

## Article

# ResGASP-GAN: A Residual Group-Normalized ASPP-SE GAN with a PatchGAN Discriminator for Low-Light Image Enhancement

Fernando Daniel Hernandez-Gutierrez <sup>1</sup>, Paula Dalida Bravo-Aguilar <sup>2</sup>, Emmanuel Ovalle-Magallanes <sup>3</sup>,  
Mario Alberto Ibarra-Manzano <sup>1</sup>, Jose Ruiz-Pinales <sup>1</sup> and Juan Gabriel Avina-Cervantes <sup>1,\*</sup>

<sup>1</sup> Telematics and Digital Signal Processing Research Groups (CAs), Engineering Division, Campus Irapuato-Salamanca, University of Guanajuato, Salamanca 36885, Mexico; fd.hernandezgutierrez@ugto.mx (F.D.H.-G.); ibarram@ugto.mx (M.A.I.-M.); pinales@ugto.mx (J.R.-P.)

<sup>2</sup> PrepaTec Irapuato, Tecnológico de Monterrey, Paseo Mirador del Valle 445, Irapuato 36670, Mexico; paula.dalida@tec.mx

<sup>3</sup> Dirección de Investigación y Doctorado, Facultad de Ingenierías y Tecnologías, Universidad La Salle Bajío, Av. Universidad 602. Col. Lomas del Campestre, León 37150, Mexico; eoalle@lasallebajio.edu.mx

\* Correspondence: avina@ugto.mx

## Abstract

Low-light color image enhancement remains a challenging task for vision-based decision systems, which must simultaneously address illumination correction, noise suppression, contrast recovery, and color preservation from a single degraded observation. This study proposes ResGASP-GAN, a GAN-based low-light image enhancement framework built around a residual-output generator that integrates batch-size-independent normalization, multi-scale contextual aggregation, and channel-wise feature recalibration within a conditional adversarial setting. Group Normalization is integrated into the proposed generator to reduce dependence on batch statistics during small-batch training, while a Squeeze-and-Excitation (SE) module adaptively enables channel-wise feature recalibration and helps preserve structural and chromatic information. The proposed generator uses reflection-padded convolutions to reduce boundary artifacts, and it features a multi-scale bottleneck composed of dilated residual blocks and Atrous Spatial Pyramid Pooling to capture spatially varying illumination patterns. The model is optimized using a compound objective that combines an adversarial term with an  $\ell_1$  reconstruction loss, balancing perceptual realism with pixel-level fidelity. Experimental evaluation employed the LOL-v1, LOL-v2-Real, and LOL-v2-Synthetic datasets using reference-based metrics: PSNR, SSIM, and LPIPS. No-reference perceptual metrics were also used, including NIQE and BRISQUE. The results indicate that the proposed method achieves competitive structural similarity and visual image quality on LOL-v2-Real, competitive reconstruction performance on LOL-v1, and good generalization on LOL-v2-Synthetic, with the second-best metrics of PSNR = 22.30 dB, SSIM = 0.9154, and LPIPS = 0.1022 among the reported methods on this last dataset. In contrast, using no-reference metrics, this study achieved very good results, with the lowest BRISQUE of 10.4540 and a competitive NIQE of 4.4396, providing high-quality visual-perceptual reconstruction. Overall, the proposed architecture provides a competitive GAN-based alternative for low-light image enhancement, combining residual connections, multi-scale contextual modeling, and channel-wise feature refinement. This architecture provides the lowest inference time over all discussed models, which is highly required in real-time outdoor applications such as robot navigation and mapping.



Academic Editor: Ioannis Tsoulos

Received: 9 June 2026

Revised: 14 July 2026

Accepted: 15 July 2026

Published: 17 July 2026

**Copyright:** © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

**Keywords:** Low-Light Image Enhancement; Generative Adversarial Networks; ResUNet; PatchGAN; Group Normalization; Squeeze-and-Excitation Attention

**MSC:** 68U10; 65D19; 90-08

## 1. Introduction

Dim lighting conditions degrade image quality by increasing noise, reducing contrast, and distorting color, thereby impairing key computer vision tasks. Enhancing image visibility under these conditions is essential for applications such as autonomous driving [1], surveillance [2], medical imaging [3], robotics [4], and consumer photography [5]. Accordingly, developing models that reliably enhance low-light images has become increasingly important for recovering discriminative details and reducing error propagation in downstream analysis. As a result, low-light image enhancement (LLIE) has become an active interdisciplinary research area.

In low-light conditions, cameras improve visibility primarily through three strategies [6]: The first is to increase exposure time, which may introduce motion blur. The second is to increase the camera's ISO sensitivity, allowing the image sensor gain to amplify the captured signal. However, this method also amplifies sensor noise, degrading image quality. Finally, scene illumination can be enhanced artificially using external lighting devices, such as flash systems. Nevertheless, this approach may lead to overexposure and an unnatural appearance.

Despite their simplicity and low computational cost, conventional low-light enhancement techniques often struggle to deliver robust, visually consistent results across diverse scenes [7]. Gray-level transformations often fail to handle wide luminance ranges effectively, leading to the loss of detail in overexposed and underexposed regions of the image. Similarly, histogram-based methods can amplify noise and overenhance certain regions, resulting in uneven contrast across the image. Retinex-based methods can suffer from color distortions and halo artifacts near bright light sources. These methods also rely on difficult-to-tune parameters that often fail to generalize across images.

Despite these advancements, LLIE remains challenging because illumination, contrast, color fidelity, and noise must be corrected simultaneously from a single degraded observation [8]. This difficulty is further compounded by highly diverse real-world lighting conditions, which yield complex, spatially varying illumination patterns.

Generative Adversarial Networks (GANs) support LLIE by using a generator to produce enhanced images while a discriminator distinguishes them from well-lit references [9]. GAN-based methods also demonstrate competitive performance under extremely low-light conditions [10]. These frameworks also support unpaired training, learning low-light-to-normal-light translation without requiring aligned image pairs in the training set [11]. Nevertheless, existing GAN-based LLIE architectures still struggle to jointly recover spatially varying illumination, preserve structural details, and maintain color consistency, often resulting in noise amplification, artifacts, color deviations, or over- or under-exposure [12]. These limitations motivate the proposed design, which is based on Group Normalization, multi-scale contextual modeling, and channel-wise feature recalibration.

