and third columns of Tables 1, 2, and 3, we observe that incorporating ViT results in fluctuating and unstable performance across different lighting conditions. This unexpected behavior can be partially attributed to the training schedule adopted from [1], which consists of only 10,000 optimization steps and may be insufficient to adequately fine-tune the ViT-based architecture. Consequently, these results indicate that directly replacing CNN components with ViT for illumination modeling can be suboptimal without extended training or more carefully designed optimization strategies tailored to transformer-based models.

	baseline	hsv	vit
bike	18.277/0.7491/0.412	**20.141**/**0.7755**/**0.376**	15.600/0.7599/0.390
buu	18.095/0.8775/0.193	**20.957**/**0.9170**/**0.129**	19.674/0.9124/0.173
chair	19.827/0.8357/0.367	**22.462**/**0.8801**/**0.268**	17.602/0.8493/0.334
shrub	15.405/0.6665/0.242	**18.905**/**0.7352**/**0.177**	11.432/0.6098/0.267
sofa	20.122/0.8707/0.259	**23.412**/**0.9051**/**0.249**	21.866/0.9054/0.250
mean	18.345/0.7999/0.295	**21.175**/**0.8426**/**0.240**	17.235/0.8074/0.283

Table 1. Results of LOM dataset[3] low-light subset (PSNR↑,SSIM↑,LPIPS↓). Best results are bolded.

	baseline	hsv	vit
bike	**24.050**/**0.8511**/0.216	19.659/0.7930/0.281	20.025/0.8452/**0.214**
buu	**19.671**/0.8119/0.312	18.495/0.8389/0.371	15.811/**0.8705**/**0.276**
chair	**22.636**/0.8565/**0.208**	18.529/0.7834/0.273	20.286/**0.8631**/0.247
shrub	16.045/0.7809/0.173	**17.518**/**0.7929**/0.247	16.165/0.7863/**0.167**
sofa	21.161/0.8778/**0.204**	17.410/0.8140/0.306	**21.638**/**0.8906**/0.230
mean	**20.713**/0.8356/**0.223**	18.322/0.8044/0.296	**18.785**/0.8511/0.227

Table 2. Results of LOM dataset[3] overexposure subset (PSNR↑,SSIM↑,LPIPS↓). Best results are bolded.

4.2 Target Luminance for 3DGS

To evaluate our results, we calculate the average luminance of the images rendered during the evaluation process to serve as the metric for the luminance of the scene. We performed extensive experiments on various scenes from two datasets with three target luminance settings.

From Table 4, we can observe that our proposed method effectively adjusts the overall luminance of the scene to be relatively close to the target value. In addition, higher luminance settings are more difficult to achieve. We suggest that this comes from the fact that the natural luminance of a scene typically falls in the range of 0.3–0.4, indicating that forcing the model to learn to reconstruct a scene with a luminance far from the optimal value is more difficult. Besides, higher luminance tends to introduce more artifacts compared to lower luminance. As a result, it is more difficult to achieve high scene luminance without damaging image quality. More results tested on LOM dataset[3] can be found in appendix.

Furthermore, Fig. 3 shows the difference between our initial method and the normal distribution method regarding artifacts created in high-luminance situations. As shown in the figure, our normal distribution approach significantly reduces artifact appearances in black areas.

To sum up, we explored several approaches to achieve our goal of enabling users to set their ideal luminance for the reconstructed scene. Our final approach delivers the best performance across multiple settings, with fewer artifacts under high-luminance conditions.

	baseline	hsv	vit
bicycle	18.159/0.6415/0.342	**18.828**/**0.6703**/**0.307**	13.685/0.5961/0.334
bonsai	15.613/0.5598/0.438	**17.490**/**0.6465**/**0.298**	11.289/0.5025/0.377
counter	16.298/0.6313/0.314	**18.815**/**0.7174**/**0.254**	13.273/0.6021/0.263
garden	21.809/0.7833/0.204	**22.927**/**0.8209**/**0.157**	20.055/0.8091/0.172
kitchen	**23.781**/0.8606/0.111	19.305/0.8386/0.106	23.658/**0.8819**/**0.098**
room	16.772/0.6645/0.299	**18.975**/**0.7332**/**0.255**	12.420/0.5939/0.308
stump	15.609/0.5340/0.384	**16.784**/**0.5709**/**0.348**	12.011/0.4596/0.414*
mean	18.292/0.6679/0.299	**19.018**/**0.7140**/**0.246**	15.199/0.6350/0.281

Table 3. Results of Mip-NeRF 360 dataset [4] varying subset (PSNR↑,SSIM↑,LPIPS↓). Best results are bolded. "*" means that the space cost is too high to run the exxperiment.

