


## Article

# A 5K Efficient Low-Light Enhancement Model by Estimating Increment between Dark Image and Transmission Map Based on Local Maximum Color Value Prior

Qikang Deng <sup>1</sup> , Dongwon Choo <sup>2</sup>, Hyochul Ji <sup>2</sup> and Dohoon Lee <sup>1,\*</sup>

<sup>1</sup> Department of Information Convergence Engineering, Pusan National University, Geumjeong-gu, Busan 46241, Republic of Korea; dengqikang@pusan.ac.kr

<sup>2</sup> InBic Inc., 20F, 8, Seongnam-daero 331beon-gil, Bundang-gu, Seongnam-si 13637, Republic of Korea; dongwon.choo@inbic.ai (D.C.); hyochul.ji@inbic.ai (H.J.)

\* Correspondence: dohoon@pusan.ac.kr; Tel.: +82-51-510-2491

**Abstract:** Low-light enhancement (LLE) has seen significant advancements over decades, leading to substantial improvements in image quality that even surpass ground truth. However, these advancements have come with a downside as the models grew in size and complexity, losing their lightweight and real-time capabilities crucial for applications like surveillance, autonomous driving, smartphones, and unmanned aerial vehicles (UAVs). To address this challenge, we propose an exceptionally lightweight model with just around 5K parameters, which is capable of delivering high-quality LLE results. Our method focuses on estimating the incremental changes from dark images to transmission maps based on the low maximum color value prior, and we introduce a novel three-channel transmission map to capture more details and information compared to the traditional one-channel transmission map. This innovative design allows for more effective matching of incremental estimation results, enabling distinct transmission adjustments to be applied to the R, G, and B channels of the image. This streamlined approach ensures that our model remains lightweight, making it suitable for deployment on low-performance devices without compromising real-time performance. Our experiments confirm the effectiveness of our model, achieving high-quality LLE comparable to the IAT (local) model. Impressively, our model achieves this level of performance while utilizing only 0.512 GFLOPs and 4.7K parameters, representing just 39.1% of the GFLOPs and 23.5% of the parameters used by the IAT (local) model.

**Keywords:** low light enhancement; lightweight; real time; local maximum color value prior; transmission map estimation; surveillance video; low-performance devices



**Citation:** Deng, Q.; Choo, D.; Ji, H.; Lee, D. A 5K Efficient Low-Light Enhancement Model by Estimating Increment between Dark Image and Transmission Map Based on Local Maximum Color Value Prior. *Electronics* **2024**, *13*, 1814. <https://doi.org/10.3390/electronics13101814>

Academic Editor: Luca Mesin

Received: 28 February 2024

Revised: 30 March 2024

Accepted: 12 April 2024

Published: 8 May 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

From early humans seeking to detect predators in the darkness to modern-day applications like surveillance systems, autonomous vehicles, and smartphone cameras, the ability to see clearly in low-light environments remains a crucial technological challenge.

Despite significant progress in the development of LLE techniques, including classical algorithms [1–3], deep learning methods [4–6], and image generation techniques [7–11], several key challenges persist in current methodologies. These challenges revolve around achieving a balance between enhancement quality and real-time processing, optimizing models for lightweight deployment on resource-constrained devices, and ensuring the reliability of image enhancement results.

Classical algorithms are known for their exceptional reliability due to logical design principles. In contrast, deep learning models excel in efficiency and accuracy but may suffer from design redundancy. Our objective is to leverage the strengths of both approaches by developing a hybrid deep learning model that is efficient, lightweight, and produces high-quality results while ensuring reliability.

To achieve this goal, we focus on deconstructing the classical local maximum color value (LMCV) prior-based LLE algorithm [3]. We extract critical functions such as transmission map estimation, transmission value correction using guided filters, atmospheric light calculation, and adaptive exposure adjustment, which are all suitable for replacement with deep learning models.

We introduce a transmission estimation module (TEM) to estimate the increment between the dark image and the transmission map, thereby replacing the transmission map estimation function. The three-channel transmission map produced by TEM can preserve more details and intelligence while avoiding checkerboard artifacts, thus eliminating the need for the transmission value correction function. Additionally, a correction module is introduced to replace atmospheric light calculation and adaptive exposure adjustment functions. These functions were redefined as color correction and denoise layers within the correction module.

Finally, we created a novel and remarkably compact LLE model, and the experimental results show that our model achieves high-quality enhancements while utilizing only around 5K parameters and 0.512 GFLOPs. These numbers represent a significant reduction compared to other existing models, showcasing the efficiency and effectiveness of our proposed method.

Our main contributions can be summarized as follows:

1. A novel method for estimating a three-channel transmission map by estimating the incremental values from the dark image to the transmission map.
2. Introducing a lightweight U-Net structure transmission estimation module.
3. Creation of a lightweight correction module for color correction and denoise.
4. Proposing a joint loss function specifically designed for optimizing the training process of the low-light enhancement model.

## 2. Related Work

In the development of LLE, numerous physical models and algorithms have been proposed. The most fundamental physical theories include the Atmospheric Scattering Model (ASM) [12] and the Retinex model [13]. As for the latest deep learning algorithms, transformers [14] and AI-generated content (AIGC) [15] have been widely utilized.

### 2.1. Atmospheric Scattering-Based Model

The ASM is a fundamental physical framework for the LLE and defogging models. It elucidates the correlations between scene radiance  $J(x)$ , transmission map  $t(x)$  and atmospheric light  $A$  in the source images  $I(x)$  as depicted in Equation (1);  $x$  donates the pixel coordinates.

$$I(x) = J(x)t(x) + (1 - t(x))A \quad (1)$$

Several studies have integrated the ASM approach to developing the LLE model, including those by Wang et al. [1], Liu et al. [16], Jeon et al. [17] and Makwana et al. [18]. These endeavors aimed to estimate scene radiance  $J(x)$  by determining the values of transmission  $t(x)$  and atmospheric light  $A$ , resulting in satisfactory performance.

### 2.2. Retinex-Based Model

The Retinex-based model was proposed by Land et al. [13] in 1963. It represents another classical theory for the LLE model. The Retinex model decomposes the source image  $I(x)$  into reflectance  $R(x)$  and illumination  $L(x)$ , as shown in Equation (2).

$$I(x) = R(x) \circ L(x) \quad (2)$$

where  $\circ$  denotes element-wise multiplication.

Consequently, to derive the reflectance  $R(x)$  result, we can estimate the illumination  $L(x)$ . The Retinex can be viewed as a simplified ASM that integrates the estimation for atmospheric light  $A$  and transmitted light  $t(x)$  into an illumination  $L(x)$ . The latest Retinex-



based models include Zero-DCE++ [19], CSDNet [20], DeepUPE [21], MBLLEN [22], RetinexNet [23], RUAS [24], TBEFN [25], SGM [26], DRBN [27], and KinD++ [28].

### 2.3. Transformer-Based Model

The Transformer model has an exceptional capability to extract both global and local information, rendering it suitable for various tasks as a plug-and-play module. Transformer-based LLEs, such as Retinexformer [29], Uformer [30], Restormer [31], LLFormer [32], and LYT-Net [33] have demonstrated significant accuracy advantages. However, Transformers demand substantial computational and memory resources, which is unsuitable for low-performance devices. Further efforts are required to address the computational and memory requirements of Transformer-based models and make progress in lightweight models.

### 2.4. AI-Generated Content-Based Model

In 2022, AI-generated content [15] emerged as a prominent topic worldwide, leading to significant advancements in image generation. Numerous image generation models for LLE have been proposed, including LPDM [7], PyDiff [8], DiffLIE [9], L<sup>2</sup>DM [10], and Diff-Retinex [11]. These models achieved LLE by generating normal light images from dark images. They rely on a probabilistic distribution-based random mapping to mitigate limitations such as noise, color shifts, and information loss from the source image, resulting in a perfect enhancement of low-light images to ground truth (GT) images. However, it is essential to note that the generated images may not always be reliable, as generative models might introduce content not present in the source image. It posed challenges for tasks requiring content authenticity.

### 2.5. Challenges

LLE methods leverage physical models like the Atmospheric Scattering Model (ASM) and Retinex model, or deep learning techniques such as Transformer-based models and AI-generated content (AIGC) methods. ASM and Retinex models form the basis of many algorithms but suffer from complex calculations regarding transmission maps and atmospheric light. Transformer-based models offer high accuracy but demand significant computational resources, limiting their deployment on low-performance devices. AIGC methods show promise in image enhancement but require substantial computational overhead and may lack content authenticity, posing challenges for practical applications. Future research needs to address these challenges to develop efficient and reliable low-light image enhancement solutions suitable for various computing environments.