The main contributions of this study are summarized as follows:

1. A ResUNet-like generator is introduced for low-light image enhancement. Training is often constrained to small batch sizes, which can destabilize optimization and hinder convergence. To alleviate this, Group Normalization (GN) is employed and

coupled with reflection-padded convolutions. This combination provides more stable optimization for training while using small mini-batches.

2. A multi-level context mechanism is implemented to address spatially diverse illumination. Contextual information is extracted using an Atrous Spatial Pyramid Pooling (ASPP) module. ASPP employs parallel dilated convolutions to capture multi-scale spatial features, while dilated residual blocks further refine the contextual representation. This design enables the model to account for broad illumination fluctuations while maintaining sensitivity to subtle, fine-scale light variations.
3. Lightweight SE modules are incorporated into the encoder–decoder pathway to perform channel-wise feature recalibration. By adaptively emphasizing informative channels and attenuating responses to less informative features, these modules help preserve structural details and chromatic consistency during illumination correction. This design is particularly useful in aggressive brightening scenarios, where color shifts, contrast degradation, and loss of fine textures may otherwise occur.

## 2. Related Works

Traditional image enhancement methods remain widely adopted due to their simplicity and computational efficiency, with histogram equalization, gamma correction, and Retinex-based techniques among the most commonly used. However, these techniques can produce over-enhancement or visible artifacts when applied to complex real-world scenes [13].

In low-light conditions, preserving color fidelity is essential for driver perception and object localization. Wu et al. [14] presented the Multi-Feature Domain Fusion Diffusion Framework (WFDiff) on the LOL-v1 and Hazy-Dark Moderate datasets. They combined Fourier-wavelet dual-domain features and the inverse diffusion process via differentiable operators to construct a multi-scale degradation-correction system. Specifically, the wavelet transform captures high-frequency details and edge structures in the spatial domain, improving the prediction process. A major drawback of this approach is the slow inference speed typically associated with diffusion models, leading to high computational costs and reliance on high-end GPUs.

Cha and Tang [15] proposed a zero-shot method for enhancing low-light images by incorporating light scattering effects into the Retinex Decomposition (ZESRD) model and evaluated it on the Low-Light (LOL) and Scene Image Contrast Enhancement (SICE) datasets. The method extends Retinex decomposition with the Koschmieder scattering model to overcome the limited ability of conventional Retinex formulations to model light scattering, enabling the estimation of scattering attenuation and ambient light to correct contrast loss and uneven illumination. However, limitations arise in complex light-transport scenarios, such as multi-layer interfaces, where the isotropic scattering assumption may no longer hold.

UNet architectures have been used across multiple fields due to their end-to-end learning capability. From this perspective, Ai and Kwon [2] proposed a UNet-based architecture incorporating an attention gate on the See in the Dark (SID) dataset. They implemented a modified UNet that incorporates an attention gate module into the base UNet. This modification transformed extremely dark images into higher-contrast, more visible ones under low-light conditions using a standard computer vision approach, without requiring raw sensor data.

The proposed architecture uses two processing branches: The first focuses on Multi-layer Feature Aggregation (MFA) to extract details from low-level layers during image enhancement. The second branch provides color histogram blocks and dense denoising blocks. Both branches consist of three modules: encoder, decoder, and enhancer.

Residual artifacts may remain visible under extreme lighting conditions, and the limited availability of large-scale, unpaired data limits generalization. Additionally, conventional full-reference metrics such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) may not reflect perceptual quality, and severe noise can lead to slight blurring and loss of fine details.

Zainab et al. [16] proposed a lightweight CNN-based enhancement framework to improve efficiency and reduce inference time on the LOL-v1 dataset. The model consisted of five modules. The preprocessing module included five convolutional layers with  $3 \times 3$  filters. The LLIE module computes a transformation between different lighting conditions, and its output subsequently feeds into the fusion module, which uses an attention-and-refinement unit to merge them. Although efficient and suitable for low-power deployment, the architecture can struggle with highly noisy images.

Jiang et al. [17] proposed a Semantic-aware Knowledge-guided Framework (SKF) in the LOL-v1 and LOL-v2 datasets. The framework integrates a Semantic Feature Enhancement Module for integrating hierarchical semantic features, a Semantic-Guided Color Histogram Loss to enforce color consistency, and a Semantic-Guided Adversarial Loss to enhance perceptual realism. However, SKF still requires further optimization for lightweight design, real-time inference, and broader semantic integration to better support real-world, resource-constrained deployments.

Beyond natural-image LLIE tasks, Chen et al. [3] proposed a lightweight Denoising Diffusion Probabilistic Model (DDPM), termed LightTDif, and evaluated it on the EndoVis17 and EndoVis18 datasets for surgical instrument segmentation. They also introduced a Temporal Light Unit (TLU) to improve training stability and segmentation performance. The TLU links time steps to denoised image features, establishing temporal dependencies and enhancing denoising results. Nevertheless, the reported results showed at most similar performance to the evaluated baselines. He et al. [18] introduced a Swin Transformer-based Light-enhancing Generative Adversarial Network (SwinLightGAN) on the LOL-v1 and SICE datasets. The architecture integrates a Residual Jump UNet Generator (RSUnet) and a Swin Transformer Enhanced Illumination Network (SwinTE-HIEN) to simultaneously enhance image brightness and sharpness.

Yin et al. [19] proposed a Differentiable Histogram-guided Unsupervised Retinex Enhancement (DHURE) on the LOL-v1 and SICE datasets. This method decomposes the paired low-light images into Retinex components—the illumination and reflectance maps—using a P-Net, L-Net, and R-Net. Song [20] introduced a noise-resilient LLIE framework that incorporates CLIP-guided loss (CLIP-LLA) and Pixel-Reordering Subsampling (PRS) into the Retinexformer backbone, and then evaluated it on the LOL-v1 and LOL-v2 datasets. CLIP-LLA enhances perceptual realism and global tone, while PRS-based multi-scale training improves denoising robustness without architectural changes or additional inference costs.

Zeng et al. [21] introduced TCSTNet, a text-driven color-style transfer network designed for low-light image enhancement on the Multi-Exposure Gradient (MEG), Low-Light (LOL), and Low-Light version 2 (LOL-v2) datasets. They used a UNet and a Residual Convolutional Neural Network to extract color details from the original image. The method incorporates a VGG-based style loss to guide the style transfer towards photorealistic outputs. Color saturation artifacts may occur when the brightness difference between the style and content images is significant, especially if the content image contains highly saturated regions. In addition, the method requires a more complex training procedure than competing approaches.