Scene	Target	Result
bicycle	0.2/0.4/0.6	0.23/0.41/0.59
bonsai	0.2/0.4/0.6	0.21/0.39/0.61
counter	0.2/0.4/0.6	0.22/0.39/0.57
garden	0.2/0.4/0.6	0.17/0.36/0.55
kitchen	0.2/0.4/0.6	0.19/0.37/0.56
room	0.2/0.4/0.6	0.18/0.36/0.53
stump	0.2/0.4/0.6	0.19/0.37/0.54
mean (MipNeRF-360)	**0.2**/**0.4**/**0.6**	**0.199**/**0.379**/**0.564**

Table 4. Results of average luminance for various scenes from Mip-NeRF 360 dataset[4] varying subset.

4.3 Saturation compensation

As mentioned earlier, the HSV-based method can cause saturation loss under overexposure conditions. To address this, we investigate two approaches: (1) initializing the saturation with a predefined value while learning a gamma correction function, and (2) using the output of CoTF [5] to initialize the saturation, followed by learning a gamma correction function.

Quantitative results are summarized in Table 5, while representative qualitative examples are shown in Fig. 4. Both methods improve the metrics (PSNR, SSIM, LPIPS) compared to the baseline, although the overall performance still slightly lags behind the original 3DGS model.

We hypothesize that this limitation arises because the 3DGS model is originally trained with pseudo ground truth images in which saturation has not been fully corrected. Moreover, the incorrect 3DGS output also influences our saturation learning, since the loss affects both the model's predictions and the gamma correction function. As a result, the training tends to converge toward an averaged saturation condition. Because the pseudo ground truth cannot fully correct the saturation errors in the 3DGS output, the supervision remains suboptimal, leading to less accurate reconstruction.

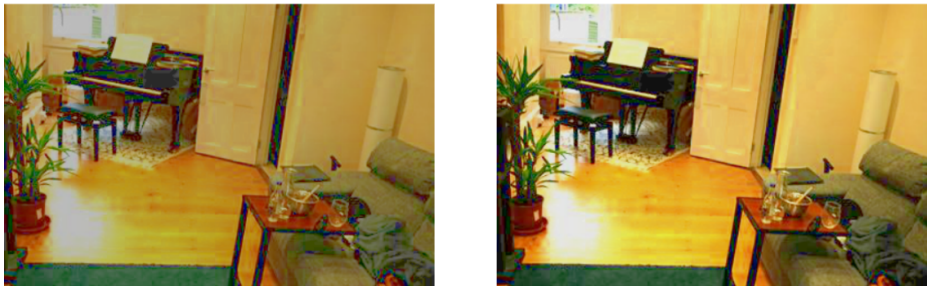

Fig. 3. Comparison between our initial method and the normal distribution method regarding artifact creation in high-luminance situations.

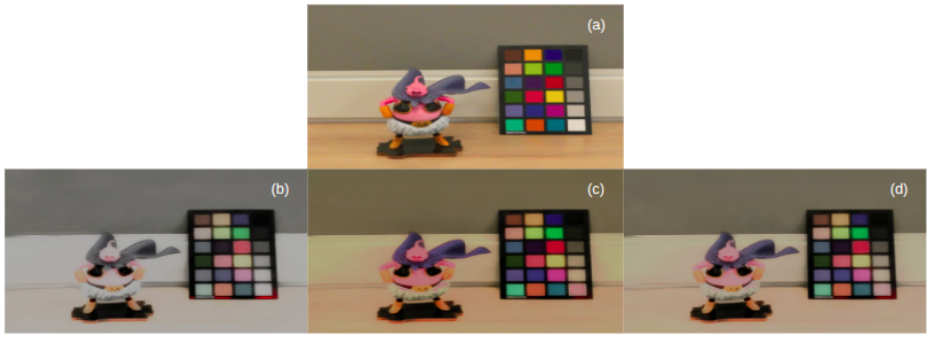

Fig. 4. Comparison between the ground truth image and our baseline method with two saturation compensation strategies. (a) Ground truth image. (b) Output of the HSV-based method. (c) Saturation compensation using a predefined initial value and learning gamma correction. (d) Saturation compensation using CoTF for initialization and learning gamma correction. Compared to the baseline, both saturation compensation methods partially recover the color appearance; however, noticeable discrepancies with the ground truth remain.

4.4 Target Correlated Color Temperature

We evaluate two methods: the standard Gray World algorithm and its extension for target correlated color temperature (CCT).

For the Gray World method, Fig. 5 shows that the images are effectively white-balanced. By examining the RGB channel mean values displayed below each image, we can observe the effectiveness of this approach.

For the extended method, Fig. 6 demonstrates its performance under varying input CCTs. To quantitatively verify its effectiveness, we compute the CCT using the formula proposed by McCamy [6]. Even when the input images have different CCTs, the output images successfully approximate the reference image's CCT.