## 3. Methodology

### 3.1. Motivation

The Local Maximum Color Value (LMCV) algorithm was proposed by Dong et al. [3] in 2016. The LMCV algorithm stands as a remarkable classical LLE method due to its introduction of the LMCV prior concept based on well-exposed images. By combining the ASM model with the LMCV prior, the algorithm effectively enhances low-light images while also offering adaptive exposure adjustment for under-exposed regions. This innovative approach not only improves image quality but also demonstrates the adaptability of LLE methods to varying lighting conditions, making it a significant contribution to the field of image processing.

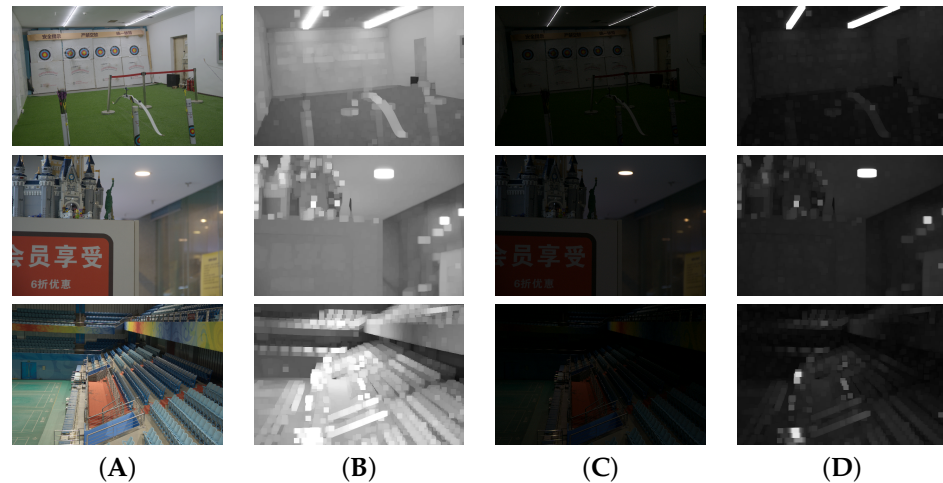
Firstly, they introduce a new term LMCV map  $J^{LMCV}$  that represents a map where each pixel's value is the local maximum color value within its neighborhood. Specifically, in an image processed using the LMCV algorithm, each pixel's value in the LMCV map is derived from the highest color value found in its local area, which is represented by Equation (3).

$$J^{LMCV}(x) = \max_{c \in \{r,g,b\}} \max_{y \in \Omega(x)} J^c(y) \quad (3)$$

where  $x$  represents the pixel coordinates,  $J$  represents the scene radiance,  $c$  is the RGB channel and  $\Omega(x)$  represents the points within a certain region centered at  $x$ . Specifically, in the LMCV algorithm, the region size is typically set to  $15 \times 15$  pixels.

Moreover, through their statistical study, they observed that in an image with sufficient exposure, each pixel's value in its LMCV map tends to approximate the highest value within the pixel's range (see Figure 1). This observation is formally denoted as the LMCV prior, which is represented by Equation (4).

$$J^{LMCV}(x) \rightarrow 1 \quad (4)$$



**Figure 1.** LMCV map: Each pixel in the LMCV map corresponds to the maximum color intensity value within a local patch around it. Column (A) displays well-exposed images alongside their respective LMCV maps in Column (B). Similarly, Column (C) showcases low-light images, which are accompanied by their corresponding LMCV maps in Column (D). The LMCV map exhibits brighter points with values close to 1, indicating higher color intensity, while darker points tend to have values closer to 0, signifying lower color intensity. Based on the LMCV prior, each pixel value in the LMCV map of a well-exposed image is close to the maximum value of 1.

Therefore, when we apply the LMCV map operation to both sides of the ASM Equation (1), we obtain the modified Equation (5),

$$I^{LMCV}(x) = t(x)J^{LMCV}(x) + (1 - t(x)) \max_{c \in \{r,g,b\}} A^c \quad (5)$$

where  $I^{LMCV}(x)$  represents the LMCV map of the source image,  $J^{LMCV}(x)$  represents the LMCV map of the scene radiance, and  $t(x)$  represents the transmission map.

Based on the LMCV prior, we assume that the scene radiance  $J(x)$  represents the well-exposed image we desire; then, we can set its LMCV map as  $J^{LMCV}(x) = 1$ . Subsequently, Equation (5) can be simplified to Equation (6).

$$I^{LMCV}(x) = t(x) \times 1 + (1 - t(x)) \max_{c \in \{r,g,b\}} A^c \quad (6)$$

Consequently, as shown in Equation (7), the transmission map  $t(x)$  can be considered as composed of  $I^{LMCV}(x)$  and the atmospheric light  $A$ .

$$t(x) = \frac{I^{LMCV}(x) - \max_{c \in \{r,g,b\}} A^c}{1 - \max_{c \in \{r,g,b\}} A^c} \quad (7)$$

Once we obtain the transmission map  $t(x)$ , we can solve for the scene radiance  $J(x)$  based on the ASM model, as shown in Equation (8).

$$J(x) = \frac{I(x) - A}{t(x)} + A \quad (8)$$

In the original LMCV algorithm, a  $15 \times 15$  kernel operation is employed to generate the LMCV map  $I^{LMCV}$ , while complex matrix element lookups are used to determine the atmospheric light  $A$ . Additionally, the algorithm relies on the highly computationally intensive guided filter to mitigate checkerboard artifacts resulting from the  $15 \times 15$  kernel operation (as shown in Figure 2, middle). These operations contribute significantly to the overall complexity of computations within the LMCV algorithm. To streamline the algorithm and improve computational efficiency, it is imperative to simplify these operations or replace them with more efficient deep learning techniques.

In this paper, we introduce a novel method for estimating the transmission map called Dark Image Increment Estimation (DIIE). Furthermore, we propose two deep learning modules: the Transmission Estimation Module (TEM) and the Correction Module (CM). These methods and modules are specifically designed to replace the computationally intensive functions present in the original LMCV algorithm. By leveraging DIIE and these modules, we aim to enhance efficiency while either maintaining or improving accuracy in low-light image enhancement tasks.



**Figure 2.** (Left): Transmission map generated by LMCV [3] algorithm. (Middle): Transmission map on the left processed with a guided filter to eliminate checkerboard artifacts. (Right): Three-channel transmission map estimated by our method. Our estimated 3-channel transmission map captures richer color and detail information without checkerboard artifacts. The red circles show the localized zoomed-in content at the same location of the images.

### 3.2. Dark Image Increment Estimation (DIIE) for Transmission Map

In Section 3.1, we derive the transmission map  $t(x)$  and scene radiance  $J(x)$  from the dark source image  $I(x)$ , its LMCV map  $I^{LMCV}(x)$ , and the atmospheric light  $A$ , as shown in Equations (7) and (8). If we assume that any missing influence of atmospheric light  $A$  can be addressed through deep learning model adaptivity, by ignoring  $A$  in Equations (7) and (8), we can simplify them to Equations (9) and (10).

$$t(x) = I^{LMCV}(x) \quad (9)$$

$$J(x) = \frac{I(x)}{t(x)} \quad (10)$$

where  $t(x)$  represents the simplified estimation of the transmission map and  $J(x)$  refers to the scene radiance derived from our simplified LMCV algorithm.