Zhao and Shi [22] proposed RetinexGDP, a training-free low-light image enhancement method that integrates a Retinex-based decomposition model with a generative diffusion

prior on the LOL-v1 and VE-LOL datasets. The method employs zero-shot illumination estimation and reflectance-guided diffusion sampling to enhance image visibility while preserving structural details.

### 3. Materials and Methods

GANs have garnered considerable attention for their adversarial learning framework, which involves two main components: a generator and a discriminator. The generator produces synthetic images from the training data, while the discriminator learns to distinguish between real and generated images. Both networks are trained end-to-end in a two-player minimax game until the discriminator can no longer reliably distinguish between real and generated samples. While other methods struggle with uncontrollable illumination changes and dynamic objects in the scene, GANs can generate images similar to the target image [6]. However, these architectures should be subject to appropriate constraints, as they may generate unrealistic images that do not accurately reflect the intended illumination conditions.

#### 3.1. Spectral Normalization

Spectral normalization (SN) is a weight reparameterization method that stabilizes training by explicitly constraining the spectral norm of the weight operator in each learnable layer [23]. Specifically, the *spectral norm of a layer* is the spectral norm of its weight matrix (or, equivalently, the linear operator defined by its weights), which quantifies the maximum amplification factor induced by that operator. Formally, for a linear mapping parameterized by a weight matrix  $W$ , the spectral norm  $\sigma(W)$  is here defined as the induced Euclidean operator norm,

$$\sigma(W) = \|W\|_2 = \max_{\|h\|_2=1} \|Wh\|_2 = \sigma_{\max}(W), \quad (1)$$

where  $h$  denotes an arbitrary unit-norm input vector, and  $Wh$  is the corresponding output after applying the linear transformation;  $\sigma(W)$  measures the largest gain of the layer in the Euclidean sense. For convolutional layers, the same interpretation holds because convolution is a linear operator; in practice,  $\sigma(W)$  is estimated efficiently and used to rescale the convolutional weights [23].

For a feed-forward network  $f(\cdot)$  constructed by composing such linear layers and point-wise nonlinearities, the Lipschitz constant can be bounded by the product of the Lipschitz constants of its components [23]. In particular, when each nonlinearity is 1-Lipschitz, the network Lipschitz constant satisfies

$$\|f\|_{\text{Lip}} \leq \prod_{l=1}^L \sigma(W_l), \quad (2)$$

where  $\sigma(W_l)$  denotes the spectral norm of the weight operator of layer- $l$ . SN enforces a controlled operator norm by normalizing each layer as

$$\bar{W}_l = \frac{W_l}{\sigma(W_l)}, \quad (3)$$

which yields  $\sigma(\bar{W}_l) = 1$  by construction [23]. More generally, a target bound  $K > 0$  can be imposed through  $\bar{W}_l = (K/\sigma(W_l)) W_l$ , thereby controlling the effective layer-wise operator norm [24].

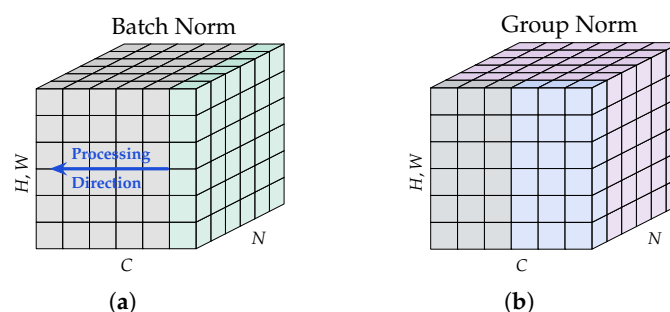
Since computing  $\sigma(W_l)$  exactly via singular value decomposition at every iteration is computationally expensive, it is typically approximated using power iteration [23]. Given unit vectors  $u$  and  $v$ , one performs

$$v \leftarrow \frac{W^T u}{\|W^T u\|_2}, \quad u \leftarrow \frac{Wv}{\|Wv\|_2}, \quad \sigma(W) \approx u^T Wv, \quad (4)$$

and then applies the normalized parameterization  $W \mapsto \bar{W}$  during training. This procedure provides an efficient approximation of the spectral norm while preserving the layer's representational capacity [23,24].

### 3.2. Group Normalization

Batch Normalization (BN) was originally introduced to mitigate internal covariate shift—that is, changes in the distribution of layer inputs during training while improving optimization stability [25]. On the other hand, batch size influences both training speed and optimization stability during the training stage [26]. To address the limitations of batch-dependent normalization, Group Normalization (GN) computes the mean and variance within each sample by dividing the channel dimension into  $G$  groups, rather than relying on mini-batch statistics [27]. Unlike BN, GN is independent of batch size and is generally more suitable for training with small mini-batches, as illustrated in Figure 1.



**Figure 1.** Illustration of Batch Normalization (BN) and Group Normalization (GN). Highlighted elements are normalized using their respective statistics (mean and variance). In BN, the statistics are computed across the mini-batch and spatial dimensions for each channel. In GN, by contrast, the statistics are computed independently for each sample across channels within each group and across spatial dimensions. (a) Batch Normalization, (b) Group Normalization. Here,  $C$  denotes the channel axis,  $N$  represents the batch axis, and  $(H, W)$  denote the spatial axes.

Mathematically, the difference between BN and GN lies in the domain over which the normalization statistics are estimated. In BN, the mean and variance are estimated over the mini-batch, whereas in GN they are computed within each channel group for each sample. BN depends on batch-level statistics, while GN performs normalization independently of the mini-batch size. These normalizations are defined as follows:

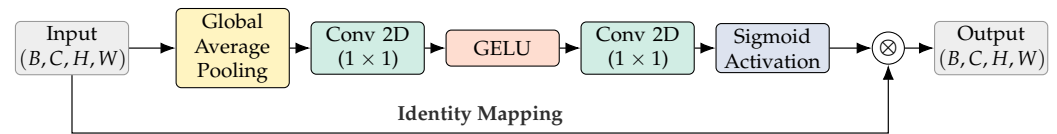
$$y_i = \gamma \hat{x}_i + \beta, \quad \forall \hat{x}_i = \begin{cases} \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, & \text{if BN} \\ \frac{x_i - \mu_{g(i)}}{\sqrt{\sigma_{g(i)}^2 + \epsilon}}, & \text{if GN} \end{cases} \quad (5)$$

where  $\mu_B$  and  $\sigma_B^2$  denote the mean and variance computed over the mini-batch in BN, whereas  $\mu_{g(i)}$  and  $\sigma_{g(i)}^2$  represent the corresponding statistics computed over the group  $g(i)$  associated with the activation  $x_i$  in GN. Here,  $\epsilon$  is a small constant for numerical stability, while  $\gamma$  and  $\beta$  are learnable scale and shift parameters.