Despite the simplicity of this extension, it produces significant improvements in color reproduction. Moreover, as shown in Fig. 7, selecting an appropriate

	baseline	hsv	predefined	CoTF
bike	**24.050**/**0.8511**/**0.216**	19.659/0.7930/0.281	19.769/0.7937/0.254	19.406/0.7822/0.260
buu	19.671/0.8119/0.312	18.495/0.8389/0.371	**21.414**/0.8549/**0.237**	21.346/**0.8552**/0.264
chair	**22.636**/**0.8565**/**0.208**	18.529/0.7834/0.273	18.378/0.7856/0.248	18.239/0.7777/0.210
shrub	16.045/0.7809/**0.173**	17.518/0.7929/0.247	17.547/0.8073/0.225	**17.744**/**0.8078**/0.215
sofa	**21.161**/**0.8778**/**0.204**	17.410/0.8140/0.306	17.074/0.8184/0.273	16.044/0.8026/0.313
mean	**20.713**/**0.8356**/**0.223**	18.322/0.8044/0.296	18.836/0.8120/0.247	18.556/0.8051/0.252

Table 5. Results of LOM dataset[3] overexposure subset (PSNR↑,SSIM↑,LPIPS↓). Best results are bolded.

Fig. 5. Result of the Gray World method. The mean RGB values of the final image are $R = 0.3665$, $G = 0.3720$, and $B = 0.3774$, respectively. The close proximity of these values indicates the effectiveness of the Gray World assumption in guiding color balance.

reference image as the conditioning input can further enhance performance, as this approach effectively guides the color distribution of the output.

5 Conclusion

In this project, we try to improve Luminance GS[1] with several aspects. HSV method solves the color discrepancies but suffers from color information loss in overexposure situation. Vit doesn't result in a desirable performance improvement due to the lack of training steps. Our saturation compensation mitigates the loss in the HSV method. We also succeed in targeting luminance and color temperature, offering control over different options. We think that resolving the color information loss in HSV method is worth exploring.

5.1 Author Contributions

The division of work is as follows:

- Cheng-Chi Chien: HSV method, ViT method, paper writing
- I-Ting Liu: Target Luminance, paper writing
- Pei-Ju Wu: Saturation compensation, Target Correlated Color Temperature, paper writing

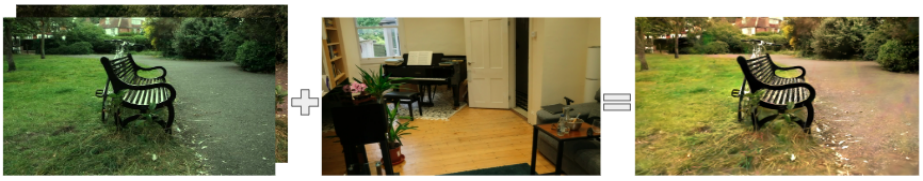

Fig. 6. Result of the extended Gray World method. The reference image has a CCT of 2823 K, while the output image has a CCT of 3016.8 K. Even when the input image exhibits a different CCT, the output CCT remains close to that of the reference image, demonstrating the effectiveness of the proposed extension.

Fig. 7. Result of the extended Gray World method using a high-quality reference image. (a) Ground truth image. (b) Output using the original reference, achieving PSNR = 18.828, SSIM = 0.6703, and LPIPS = 0.307. (c) Output using a higher-quality reference image, achieving PSNR = 20.411, SSIM = 0.6755, and LPIPS = 0.315. Although the selected reference image is not the ground truth and is sampled from the training set, the results show improved PSNR and SSIM compared to the original output.

5.2 LLM Use

- Cheng-Chi Chien: Assisted in writing the paper with ChatGPT.
- I-Ting Liu: Assisted in writing the paper and part of code with ChatGPT.
- Pei-Ju Wu: Most of the code was generated using Gemini 3, with contributions focused on debugging and paper writing with ChatGPT.

References

1. Cui, Ziteng et al. “Luminance-GS: Adapting 3D Gaussian Splatting to Challenging Lighting Conditions with View-Adaptive Curve Adjustment.” 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025): 26472-26482.
2. Dosovitskiy, Alexey et al. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.” ArXiv abs/2010.11929 (2020): n. pag.
3. Cui, Ziteng et al. “Aleth-NeRF: Low-light Condition View Synthesis with Concealing Fields.” ArXiv abs/2303.05807 (2023): n. pag.
4. Barron, Jonathan T. et al. “Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields.” 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021): 5460-5469.