Combining Equations (9) and (10) yields Equation (11), indicating that the scene radiance  $J(x)$  depends solely on the dark source image  $I(x)$  and its LMCV map  $I^{LMCV}(x)$ .

$$J(x) = \frac{I(x)}{I^{LMCV}(x)} \quad (11)$$

Additionally, considering the definition of the LMCV map as having pixel values representing maximum values in local areas,  $I^{LMCV}(x)$  must be greater than or equal to  $I(x)$  at any pixel point  $x$ , as per Equation (12).

$$I(x) \leq I^{LMCV}(x) \quad (12)$$

Thus, we decompose  $I^{LMCV}(x)$  into adding a positive increment  $f(x)$  to  $I(x)$  in Equation (13). Notably, we redefine both the LMCV map  $I^{LMCV}(x)$  and the transmission map  $t(x)$  as three-channel maps from Equation (13) onwards, aligning better with deep learning models and preserving image details and color information.

$$t(x) = I^{LMCV}(x) = I(x) + f(x) \quad (13)$$

Therefore, solving for scene radiance  $J(x)$  involves finding the correct increment  $f(x)$  as shown in Equation (14), where  $J'(x)$  is an intermediate variable for  $J(x)$ .

$$J'(x) = \frac{I(x)}{I(x) + f(x)} \quad (14)$$

To address biases from removing atmospheric light  $A$ , we need to predict a correction  $g(\cdot)$  to obtain the final scene radiance  $J(x)$ , as shown in Equation (15).

$$J(x) = g(J'(x)) \quad (15)$$

In summary, our proposed DIIE (Dark Image Increment Estimation) method simplifies the LMCV algorithm by focusing on predicting the increment  $f(x)$  and performing atmospheric light correction  $g(\cdot)$ . To achieve this, we introduce two deep learning modules: the Transmission Estimation Module (TEM) and the Correction Module (CM) to estimate them, respectively. Pseudocode Algorithm 1 provides an overview of our proposed model.

---

**Algorithm 1** Overview of our proposed model

---

```

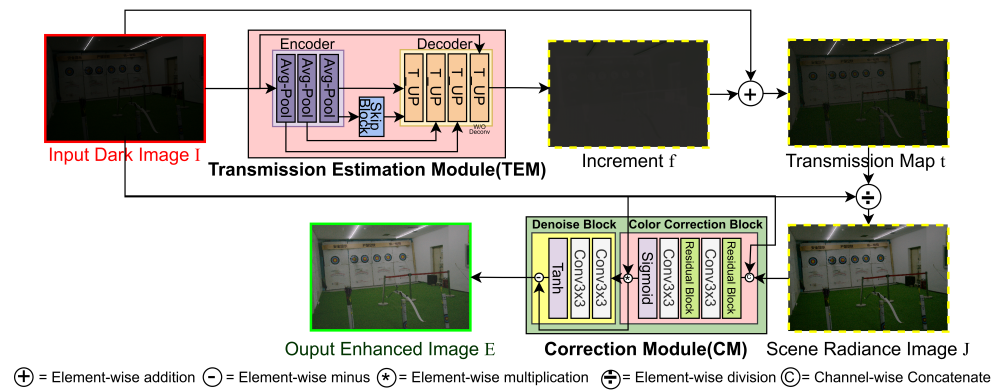
1: Input:  $I$ : Dark image:  $[C \times H \times W]$ 
2: Output:  $E$  Enhanced image:  $[C \times H \times W]$ 
3: procedure MODEL( $I$ )
4:    $f \leftarrow \text{TEM}(I)$  ▷ Estimates the increment  $f$ 
5:    $t \leftarrow I + f$  ▷ Gets 3-channel transmission map  $t$ 
6:    $J \leftarrow I \div t$  ▷ Gets scene radiance image  $J$ 
7:    $E \leftarrow \text{CM}(I, J)$  ▷ Generates the final enhanced image  $E$ 
8:   return  $E$ 
9: end procedure

```

---

### 3.3. Model Architecture: TEM and CM Module

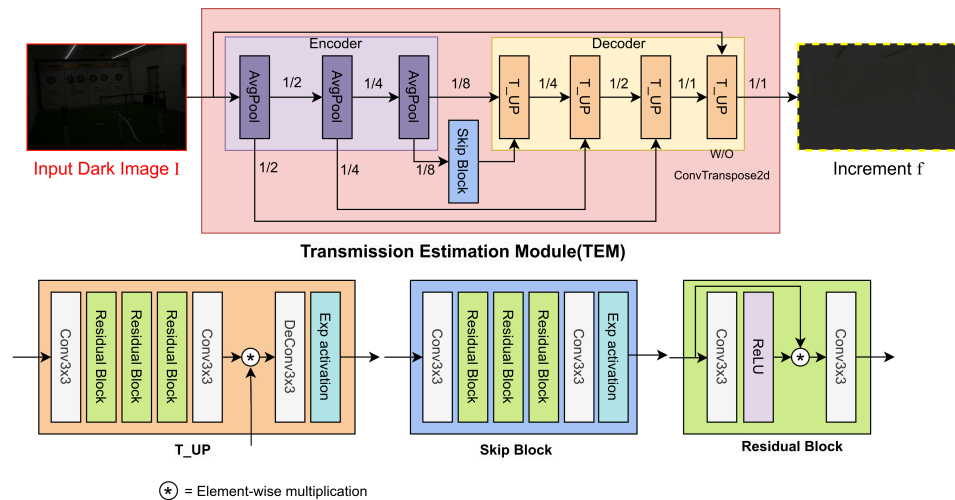
We propose an LLE model based on our proposed DIIE method, comprising approximately 5K learnable parameters, as illustrated in Figure 3. The Transmission Estimation Module (TEM) is structured as a U-Net [34] model, focusing on estimating the increment  $f(x)$  between the dark source image  $I(x)$  and the three-channel transmission map  $t(x)$  shown in Figure 2 (Right). On the other hand, the Correction Module (CM) adopts the skip connection [35] structure, which is utilized for color correction and noise reduction, compensating for the lack of atmospheric light  $A$ . Notably, our model maintains a consistent three-channel matrix output for most layers without increasing the number of channels in the matrices. This design choice significantly contributes to the model's compactness and efficiency in terms of parameters.



**Figure 3.** Total Model Architecture: Our model comprises two primary modules, namely the TEM and the CM. In Equation (12),  $I(x)$  represents the dark image, constrained by  $I(x) \leq t(x)$ , where  $t(x)$  denotes the transmission map. TEM estimates the incremental values  $f(x)$  between the dark image and the three-channel transmission map  $t(x)$ , which is given by  $t(x) = I(x) + f(x)$ . CM is employed for image color adjustment and noise reduction to eliminate the impacts of atmospheric light  $A$ .

### 3.3.1. Transmission Estimation Module

The Transmission Estimation Module (TEM) takes the dark source image  $I$  as input and is responsible for estimating the increment  $f$ . The sum of the increment  $f$  and the dark source image  $I$  yields the transmission map  $t$ . To meet this requirement, we propose a U-Net structure module for our TEM, comprising three parts: Encoder, Skip Block, and Decoder as shown in Figure 4.



**Figure 4.** Transmission Estimation Module: The model follows a UNet-like structure, where the Encoder does not contain trainable parameters but consists of three average pooling layers. Only the **Skip Block** and Decoder parts of the model have trainable parameters. Notably, we maintained the number of channels in the features without increasing them; all features in the model are kept as 3 channels throughout.

**Encoder:** does not contain any trainable parameters but includes three AvgPooling layers. These layers generate pooling images at 1/2, 1/4, and 1/8 scales of the original image. These pooling images will be fed into the Decoder.

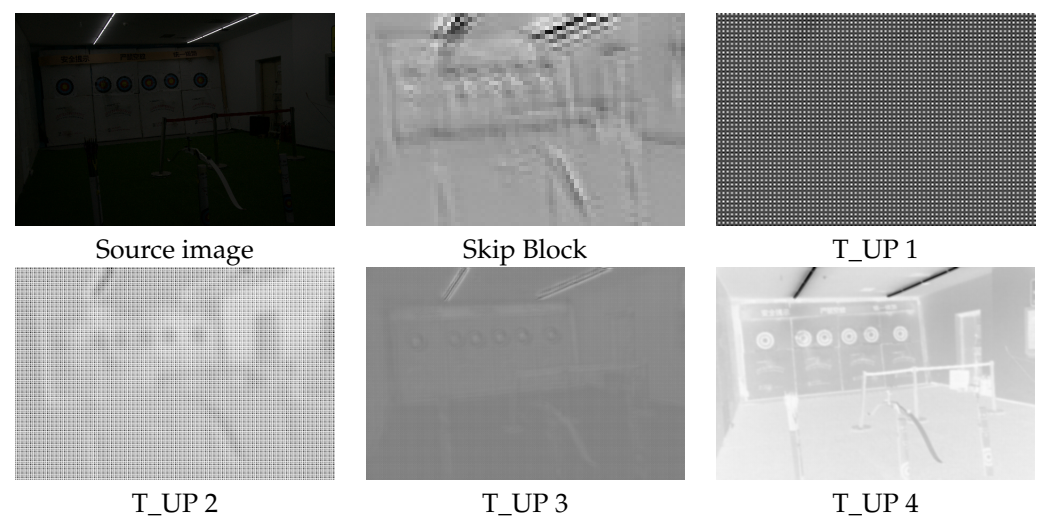
**Skip Block:** has a structure of  $\text{conv}3 \times 3 + \text{ResBlock} + \text{ResBlock} + \text{ResBlock} + \text{conv}3 \times 3$  with inputs and outputs being 3-channel features. It is primarily responsible for extracting global features from the smallest 1/8 scale pooling image, which will be fed into the first layer of the Decoder.