### 3.3. SE Attention

Attention modules in low-light image enhancement adaptively modulate features to suppress noise while preserving structure and color during illumination restoration. In particular, the Squeeze-and-Excitation (SE) module adaptively recalibrates channel-wise

feature responses. The SE module captures inter-channel dependencies and selectively emphasizes informative representations while suppressing less relevant ones [28], as shown in Figure 2. This module provides an effective, lightweight attention mechanism that emphasizes illumination- and color-relevant feature responses while avoiding unnecessary architectural complexity in the generator.



**Figure 2.** Squeeze-and-Excitation (SE) attention module. Channel-wise descriptors are extracted via global average pooling and then transformed with  $1 \times 1$  convolutional layers, Gaussian Error Linear Unit (GELU) activation, and sigmoid gating to recalibrate the input feature map. The symbol  $\otimes$  is the Hadamard product.

Given a feature map  $X \in \mathbb{R}^{B \times C \times H \times W}$ , SE first applies global average pooling to obtain a compact channel descriptor

$$z = \text{GAP}(X), \quad z_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W X_c(i, j), \quad (6)$$

where  $z \in \mathbb{R}^{B \times C}$  and  $c \in \{1, \dots, C\}$ . Let  $r$  denote the reduction ratio, and define the hidden dimension as

$$C' = \max\left(1, \left\lfloor \frac{C}{r} \right\rfloor\right). \quad (7)$$

Next, the excitation stage produces channel-wise gates through two linear projections, a Gaussian Error Linear Unit (GELU) nonlinearity, and a sigmoid activation:

$$s = \phi(W_2 \delta(W_1 z + b_1) + b_2), \quad (8)$$

where  $W_1 \in \mathbb{R}^{C' \times C}$  and  $W_2 \in \mathbb{R}^{C \times C'}$  are the weight matrices of layers  $L_1$  and  $L_2$ ;  $b_1 \in \mathbb{R}^{C'}$  and  $b_2 \in \mathbb{R}^C$  represent the bias vectors of layers  $L_1$  and  $L_2$ , respectively;  $\delta(x) \approx 0.5x \times \left(1 + \tanh\left[\sqrt{\frac{2}{\pi}}(x + 0.044715x^3)\right]\right)$  is the GELU function; and  $\phi(x) = \frac{1}{1+e^{-x}}$  denotes the sigmoid function. Finally, SE recalibrates the input feature map via channel-wise scaling:

$$Y = X \odot s, \quad Y_c(i, j) = s_c X_c(i, j), \quad (9)$$

where  $\odot$  denotes element-wise multiplication and  $s$  is broadcast along the spatial dimensions when applied to  $X$ .

### 3.4. Loss Function

The proposed model was optimized using a compound objective function consisting of an adversarial term and a pixel-wise reconstruction term. Let  $x$  denote the input low-light image,  $y$  the corresponding normal-light target image, and  $\hat{y} = G(x)$  the enhanced image generated by the generator  $G$ . Since a conditional PatchGAN discriminator  $D$  was adopted, the discriminator receives the image pairs  $(x, y)$  for real samples and  $(x, \hat{y})$  for generated samples.

The discriminator loss is mathematically stated as

$$\mathcal{L}_D = \frac{1}{2} [\mathcal{L}_{\text{BCE}}(D(x, y), \mathbf{1}) + \mathcal{L}_{\text{BCE}}(D(x, \hat{y}), \mathbf{0})], \quad (10)$$

where  $\mathcal{L}_{\text{BCE}}$  denotes the binary cross-entropy loss with logits, and  $\mathbf{1}$  and  $\mathbf{0}$  represent tensors of ones and zeros, respectively, with the same dimensions as the discriminator output. A scaling coefficient of  $1/2$  balances the gradient updates between  $G$  and  $D$ , preventing the discriminator from overpowering the generator during the early stages of training.

The adversarial loss for the generator is defined as

$$\mathcal{L}_{\text{adv}} = \mathcal{L}_{\text{BCE}}(D(x, \hat{y}), \mathbf{1}). \quad (11)$$

In addition, the pixel-wise reconstruction loss was defined as the mean absolute error between the enhanced image and the corresponding normal-light reference:

$$\mathcal{L}_{\text{L1}} = \frac{1}{N} \|\hat{y} - y\|_1, \quad (12)$$

where  $\{\hat{y}, y\} \in \mathbb{R}^{B \times C \times H \times W}$  denote the enhanced and reference images, respectively. Here,  $B$  is the batch size,  $C$  is the number of color channels, and  $H$  and  $W$  are the spatial dimensions. Therefore, the normalization factor is explicitly given by  $N = B \times C \times H \times W$ .

For the generator, the overall objective function is defined as

$$\mathcal{L}_G = (1 - \beta_o) \mathcal{L}_{\text{adv}} + \beta_o \mathcal{L}_{\text{L1}}, \quad (13)$$

where the coefficient  $\beta_o = 100/101$  establishes a relative weighting of  $100 : 1$  between the pixel-wise reconstruction term and the adversarial term. This choice places greater emphasis on the pixel-wise reconstruction term  $\mathcal{L}_{\text{L1}}$ . The adversarial term promotes perceptual realism at the patch level [29]. The larger weight reduces the likelihood of visually plausible yet spatially misaligned outputs and adheres to the standard conditional GAN formulation for paired image-to-image translation.