5. Li, Ziwen et al. “Real-Time Exposure Correction via Collaborative Transformations and Adaptive Sampling.” 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024): 2984-2994.
6. McCamy, Calvin S. "Correlated Color Temperature as an Explicit Function of Chromaticity Coordinates." Color Research and Application, vol. 17, no. 2, 1992, pp. 142-44.
7. Kerbl, Bernhard et al. "3D Gaussian Splatting for Real-Time Radiance Field Rendering" ArXiv abs/2308.04079 (2023): n. pag.

A Appendix

A.1 Proofs of transformation formulas for target luminance

Proof of Y is Uniformly Distributed on (a, b)

Proof. We show that the resulting variable Y is uniformly distributed on (l, r).

Starting with the definition:

$$Y = l + (r - l)F_X(X),$$

we compute its CDF.

$$F_Y(y) = P(Y \leq y).$$

Substitute the expression for Y:

$$P\big(l + (r - l)F_X(X) \leq y\big).$$

Subtract a from both sides:

$$P\big((r - l)F_X(X) \leq r - l\big).$$

Divide both sides by $(r - l)$:

$$P\left(F_X(X) \leq \frac{y - l}{r - l}\right).$$

Because F_X is a CDF and therefore monotonically non-decreasing, we apply the inverse CDF:

$$P\left(X \leq F_X^{-1}\left(\frac{y - l}{r - l}\right)\right).$$

By definition of the CDF:

$$F_X\left(F_X^{-1}\left(\frac{y - l}{r - l}\right)\right) = \frac{y - l}{r - l}.$$

Thus the CDF of Y is:

$$F_Y(y) = \frac{y - l}{r - l}, \qquad y \in [l, r].$$

Differentiating gives the PDF:

$$f_Y(y) = \frac{d}{dy}F_Y(y) = \frac{1}{r - l}.$$

This is exactly the PDF of a uniform distribution on (l, r):

$$Y \sim \mathcal{U}(l, r).$$

Therefore, the transformation

$$T(x) = l + (r - l)F_X(x)$$

performs histogram equalization but restricts the output intensity values to lie within the desired interval (l, r).

Proof of Y is Normal Distribution with mean μ and standard variance σ

Proof. Define

$$Z = \Phi^{-1}(F_X(X)).$$

We first show that $Z \sim \mathcal{N}(0,1)$.
Consider the CDF of Z:

$$F_Z(z) = P(Z \leq z).$$

Substitute the definition of Z:

$$P\big(\Phi^{-1}(F_X(X)) \leq z\big).$$

Apply Φ (monotonically increasing) to both sides:

$$P\big(F_X(X) \leq \Phi(z)\big).$$

Apply inverse CDF of X:

$$P\big(X \leq F_X^{-1}(\Phi(z))\big).$$

Then, by definition of the CDF:

$$F_X\big(F_X^{-1}(\Phi(z))\big) = \Phi(z).$$

Thus,

$$F_Z(z) = \Phi(z).$$

Taking derivatives:

$$f_Z(z) = \frac{d}{dz}F_Z(z) = \frac{1}{\sqrt{2\pi}}e^{-z^2/2}.$$

Therefore:

$$Z \sim \mathcal{N}(0,1).$$

Final Variable $Y = \mu + \sigma Z$
Given a standard normal Z, we define

$$Y = \mu + \sigma Z.$$

Linear transformation of a normal distribution results in:

$$Y \sim \mathcal{N}(\mu, \sigma^2).$$

Thus, the full transformation

$$T(x) = \mu + \sigma\Phi^{-1}(F_X(x))$$

maps any input distribution (with CDF F_X) into a normal distribution with mean μ and variance σ^2.

A.2 Additional results

Target Luminance for 3DGS Table 6 shows results of our method tested on LOM dataset[3].

Scene	Target	Result
bike_low	0.2/0.4/0.6	0.22/0.41/0.61
buu_low	0.2/0.4/0.6	0.21/0.39/0.57
chair_low	0.2/0.4/0.6	0.2/0.4/0.6
shrub_low	0.2/0.4/0.6	0.2/0.35/0.51
sofa_low	0.2/0.4/0.6	0.21/0.41/0.6
mean (LOM low exp)	**0.2/0.4/0.6**	**0.208/0.392/0.578**
bike_over	0.2/0.4/0.6	0.2/0.4/0.6
buu_over	0.2/0.4/0.6	0.22/0.41/0.59
chair_over	0.2/0.4/0.6	0.2/0.39/0.58
shrub_over	0.2/0.4/0.6	0.2/0.39/0.58
sofa_over	0.2/0.4/0.6	0.21/0.41/0.6
mean (LOM over exp)	**0.2/0.4/0.6**	**0.206/0.4/0.59**

Table 6. Results of average luminance for various scenes from LOM dataset[3] with both low and over exposure subset.