**Decoder:** contains most of our trainable parameters and consists of four T\_UP layers. The structure of the T\_UP layer is similar to the Skip Block, but we have added a Deconvolution layer for upscaling the pooling images from the Encoder, as shown in the T\_UP diagram in Figure 5.

Based on the outputs of the Skip Block and Decoder in Figure 5, we find that the output of the T\_UP 4 layer is the estimated increment  $f$ . At the fluorescent tube lights' position in the image  $x'$ , the increment value  $f(x')$  tends to 0, while in the dark source image  $I(x')$ , the color at position  $x'$  should tend to 1. According to Equation (14), we find that the scene radiance  $J(x')$  at this position is equal to  $1/(1+0) = 1$ , maintaining the original brightness, which aligns with our desire.

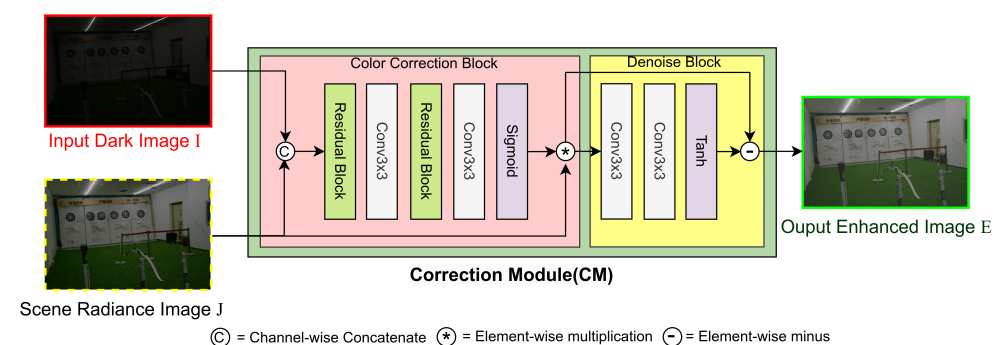
Compared to the original transmission map, our estimated 3-channel transmission map addresses the checkerboard artifacts problem while also offering enhanced detail and color information, as depicted in Figure 2.



**Figure 5.** Output features from Skip Block and Decoder T\_UP layers, respectively. T\_UP 4 is our estimated increment  $f$ .

### 3.3.2. Correction Module

While the TEM can somewhat reduce the influence of atmospheric light due to the adaptability of deep learning models, the scene radiance image  $J$  outputted by TEM may still suffer from color biases and noise introduced by image enhancement. Therefore, we introduce the Correction Module (CM), which includes a color correction block and a denoising block, as shown in Figure 6.



**Figure 6.** Correction module architecture: includes a color correction block and a denoise block for further correction of the enhanced scene radiance image. We incorporate both a color correction block and a denoise block to further enhance the scene radiance image, leveraging information from the input dark image for improved color accuracy.

**Color Correction Block:** To adjust color biases, we concatenate the dark source image  $I$  and the scene radiance image  $J$ , forming a 6-channel feature metric  $I \_ J$ . This combined feature metric  $I \_ J$  is then fed to the Color Correction block. The output of the Color Correction block is a 3-channel matrix  $c$  in the range of 0–1, and the result of  $J \times c$  represents the color-corrected result.

**Denoise Block:** Inspired by the Zero-Shot Noise2Noise model [36], we utilize two convolutional layers to estimate and subtract noise from the image, thereby producing a denoised image or, in other words, the final enhanced image  $E$ .

In summary, our proposed model, including Transmission Estimation Module (TEM) and Correction Module (CM), leverages deep learning techniques to estimate the increment  $f(x)$  for the transmission map based on our DIIE method. Furthermore, it performs color correction and noise reduction for the final enhanced image  $E$ . This streamlined approach offers computational efficiency while maintaining high quality in low-light image enhancement tasks.

### 3.4. Joint Loss

To train the proposed model to achieve the best performance, we combine multiple loss functions, which include both full-reference and no-reference loss functions.

#### (1) Full-Reference Loss Functions

**Mean Absolute (L1) Loss.** L1 Loss is a common loss function for various image-to-image tasks such as image reconstruction, image super-resolution and image denoise. It is an effective loss function as it directly computes the mean distance between the enhanced result and the ground truth. We propose L1 loss  $\mathcal{L}_{\ell_1}$  to compare the final enhanced image and the ground truth data.

$$\mathcal{L}_{\ell_1} = \frac{1}{N} \sum_{x \in p} |E(x) - GT(x)| \quad (16)$$

where  $E$  denotes the final enhanced image,  $GT$  denotes the ground truth data and  $x$  denote the pixel's coordinate in  $p$  space.  $N$  is the total number of pixels.

The L1 loss measures the pixel-level difference between the enhanced image and the ground truth, offering a highly accurate match to the ground truth data. However, its high precision can sometimes lead to overfitting issues. To address this, we introduce the Root Mean Squared Log Error (RMSLE) loss.

**Root Mean Squared Log Error (RMSLE) loss.** RMSLE loss utilizes the logarithm function based on the root mean squared error, which can reduce the impact of large differences between a few values and the ground truth in the overall error calculation. Thus, RMSLE loss  $\mathcal{L}_{RMSLE}$  allows for localized small errors. We use RMSLE loss to measure the difference between the scene radiance  $J(x)$  and the ground truth because scene radiance  $J(x)$  is not the final enhanced image processed by the correction module. Therefore, we tolerate some errors in this comparison to overcome the overfitting.

$$\mathcal{L}_{RMSLE} = \sqrt{\frac{1}{N} \sum_{x \in p} ((\log J(x) + 1) - \log(GT(x) + 1))^2} \quad (17)$$

While both the L1 and RMSLE loss functions operate at the pixel level and aim to minimize differences between pixels, they do not account for whether the enhanced image aligns with human visual perception. To enhance the visual quality of the enhanced image, we introduce the Structural Similarity (SSIM) loss.

**Structural Similarity (SSIM) loss.** SSIM loss compares two images' brightness, contrast, and structure, providing a better metric of human visual perception of image differences.

We propose the SSIM loss  $\mathcal{L}_{SSIM}$  to measure the loss between the enhanced image and the ground truth data.

$$\mathcal{L}_{SSIM} = 1 - SSIM(E, GT) \quad (18)$$

## (2) No-Reference Loss Functions

Additionally, to suppress noise and mitigate color biases, we introduce two no-reference functions.

**Illumination Smoothness Loss.** The Illumination Smoothness Loss  $\mathcal{L}_{tv}$  calculates the noise level of the enhanced image by identifying prominent points on smooth surfaces using gradient operations in both horizontal and vertical directions. Through statistical analysis of these prominent points, the loss function quantifies the noise level present in the enhanced image.

$$\mathcal{L}_{tv} = \frac{1}{N} \sum_{c \in \xi} (|\nabla_h E^c| + |\nabla_v E^c|)^2, \xi = \{R, G, B\} \quad (19)$$

where  $c$  denotes the color channel, while  $\nabla_h$  and  $\nabla_v$  denote the horizontal and vertical gradient operations.

$\mathcal{L}_{tv}$  loss aims to minimize noise and irregularities in the output image, resulting in smoother images with fewer prominent noise points.