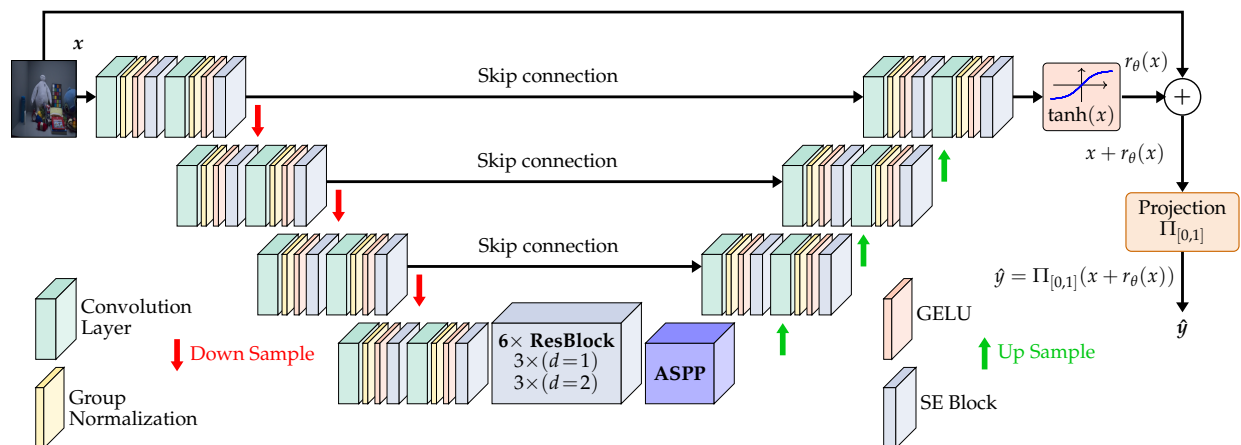
### 3.5. Generator Network

The proposed generator, denoted by  $\mathcal{G}$ , adopts a ResUNet-like encoder–bottleneck–decoder architecture with residual learning for low-light image enhancement. Figure 3 illustrates the overall structure of the proposed generator. Let  $x \in [0, 1]^{H \times W \times 3}$  denote the normalized input low-light image, where  $H$ ,  $W$ , and  $3$  correspond to the image height, width, and number of RGB channels, respectively. During mini-batch processing, the corresponding input tensor follows the channel-first format  $(B, 3, H, W)$ . The generator predicts a residual illumination correction  $r_\theta(x)$  with the same spatial dimensions as  $x$ . The enhanced image, denoted as  $\hat{y} = \mathcal{G}(x)$ , is then obtained as

$$\hat{y} = \Pi_{[0,1]}(x + r_\theta(x)), \quad (14)$$

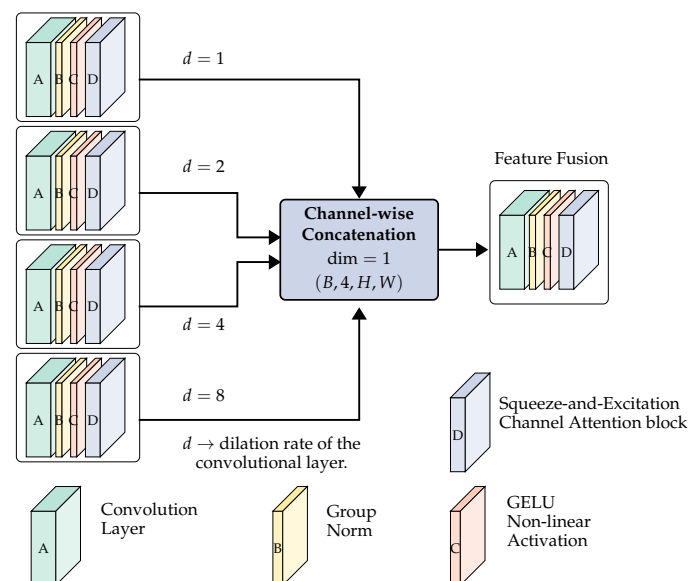
where  $\Pi_{[0,1]}(\cdot)$  denotes an element-wise clipping operation onto the valid intensity range  $[0, 1]$ . Thus, each transformed pixel remains within the normalized image range.

The encoder first extracts shallow features using two convolutional blocks, and then it performs progressive spatial downsampling across three stages. Each block consists of reflection padding, convolution, GN, GELU activation, and a Squeeze-and-Excitation (SE) module. The bottleneck residual blocks omit the SE module and focus on residual feature refinement through dilated convolutions.



**Figure 3.** Proposed ResGASP-GAN based on a ResUNet-like encoder–decoder architecture. It is based on an encoder–decoder design with ASPP at the residual bottleneck, which includes six sequential residual blocks followed by the ASPP module. Convolution (feature extraction), Group Normalization (32 channel groups), GELU activation (nonlinear representation), and Squeeze-and-Excitation (SE) channel attention blocks are used throughout the network. At the same time, skip connections transfer encoder features to the corresponding decoder stages.

At the bottleneck, the deepest feature representation is refined by a sequence of residual blocks with dilated convolutions, followed by an Atrous Spatial Pyramid Pooling (ASPP) module with dilation rates of (1, 2, 4, 8) to incorporate multi-scale contextual information, as shown in Figure 4.



**Figure 4.** ASPP-based multi-scale contextual aggregation module. Parallel dilated convolutional branches ( $d = \{1, 2, 4, 8\}$ ) extract features with different receptive fields, which are concatenated along the channel dimension and integrated through a subsequent feature-fusion block.

The UNet-like encoder–decoder already provides multi-scale representations; however, ASPP explicitly enlarges the receptive field at the deepest feature level without further reducing spatial resolution, which is consistent with LLIE architectures that rely on multi-scale contextual representations [30].

The combination of ASPP and SE is motivated by their complementary roles: ASPP captures multi-scale spatial context, whereas SE recalibrates channel-wise responses to emphasize illumination- and color-relevant features. This combination is intended to

alleviate key LLIE challenges, including noise amplification, structural distortion, color distortion, halos, and artifacts [31].

More specifically, let  $z_{\ell-1} \in \mathbb{R}^{B \times C_b \times H_b \times W_b}$  denote the input feature map to the  $\ell$ -th residual block, where  $C_b$  denotes the bottleneck channel dimension, and  $H_b = H/8$  and  $W_b = W/8$  correspond to the spatial dimensions of the deepest encoder representation. Each residual block preserves the spatial resolution and channel dimensionality by using stride-one convolutions with reflection padding equal to the dilation rate. The output of the  $\ell$ -th residual block is defined as