**Color Constancy Loss,** based on the Gray-World color constancy hypothesis [37], states that the average intensity of any channel among the RGB channels should be gray. Therefore, we propose Color Constancy Loss  $\mathcal{L}_{col}$  as a measure of whether the overall color of the enhanced image is correct.

$$\mathcal{L}_{col} = \sum_{v(k,l) \in \varepsilon} (E_{avg}^k - E_{avg}^l)^2, \varepsilon = \{(R, G), (R, B), (G, B)\} \quad (20)$$

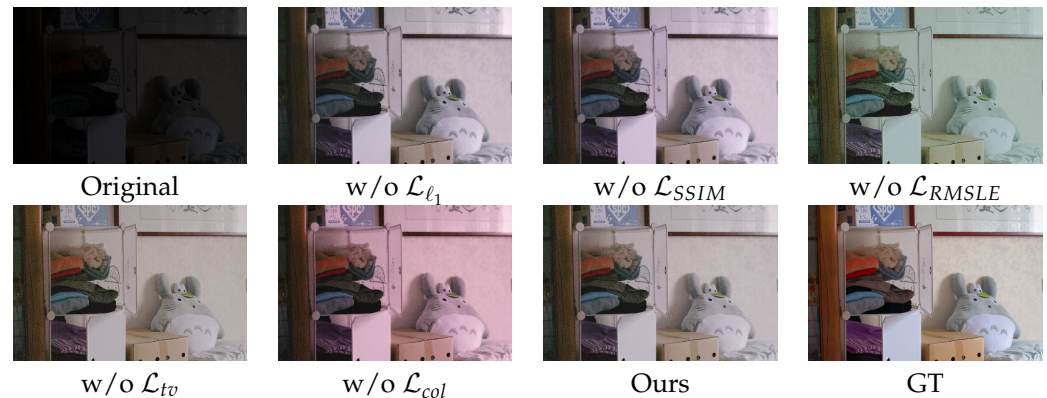
where  $E_{avg}^k$  denotes the average intensity value of the  $k$  channel in the enhanced image and  $E_{avg}^l$  denotes the average intensity value of the  $l$  channel in the enhanced image.

$\mathcal{L}_{col}$  loss focuses on maintaining color balance across RGB channels. It ensures overall enhancement without color shifts, thereby improving the overall visual quality of the enhanced images.

The total loss in this paper can be represented as Equation (21); it is important to note that no special hyperparameters need to be set for the loss functions in the equation. The outcome of each loss function is solely dependent on the enhanced image and the ground truth image.

$$\mathcal{L}_{total} = \mathcal{L}_{\ell_1} + \mathcal{L}_{RMSLE} + \mathcal{L}_{SSIM} + \mathcal{L}_{tv} + \mathcal{L}_{col} \quad (21)$$

Within the total loss function, we utilize three full-reference loss functions ( $\mathcal{L}_{\ell_1}$ ,  $\mathcal{L}_{RMSLE}$ ,  $\mathcal{L}_{SSIM}$ ) and two reference-independent loss functions ( $\mathcal{L}_{tv}$ ,  $\mathcal{L}_{col}$ ).  $\mathcal{L}_{\ell_1}$  efficiently calculates the error of the enhanced image at the pixel level but may lead to overfitting due to its high precision. To address this, we introduce  $\mathcal{L}_{RMSLE}$ , which can tolerate local large errors and help mitigate overfitting issues. Furthermore, to ensure that the enhanced images align with high-quality human perception, we use  $\mathcal{L}_{SSIM}$  to encourage brightness, contrast, and structural similarities between the enhanced image and high-quality ground truth images. To suppress noise amplification and correct color biases from post-enhancement, we employ  $\mathcal{L}_{tv}$  and  $\mathcal{L}_{col}$  functions in our final total loss function. The combination of these five loss functions contributes significantly to producing visually high-quality enhanced images while ensuring robustness. The individual impact of each loss function can be observed in Figure 7.



**Figure 7.** The impact of each loss function: when we train the model without a particular loss function, the quality of the enhanced images is noticeably affected. The quantitative analysis can be found in section 4.3.

## 4. Experiments

### 4.1. Implementation Details

**Experimentation Platform:** We implement our model on a Ubuntu 22.04 system equipped with an Intel Core i5-4690 3.50 GHz CPU from Intel corporation, Santa Clara, California, United States, an Nvidia GeForce RTX 3090 graphics card from Nvidia corporation, Santa Clara, California, United States, and 16 GB of memory from Samsung, Suwon-si, South Korea. The deep learning training framework used for the model was PyTorch 1.13.1.

**Training Hyperparameters:** We employed the Adam optimizer with exponential decay rate parameters  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ , setting the initial learning rate to  $5 \times 10^{-5}$ . Additionally, we employed the OneCycleLR learning rate scheduler to dynamically adjust the learning rate during training, and the training duration for each experiment consisted of 20 epochs.

**Datasets:** Three different datasets were used to demonstrate the performance of our model in various low-light enhancement (LLE) tasks.

For indoor LLE tasks, the Low-Light (LOL) [23,27] dataset is used in this paper to prove our model's performance. LOL has two versions: LOLV1 and LOLV2. LOLV1 consists of 485 training data and 15 test data with  $400 \times 600$  resolution images, while LOLV2 comprises the real captured dataset and synthetic dataset. The real captured dataset includes 689 training data and 100 test data with  $400 \times 600$  resolution. The synthetic dataset consists of 900 training data and 100 test data with  $384 \times 384$  resolution.

For outdoor LLE tasks, we apply the MIT-Adobe FiveK (MIT5K) [38] dataset to test our model. The MIT5K dataset collected 5000 images, and each image was enhanced by A, B, C, D, and E experts using Adobe Lightroom. In our training, we only used expert C enhanced images as ground truth images. Following the settings in the paper [21], we split 4500 images for training and 500 images for testing.

For high-contrast image LLE tasks, we utilized the local color distributions prior (LCDP) [39] dataset. This dataset comprises 1415 training images, 100 validation images, and 218 test images. The LCDP dataset is particularly well suited for assessing a model's ability to enhance images containing both overexposed and underexposed regions.

### 4.2. Performance Evaluation

Compared with real-time LLE state-of-the-art (SOTA) methods such as LIME [2], ZeroDCE [5] and IAT [6] on the LOL dataset, our method has a highly competitive performance in both enhancement quality and efficiency, as shown in Table 1. Our method achieves a high level of capability (high PSNR and SSIM) of LLE while just utilizing the minimum number of parameters and least FLOPs. While our method does not achieve the highest PSNR and SSIM values among real-time LLE models, its performance is comparable to SOTA models in terms of human perceptual experience, as shown in Figure 8.

**Table 1.** Experimental results on LOL-V1 and LOL-V2-real datasets with other SOTA models in PSNR, SSIM, and efficiency metrics. The results highlight that our model achieves comparable performance to the state-of-the-art model while leveraging significantly fewer computational resources (0.512 GFLOPs) and parameters (4.7K), showcasing its efficiency.

Methods	LOL-V1		LOL-V2-Real		Efficiency	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	FLOPs (G) ↓	#Params (M) ↓
LIME [2]	16.67	0.560	15.24	0.470	-	-
Zero-DCE [5]	14.83	0.531	14.32	0.511	2.53	0.08
RetiNexNet [23]	16.77	0.562	18.37	0.723	587.47	0.84
MBLLEN [22]	17.90	0.702	18.00	0.715	19.95	20.47
DRBN [27]	19.55	0.746	20.13	0.820	37.79	0.58
3D-LUT [40]	16.35	0.585	17.59	0.721	7.67	0.6
KIND [4]	20.86	0.790	19.74	0.761	356.72	8.16
UFormer [30]	16.36	0.771	18.82	0.771	12.00	5.29
IPT [41]	16.27	0.504	19.80	0.813	2087.35	115.63
MAXIM [42]	23.43	0.863	22.86	0.818	216.00	14.14
IAT [6]	23.38	0.809	23.50	0.824	1.44	0.09
IAT (local) [6]	20.20	0.782	20.30	0.789	1.31	0.02
Ours	19.4635	0.6144	20.6379	0.6878	0.512	0.0047

The first, second, and third best results are highlighted in red, green, and blue, respectively. The input image size is  $400 \times 600$ . ↑ represents higher is better. ↓ represents lower is better. # represents the number of.

From Table 1, we can observe that our method only utilizes 5K parameters to achieve the same performance for which some models require over a million parameters. Moreover, our model's FLOPs is only 0.512 G, indicating that our model can execute real-time LLE of high-resolution images even on low-performance devices.

The LOL dataset is a well-known basic dataset for LLE tasks, but most of its scenes are indoors. In practical applications, we also need to enhance low-light images from outdoor scenes. The MIT-Adobe FiveK dataset includes a variety of dark images from outdoor environments. When we trained our model on this dataset, we achieved the results shown in Table 2, and the results demonstrate that even for the outdoor dark images, our model can achieve a decent enhancement performance.