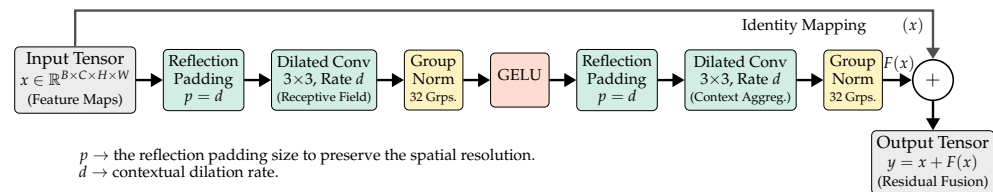
$$z_{\ell} = z_{\ell-1} + \mathcal{F}_{d_{\ell}}(z_{\ell-1}), \quad (15)$$

where  $\mathcal{F}_{d_{\ell}}(\cdot)$  denotes the residual transformation and  $d_{\ell}$  is the dilation rate of the corresponding block. The residual transformation is implemented using two consecutive  $3 \times 3$  dilated convolutional units as follows:

$$\mathcal{F}_{d_{\ell}}(z) = \text{GN}_2 \left( \text{Conv}_{3 \times 3}^{d_{\ell}} \left( \text{RP}_{d_{\ell}} \left( \text{GELU} \left( \text{GN}_1 \left( \text{Conv}_{3 \times 3}^{d_{\ell}} \left( \text{RP}_{d_{\ell}}(z) \right) \right) \right) \right) \right) \right), \quad (16)$$

where  $\text{RP}_{d_{\ell}}(\cdot)$  denotes reflection padding with a padding size of  $d_{\ell}$ ,  $\text{Conv}_{3 \times 3}^{d_{\ell}}(\cdot)$  is a  $3 \times 3$  convolution with a dilation rate of  $d_{\ell}$ , and  $\text{GN}(\cdot)$  denotes Group Normalization. The first convolutional unit uses GELU activation, whereas the second unit omits the final activation before the identity addition.

Figure 5 illustrates the internal structure of the dilated residual block used in the bottleneck.



**Figure 5.** Structure of the dilated residual block used in the bottleneck of the proposed generator. The block applies two consecutive  $3 \times 3$  dilated convolutional units with reflection padding and Group Normalization. The first unit includes GELU activation, whereas the second unit omits the final activation before adding the identity shortcut. The symbol  $+$  is the element-wise addition.

In the proposed generator,  $n_{\text{res}} = 6$  residual blocks are used at the bottleneck. The first three residual blocks use a dilation rate of  $d_{\ell} = 1$ , whereas the last three use a dilation rate of  $d_{\ell} = 2$ :

$$d_{\ell} = \begin{cases} 1, & \forall \ell = \{1, 2, 3\} \\ 2, & \forall \ell = \{4, 5, 6\}. \end{cases} \quad (17)$$

This configuration first performs local residual refinement, and then it increases the receptive field size without additional downsampling. The resulting representation is subsequently processed by the ASPP module, which aggregates multi-scale contextual information through parallel dilated convolutional branches.

The decoder then reconstructs the enhanced representation via bilinear upsampling and concatenation with the corresponding encoder features, thereby preserving fine spatial details. After feature fusion and refinement, a final  $3 \times 3$  convolution generates a three-channel residual image. The residual output is further constrained through a hyperbolic tangent activation and scaling operation, defined as

$$r_{\theta}(x) = \frac{1}{2} \tanh(f_{\text{out}}(u_0(x))), \quad (18)$$

where  $u_0$  denotes the last decoder feature map and  $f_{\text{out}}(\cdot)$  corresponds to the output projection layer. Here, the residual connection is bound to  $[-0.5, 0.5]$  because the generator performs residual enhancement on images normalized to  $[0, 1]$ . This interval allows the generator to increase or decrease each pixel intensity by up to half of the normalized dynamic range before clipping, which is sufficient for substantial illumination correction while avoiding excessively large residual updates and preserving numerical stability. In contrast, using the full  $\tanh(x)$  range  $\in [-1, 1]$  would allow the residual branch to shift a pixel across the entire intensity range, increasing the probability of clipping, saturation, color shifts, and over-enhancement. The factor  $1/2$  imposes a fixed, symmetric residual constraint that promotes controlled enhancement without introducing extra adaptive or learnable scaling parameters.

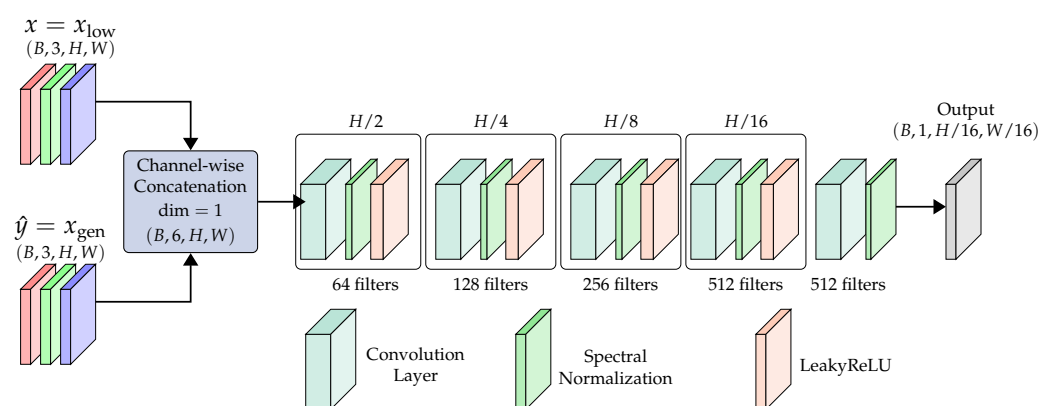
This design enables the generator to improve image illumination while preserving structural details and color fidelity. Group Normalization stabilizes optimization during small-batch training, whereas SE attention and ASPP enhance channel-wise feature recalibration and multi-scale context aggregation, respectively.

### 3.6. Discriminator Network

The discriminator, denoted by  $\mathcal{D}$ , is formulated as a conditional PatchGAN that evaluates the realism of the enhanced image relative to the corresponding low-light input. Formally, the discriminator receives the channel-wise concatenation of the low-light image  $x$  and the generated image  $\hat{y}$ , denoted as  $x_{\text{gen}}$  in Figure 6, yielding a six-channel tensor of size  $(B, 6, H, W)$ . Figure 6 shows the conditional discriminator pathway, where the low-light input  $x_{\text{low}} \in \mathbb{R}^{B \times 3 \times H \times W}$  and the generated image  $x_{\text{gen}} \in \mathbb{R}^{B \times 3 \times H \times W}$  are concatenated along the channel dimension to form an input tensor of size  $\mathbb{R}^{B \times 6 \times H \times W}$ . This conditional formulation is expressed in terms of the discriminator as

$$\mathcal{D}(x, y) \quad \text{or} \quad \mathcal{D}(x, \hat{y}), \quad (19)$$

where the objective is to determine whether the image pair represents a real or a generated sample. Although Figure 6 illustrates the generated-pair branch, the same conditional structure is used for real pairs by replacing the generated image  $\hat{y}$  with the ground-truth normal-light image  $y$ .



**Figure 6.** Conditional PatchGAN discriminator architecture with channel-wise input concatenation, spectrally normalized convolutional layers, and patch-wise logit prediction.

Architecturally, the discriminator consists of a sequence of convolutional blocks inspired by the PatchGAN architecture. Each block applies a spectrally normalized convolution followed by a LeakyReLU activation with a negative slope of 0.2. As illustrated in Figure 6, the discriminator progressively reduces the spatial resolution of the

concatenated input from  $(H, W)$  to  $\left(\frac{H}{2}, \frac{W}{2}\right)$ ,  $\left(\frac{H}{4}, \frac{W}{4}\right)$ ,  $\left(\frac{H}{8}, \frac{W}{8}\right)$ , and finally  $\left(\frac{H}{16}, \frac{W}{16}\right)$ . The number of filters increases from 64 to 128, 256, and 512, thereby producing a hierarchical feature representation before the final patch-wise prediction map. Spectral normalization is applied to all convolutional layers to improve training stability and constrain the Lipschitz constant of the discriminator. This hierarchical design enables the extraction of local representations at multiple spatial scales.

Unlike a global discriminator that produces a single scalar output, the adopted PatchGAN generates a spatial map of logits, where each element assesses the realism of a local image patch. The discriminator emphasizes high-frequency consistency and local structural fidelity, which are particularly important in low-light image enhancement, where noise, texture degradation, and illumination artifacts often appear in localized regions.

The discriminator can be written as

$$M_{\text{real}} = \mathcal{D}(x, y), \quad M_{\text{fake}} = \mathcal{D}(x, \hat{y}), \quad (20)$$

where  $M_{\text{real}}$  and  $M_{\text{fake}}$  denote the real and fake logit maps, respectively. Since the discriminator outputs logits rather than probabilities, no sigmoid activation is applied at the final layer. Hence, the output tensor has size  $\mathbb{R}^{B \times 1 \times \frac{H}{16} \times \frac{W}{16}}$ , where each spatial element corresponds to the discriminator's response for a local receptive field.

The conditional PatchGAN design allows the discriminator to assess not only the perceptual realism of the enhanced image but also its consistency with the corresponding low-light input. Spectral normalization further stabilizes the adversarial optimization process.

## 4. Experimental Results

The experimental evaluation employed a workstation equipped with an NVIDIA RTX 4070 Ti GPU (12 GB of GDDR6X VRAM) and an Intel Core i7 processor. The implementation used Python (version 3.11.15) and the PyTorch framework (version 2.5.1). Data augmentation was not applied, so as to prevent further variability. The model was trained for 500 epochs with a mini-batch size of 2. All computations were performed in single-precision (32-bit) floating-point (FP32).

The checkpoint yielding the lowest total training loss was retained throughout the optimization process. Early stopping was employed to halt training when validation metrics no longer improved, using a patience of 20 epochs. The source code used in this study is publicly available on GitHub [32].

The main training and optimization settings adopted in this study include the optimizer (AdamW), learning rate ( $1.0 \times 10^{-3}$ ), AdamW momentum parameters ( $\beta_1 = 0.500$ ,  $\beta_2 = 0.999$ ), weight decay ( $4.0 \times 10^{-5}$ ), batch size (2), number of epochs (500), and random seed (42).

### 4.1. Datasets

The LOL-v1 dataset comprises 500 image pairs from real-world environments [14]. It provides corresponding low-light and normal-light images. Following the standard test-split protocol, 485 pairs were used for training and 15 for testing. The image resolution of all samples is  $400 \times 600$  pixels. LOL-v2 was subsequently extended to include both real and synthetic low- and normal-light image pairs, capturing diverse illumination conditions and more challenging degradations [33]. LOL-v2 comprises two subsets, LOL-v2-Real and LOL-v2-Synthetic, both of which provide paired low-light and normal-light images [34]. LOL-v2-Real includes 689 training pairs and 100 testing pairs, while LOL-v2-Synthetic includes 900 training pairs and 100 testing pairs. In total, LOL-v2 comprises 1789 paired

samples, corresponding to 3578 individual images when low-light inputs and normal-light references are counted separately.

Table 1 summarizes the datasets and their train/test partitions according to the evaluation protocols commonly adopted in the LLIE literature.

**Table 1.** Summary of the studied datasets, including the number of training and testing images used for training and testing in LOL-v1, LOL-v2 Real, and LOL-v2 Synthetic.

| Dataset          | Training | Test |
|------------------|----------|------|
| LOL-v1           | 485      | 15   |
| LOL-v2 Real      | 689      | 100  |
| LOL-v2 Synthetic | 900      | 100  |

Data augmentation was intentionally omitted to preserve the original low-light/normal-light image pairs (i.e., the original data distribution), thereby avoiding augmentation-dependent bias and promoting fair comparisons across the assessed datasets.

#### 4.2. Evaluation Metrics

This section presents the evaluation metrics used to evaluate the proposed framework. The evaluation metrics are categorized as full-reference and no-reference, depending on whether they involve a reference image.

##### 4.2.1. Peak Signal-to-Noise Ratio

Peak Signal-to-Noise Ratio (PSNR) is a standard metric for comparing enhanced images to clear reference images. It measures pixel-wise error, with higher PSNR indicating greater fidelity to the original (ground truth) [20].

$$\text{PSNR} = 10 \log_{10} \left( \frac{I_{\max}^2}{\text{MSE}} \right), \quad (21)$$

where  $I_{\max}$  denotes the maximum possible pixel value of the image, which ranges from  $[0, 255]$  for 8-bit images. Here, MSE denotes the Mean Squared Error:

$$\text{MSE} = \frac{1}{W \times H} \sum_{i=1}^W \sum_{j=1}^H [I(i, j) - \hat{I}(i, j)]^2, \quad (22)$$

where  $I$  represents the reference image (normal-light),  $\hat{I}$  is the enhanced image, and  $W \times H$  are the image dimensions.

##### 4.2.2. Structural Similarity Index Measure

The Structural Similarity Index Measure (SSIM) assesses image quality based on luminance, contrast, and structure, thereby aligning more closely with human visual perception [35].

$$\text{SSIM}(I, \hat{I}) = \frac{2\mu_I\mu_{\hat{I}} + C_1}{\mu_I^2 + \mu_{\hat{I}}^2 + C_1} \cdot \frac{2\sigma_{I\hat{I}} + C_2}{\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2}, \quad (23)$$

where  $\mu_I$  and  $\mu_{\hat{I}}$  are the local mean intensity values. Here,  $\sigma_I^2$  and  $\sigma_{\hat{I}}^2$  are the corresponding local intensity variances, and  $\sigma_{I\hat{I}}$  represents the local cross-covariance between  $I$  and  $\hat{I}$ .