**Table 2.** Experimental results on MIT-Adobe FiveK [38] dataset. Quantitative comparison with other SOTA methods on outdoor LLE dataset.

Metric	White-Box [43]	DPED [44]	D-UPE [21]	3D-LUT [40]	IAT [6]	Ours
PSNR ↑	18.57	21.76	23.04	25.21	25.32	20.68
SSIM ↑	0.701	0.871	0.893	0.922	0.920	0.832
#Params (M) ↓	-	-	1.0	0.6	0.09	0.0047

The best results are highlighted in red. ↑ represents higher is better. ↓ represents lower is better. # represents the number of.

Enhancing the high-contrast images containing both overexposed and underexposed regions poses another significant challenge in LLE tasks. The LLE model must enhance the underexposed regions while reducing the brightness in overexposed areas. Remarkably, our model also achieved outstanding performance on the LCDP dataset (a dataset created specifically for such tasks), as shown in Table 3.

In our performance evaluation, we compared our proposed method with SOTA LLE models on the LOL-V1 and LOL-V2-real datasets. Despite utilizing only 4.7K parameters and 0.512 G FLOPs, our method achieved competitive performance in terms of PSNR and SSIM metrics, showcasing its efficiency in real-time LLE tasks. Although our model did not achieve the highest PSNR and SSIM values among real-time LLE models, it demonstrated comparable performance to SOTA models in terms of human perceptual experience. Additionally, we evaluated our method on the MIT-Adobe FiveK dataset for outdoor scenes and



observed promising results, highlighting the versatility of our model. Furthermore, our model showed excellent performance on the LCDP dataset for handling high-contrast images, surpassing some SOTA models in terms of PSNR and SSIM metrics while maintaining a significantly lower number of parameters ( $<0.005$  M) compared to other methods. Overall, our method excels in achieving a balance between enhancement quality, computational efficiency, and reliability across various LLE tasks and datasets.

**Table 3.** Experimental results on LCDP [39] dataset. Quantitative comparison with other SOTA methods on high-contrast image LLE dataset.

Metric	LIME [2]	RetinexNet [23]	Zero-DCE [5]	D-UPE [21]	LCDP [39]	Ours
PSNR $\uparrow$	17.335	19.250	12.587	20.970	23.239	20.016
SSIM $\uparrow$	0.686	0.704	0.653	0.818	0.842	0.793
#Params (M) $\downarrow$	-	0.84	0.08	1.0	0.28	0.0047

The best results are highlighted in red.  $\uparrow$  represents higher is better.  $\downarrow$  represents lower is better. # represents the number of.

#### 4.3. Ablation Study

In our study, we deployed five distinct loss functions and a correction module to enhance our model's performance. Each component addresses specific aspects crucial for image enhancement, such as image smoothness, color balance, error tolerance, human perceptual alignment, and basic error measurement. Our goal was to evaluate the impact of each loss function and the correction module on the overall model performance through ablation experiments, as summarized in Table 4.

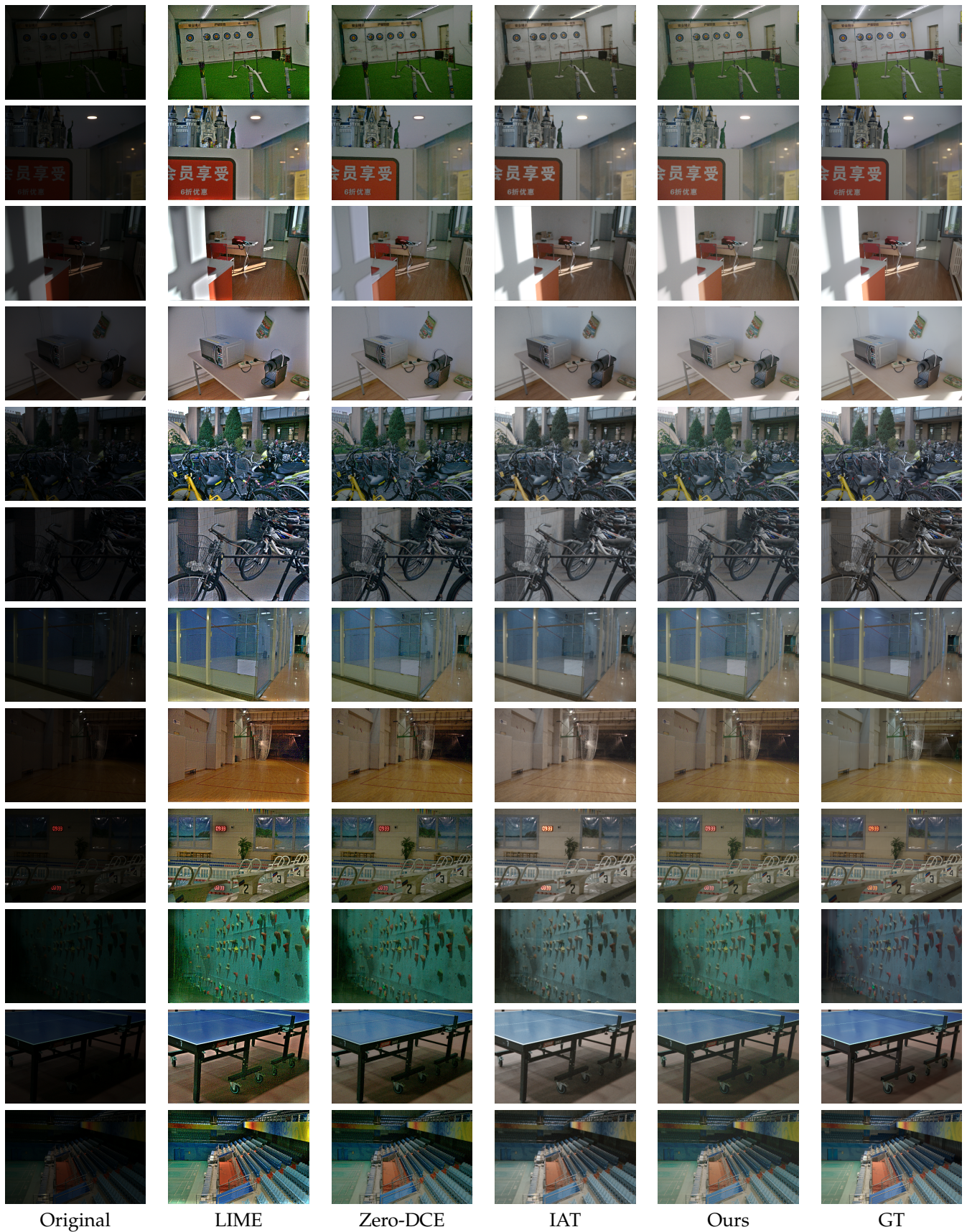
**Table 4.** Ablation experiment results: impact of removing correction module or various loss functions on model performance. We observed that excluding either the correction module or individual loss functions leads to a degradation in our model's performance

Methods	LOL-V1		LOL-V2-Real	
	PSNR $\uparrow$	SSIM $\uparrow$	PSNR $\uparrow$	SSIM $\uparrow$
w/o CM	17.8738	0.5816	18.6065	0.5916
w/o $\mathcal{L}_{\ell_1}$	18.8969	0.6440	19.5882	0.6877
w/o $\mathcal{L}_{SSIM}$	18.3100	0.5754	19.8165	0.6646
w/o $\mathcal{L}_{RMSLE}$	18.8289	0.6354	19.5118	0.6850
w/o $\mathcal{L}_{tv}$	19.2315	0.6125	18.7706	0.6238
w/o $\mathcal{L}_{col}$	18.3861	0.6227	20.0880	0.7022
Ours	19.4635	0.6144	20.6379	0.6878

The best results are highlighted in red. CM is Correction module.  $\uparrow$  represents higher is better.  $\downarrow$  represents lower is better.

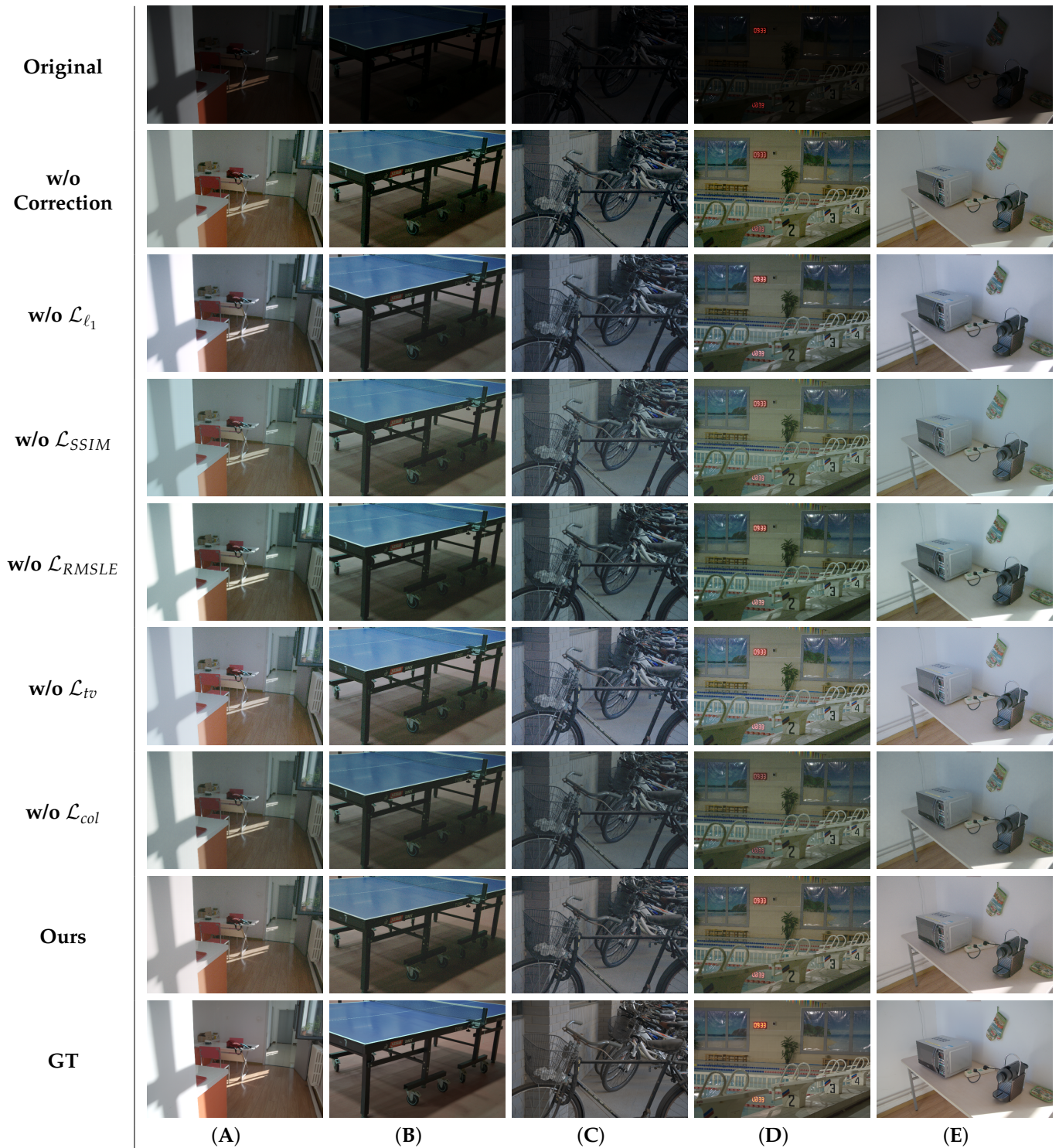
Observing Table 4, we found that excluding either the correction module or individual loss functions led to a noticeable degradation in model performance across both LOL-V1 and LOL-V2-real versions.  $\mathcal{L}_{\ell_1}$ ,  $\mathcal{L}_{SSIM}$ ,  $\mathcal{L}_{RMSLE}$ , and  $\mathcal{L}_{col}$  contributed significantly to improving the PSNR and SSIM metrics in both datasets, showcasing their importance in enhancing image quality. A visual difference is displayed in Figure 9.

Our ablation study underscores the importance of selected loss functions and integrated correction module in our model. They are crucial in improving image enhancement quality, contributing collectively to overall model performance. However, further research and refinement in addressing persistent challenges like color bias and noise amplification are essential for advancing low-light image enhancement techniques.



**Figure 8.** Qualitative comparison results on LOL-V2-real [27] dataset compared with enhancement methods LIME [2], Zero-DCE [5] and IAT [6]. Our model outperforms other SOTA models in certain cases, but we still encounter challenges such as noise and color inaccuracy.





**Figure 9.** Results of our model after removing the color correction module or individual loss functions, respectively. A–E represent the test images sampled from LOLV2 dataset.

## 5. Conclusions

To develop a lightweight, real-time, and efficient low-light enhancement (LLE) model applicable to low-performance devices such as surveillance systems, autonomous vehicles, UAVs, and smartphones, we innovatively combined traditional LMCV algorithms with deep learning techniques. We proposed the Dark Image Increment Estimation (DIIE) method, which estimates the incremental changes between dark images and the three-channel transmission map. Based on this method, we constructed a deep learn-

ing model with the Transmission Enhancement Module (TEM) and Correction Module (CM). Our model seamlessly integrates the lightweight and interpretability of traditional algorithms with the efficiency, adaptability, and robustness of deep learning methods.

Despite having only 39.1% and 23% of the parameters and computational load of the latest real-time SOTA model IAT (local), our model achieves a significant LLE performance. This advancement contributes significantly to providing high-quality real-time LLE models on performance-constrained devices.

While our contribution is substantial, our method still has some limitations. For instance, we only conducted qualitative analysis of the loss functions without quantitative analysis. Although there are no hyperparameters to adjust within each loss function, we can introduce weights for each loss function in the total loss function equation such as:

$$\mathcal{L}_{total} = w_1\mathcal{L}_{\ell_1} + w_2\mathcal{L}_{RMSLE} + w_3\mathcal{L}_{SSIM} + w_4\mathcal{L}_{tv} + w_5\mathcal{L}_{col} \quad (22)$$

In our future work, we will perform quantitative analysis to determine the optimal combination of weights  $w_1, w_2, w_3, w_4$ , and  $w_5$ .

In addition, our model still encounters notable challenges in terms of image color correction and denoising. Further refinement of the TEM and CM is necessary. Moreover, we have observed mutual impacts on learning efficiency between the TEM and CM modules during initial model training. In future work, we intend to train these modules independently to alleviate their negative interactions and enhance the overall model performance.

**Author Contributions:** Conceptualization, Q.D. and D.L.; methodology, Q.D. and D.C.; software, Q.D.; validation, Q.D.; formal analysis, Q.D.; investigation, Q.D.; resources, Q.D. and D.C.; data curation, Q.D., H.J. and D.C.; writing—original draft preparation, Q.D.; writing—review and editing, Q.D. and D.L.; visualization, Q.D.; supervision, D.L.; project administration, D.L.; funding acquisition, D.L. and H.J. All authors have read and agreed to the published version of the manuscript.

**Funding:** This paper contains the results of a study on the “Leaders in Industry–University Cooperation 3.0” Project, which is supported by the Ministry of Education and National Research Foundation of Korea.

**Data Availability Statement:** The original data presented in the study are openly available in LOLV1 and LOLV2 dataset at <https://github.com/flyywh/CVPR-2020-Semi-Low-Light>; MIT5K dataset at <https://data.csail.mit.edu/graphics/fivek/> and LCDP dataset at <https://www.why.site/paper/lcdp>.

**Conflicts of Interest:** Author Dongwon Choo and Hyochul Ji were employed by the company InBic Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

ASM	atmospheric scattering model
CM	Correction Module
DCP	dark channel prior
DIIE	Dark Image Increment Estimation
FLOPs	floating point operations
GT	ground truth
LLE	low-light enhancement
LMCV	Local Maximum Color Value
LOL	low-light dataset
PSNR	peak signal to noise ratio
SOTA	state of the art
SSIM	structural similarity
TEM	Transmission Estimation Module



## References

- Wang, Y.F.; Liu, H.M.; Fu, Z.W. Low-light image enhancement via the absorption light scattering model. *IEEE Trans. Image Process.* **2019**, *28*, 5679–5690. [[CrossRef](#)] [[PubMed](#)]
- Guo, X.; Li, Y.; Ling, H. LIME: Low-light image enhancement via illumination map estimation. *IEEE Trans. Image Process.* **2016**, *26*, 982–993. [[CrossRef](#)] [[PubMed](#)]
- Dong, X.; Wen, J. *Low Lighting Image Enhancement Using Local Maximum Color Value Prior*; Frontiers of Computer Science; Springer: Berlin/Heidelberg, Germany, 2016; Volume 10; pp. 147–156.
- Zhang, Y.; Zhang, J.; Guo, X. Kindling the darkness: A practical low-light image enhancer. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 1632–1640.
- Guo, C.; Li, C.; Guo, J.; Loy, C.C.; Hou, J.; Kwong, S.; Cong, R. Zero-reference deep curve estimation for low-light image enhancement. In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1780–1789.
- Cui, Z.; Li, K.; Gu, L.; Su, S.; Gao, P.; Jiang, Z.; Qiao, Y.; Harada, T. You Only Need 90K Parameters to Adapt Light: A Light Weight Transformer for Image Enhancement and Exposure Correction. In Proceedings of the 33rd British Machine Vision Conference 2022, London, UK, 21–24 November 2022.
- Panagiotou, S.; Bosman, A.S. Denoising Diffusion Post-Processing for Low-Light Image Enhancement. *arXiv* **2023**, arXiv:2303.09627.
- Zhou, D.; Yang, Z.; Yang, Y. Pyramid Diffusion Models for Low-light Image Enhancement. *arXiv* **2023**, arXiv:2305.10028.
- Wu, G.; Jin, C. DiffLIE: Low-Light Image Enhancement based on Deep Diffusion Model. In Proceedings of the 2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS), Chengdu, China, 16–18 June 2023; pp. 522–526.
- Lv, X.; Dong, X.; Jin, Z.; Zhang, H.; Song, S.; Li, X. L 2 DM: A Diffusion Model for Low-Light Image Enhancement. In Proceedings of the Chinese Conference on Pattern Recognition and Computer Vision (PRCV), Shenzhen, China, 14–17 October 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 130–145.
- Yi, X.; Xu, H.; Zhang, H.; Tang, L.; Ma, J. Diff-retinex: Rethinking low-light image enhancement with a generative diffusion model. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 12302–12311.
- McCartney, E.J. *Optics of the Atmosphere: Scattering by Molecules and Particles*; John Wiley and Sons, Inc.: New York, NY, USA, 1976.
- Land, E.H. *The Retinex*; The Scientific Research Honor Society: Triangle Park, NC, USA, 1965; pp. 217–227.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
- Cao, Y.; Li, S.; Liu, Y.; Yan, Z.; Dai, Y.; Yu, P.S.; Sun, L. A comprehensive survey of ai-generated content (aigc): A history of generative ai from gan to chatgpt. *arXiv* **2023**, arXiv:2303.04226.
- Liu, W.; Zhao, P.; Zhang, B.; Xuan, W. A low-light image enhancement method based on atmospheric scattering model. In Proceedings of the 2022 2nd International Conference on Computer Graphics, Image and Virtualization (ICCGIV), Chongqing, China, 23–25 September 2022; pp. 145–150.
- Jeon, J.J.; Park, J.Y.; Eom, I.K. Low-light image enhancement using gamma correction prior in mixed color spaces. *Pattern Recognit.* **2024**, *146*, 110001. [[CrossRef](#)]
- Makwana, D.; Deshmukh, G.; Susladkar, O.; Mittal, S. LIVENet: A novel network for real-world low-light image denoising and enhancement. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 1–10 January 2024; pp. 5856–5865.
- Li, C.; Guo, C.; Loy, C.C. Learning to enhance low-light image via zero-reference deep curve estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 4225–4238. [[CrossRef](#)] [[PubMed](#)]
- Ma, L.; Liu, R.; Zhang, J.; Fan, X.; Luo, Z. Learning deep context-sensitive decomposition for low-light image enhancement. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 5666–5680. [[CrossRef](#)] [[PubMed](#)]
- Wang, R.; Zhang, Q.; Fu, C.W.; Shen, X.; Zheng, W.S.; Jia, J. Underexposed photo enhancement using deep illumination estimation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 6849–6857.
- Lv, F.; Lu, F.; Wu, J.; Lim, C. MBLLEN: Low-Light Image/Video Enhancement Using CNNs. *Proc. BMVC* **2018**, *220*, 4.
- Wei, C.; Wang, W.; Yang, W.; Liu, J. Deep retinex decomposition for low-light enhancement. *arXiv* **2018**, arXiv:1808.04560.
- Liu, R.; Ma, L.; Zhang, J.; Fan, X.; Luo, Z. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 10561–10570.
- Lu, K.; Zhang, L. TBEFN: A two-branch exposure-fusion network for low-light image enhancement. *IEEE Trans. Multimed.* **2020**, *23*, 4093–4105. [[CrossRef](#)]
- Yang, W.; Wang, W.; Huang, H.; Wang, S.; Liu, J. Sparse gradient regularized deep retinex network for robust low-light image enhancement. *IEEE Trans. Image Process.* **2021**, *30*, 2072–2086. [[CrossRef](#)] [[PubMed](#)]
- Yang, W.; Wang, S.; Fang, Y.; Wang, Y.; Liu, J. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 3063–3072.



28. Zhang, Y.; Guo, X.; Ma, J.; Liu, W.; Zhang, J. Beyond brightening low-light images. *Int. J. Comput. Vis.* **2021**, *129*, 1013–1037. [[CrossRef](#)]
29. Cai, Y.; Bian, H.; Lin, J.; Wang, H.; Timofte, R.; Zhang, Y. Retinexformer: One-stage Retinex-based Transformer for Low-light Image Enhancement. *arXiv* **2023**, arXiv:2303.06705.
30. Wang, Z.; Cun, X.; Bao, J.; Zhou, W.; Liu, J.; Li, H. Uformer: A General u-Shaped Transformer for Image Restoration. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17683–17693.
31. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.H. Restormer: Efficient Transformer for High-Resolution Image Restoration. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 5728–5739.
32. Jie, H.; Zuo, X.; Gao, J.; Liu, W.; Hu, J.; Cheng, S. Llformer: An efficient and real-time lidar lane detection method based on transformer. In Proceedings of the 2023 5th International Conference on Pattern Recognition and Intelligent Systems, Shenyang, China, 28–30 July 2023; pp. 18–23.
33. Brateanu, A.; Balmez, R.; Avram, A.; Orhei, C. LYT-Net: Lightweight YUV Transformer-based Network for Low-Light Image Enhancement. *arXiv* **2024**, arXiv:2401.15204.
34. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the 18th International Conference—Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015, Munich, Germany, 5–9 October 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
35. Targ, S.; Almeida, D.; Lyman, K. Resnet in resnet: Generalizing residual architectures. *arXiv* **2016**, arXiv:1603.08029.
36. Mansour, Y.; Heckel, R. Zero-Shot Noise2Noise: Efficient Image Denoising without any Data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 14018–14027.
37. Cepeda-Negrete, J.; Sanchez-Yanez, R.E. Gray-world assumption on perceptual color spaces. In Proceedings of the 6th Pacific-Rim Symposium, PSIVT 2013—Image and Video Technology, Guanajuato, Mexico, 28 October–1 November 2013; Springer: Berlin/Heidelberg, Germany, 2014; pp. 493–504.
38. Bychkovsky, V.; Paris, S.; Chan, E.; Durand, F. Learning Photographic Global Tonal Adjustment with a Database of Input/Output Image Pairs. In Proceedings of the Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011.
39. Wang, H.; Xu, K.; Lau, R.W. Local color distributions prior for image enhancement. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 343–359.
40. Zeng, H.; Cai, J.; Li, L.; Cao, Z.; Zhang, L. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 2058–2073. [[CrossRef](#)] [[PubMed](#)]
41. Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; Gao, W. Pre-trained image processing transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12299–12310.
42. Tu, Z.; Talebi, H.; Zhang, H.; Yang, F.; Milanfar, P.; Bovik, A.; Li, Y. Maxim: Multi-axis mlp for image processing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LO, USA, 18–24 June 2022; pp. 5769–5780.
43. Hu, Y.; He, H.; Xu, C.; Wang, B.; Lin, S. Exposure: A white-box photo post-processing framework. *ACM Trans. Graph. (TOG)* **2018**, *37*, 1–17. [[CrossRef](#)]
44. Ignatov, A.; Kobyshev, N.; Timofte, R.; Vanhoey, K.; Van Gool, L. Dslr-quality photos on mobile devices with deep convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3277–3285.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.