##### 4.2.3. Learned Perceptual Image Patch Similarity

Learned Perceptual Image Patch Similarity (LPIPS) compares the deep feature maps extracted from a pretrained backbone [14]. Let  $F_l(\cdot)$  be the mapping implemented by layer  $l$

of the pretrained network. For an input image  $X$ , the corresponding activation (feature map) at layer  $l$  is  $\mathbf{y}^l(X) = F_l(X)$ , whose spatial dimensions are  $H_l \times W_l$ . To reduce sensitivity to feature magnitude, the channel vector at each spatial location  $(h, w)$  is  $\sigma(W)$ -normalized across channels as

$$\hat{\mathbf{y}}^l(X)_{hw} = \frac{\mathbf{y}^l(X)_{hw}}{\|\mathbf{y}^l(X)_{hw}\|_2}, \quad (24)$$

where  $\mathbf{y}^l(X)_{hw}$  and  $\hat{\mathbf{y}}^l(X)_{hw}$  denote the un-normalized and normalized feature vectors at location  $(h, w)$ , respectively. LPIPS then aggregates, over  $L$  selected layers, the spatially averaged weighted squared distance  $\|\cdot\|_2^2$  between the normalized features of  $I$  and  $\hat{I}$ :

$$\text{LPIPS}(I, \hat{I}) = \sum_{l=1}^L \frac{1}{H_l \times W_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} \left\| \mathbf{w}_l \odot \left( \hat{\mathbf{y}}^l(I)_{hw} - \hat{\mathbf{y}}^l(\hat{I})_{hw} \right) \right\|_2^2, \quad (25)$$

where  $H_l$  and  $W_l$  are the height and width of the layer  $l$  feature map,  $\odot$  denotes the element-wise product, and  $\mathbf{w}_l$  is a vector of per-channel weights for layer  $l$  (learned in the calibrated LPIPS variant). Hence, lower LPIPS values indicate higher perceptual similarity between the enhanced image  $\hat{I}$  and the reference image  $I$ .

#### 4.2.4. Natural Image Quality Evaluator

The Natural Image Quality Evaluator (NIQE) is a no-reference metric that assesses perceptual quality by measuring the extent to which an image deviates from the natural scene statistics (NSS) learned from pristine images [36]. NIQE is computed only on the enhanced image  $\hat{I}$  and is independent of the ground-truth image  $I$ . Lower values indicate better perceptual quality.

$$\text{NIQE}(\hat{I}) = \sqrt{(\mathbf{v}_{\text{nat}} - \mathbf{v}_{\hat{I}})^\top \left( \frac{\boldsymbol{\Sigma}_{\text{nat}} + \boldsymbol{\Sigma}_{\hat{I}}}{2} \right)^{-1} (\mathbf{v}_{\text{nat}} - \mathbf{v}_{\hat{I}})}, \quad (26)$$

where  $\mathbf{v}_{\text{nat}}$  and  $\boldsymbol{\Sigma}_{\text{nat}}$  denote the mean vector and covariance matrix of the Multivariate Gaussian (MVG) model fitted on NSS features extracted from pristine natural images. In contrast,  $\mathbf{v}_{\hat{I}}$  and  $\boldsymbol{\Sigma}_{\hat{I}}$  are the corresponding MVG parameters estimated from the NSS features of  $\hat{I}$ .

#### 4.2.5. Blind/Referenceless Image Spatial Quality Evaluator

BRISQUE is a no-reference image quality assessment metric that operates in the spatial domain by modeling deviations from natural scene statistics. In particular, the method relies on Mean-Subtracted Contrast-Normalized (MSCN) coefficients and the statistical dependencies between neighboring coefficients [37].

Given a luminance image  $I(i, j)$ , the MSCN coefficients are defined as

$$\hat{I}(i, j) = \frac{I(i, j) - \mu_{ij}}{\sqrt{\sigma_{ij}^2 + \epsilon}}, \quad (27)$$

where  $\epsilon$  is a small positive constant introduced for numerical stability. The local mean  $\mu_{ij}$  and local variance  $\sigma_{ij}^2$  are computed by

$$\mu_{ij} = \sum_{k=-K}^K \sum_{l=-L}^L w_{k,l} I(i-k, j-l), \quad (28)$$

$$\sigma_{ij}^2 = \sum_{k=-K}^K \sum_{l=-L}^L w_{k,l} [I(i-k, j-l) - \mu_{ij}]^2, \quad (29)$$

where  $w_{k,l}$  denotes a normalized Gaussian weighting window.

BRISQUE captures local spatial correlations through pairwise products of adjacent MSCN coefficients at a one-pixel distance along four orientations: horizontal, vertical, main diagonal, and anti-diagonal:

$$\begin{pmatrix} H(i,j) \\ V(i,j) \\ D_1(i,j) \\ D_2(i,j) \end{pmatrix} = \hat{I}(i,j) \begin{pmatrix} \hat{I}(i,j+1) \\ \hat{I}(i+1,j) \\ \hat{I}(i+1,j+1) \\ \hat{I}(i+1,j-1) \end{pmatrix}. \quad (30)$$

The empirical distributions of these pairwise products are modeled using an Asymmetric Generalized Gaussian Distribution (AGGD), which is defined as

$$g(x) = \frac{\alpha}{(\beta_l + \beta_r)\Gamma(1/\alpha)} \begin{cases} \exp\left\{-\left(-\frac{x}{\beta_l}\right)^\alpha\right\}, & x < 0, \\ \exp\left\{-\left(+\frac{x}{\beta_r}\right)^\alpha\right\}, & x \geq 0, \end{cases} \quad (31)$$

where  $\beta_l = \sigma_l \sqrt{\frac{\Gamma(1/\alpha)}{\Gamma(3/\alpha)}}$  and  $\beta_r = \sigma_r \sqrt{\frac{\Gamma(1/\alpha)}{\Gamma(3/\alpha)}}$ . Here,  $\alpha > 0$  denotes the shape parameter, whereas  $\sigma_l$  and  $\sigma_r$  represent the left and right scale parameters, respectively. The quantities  $\beta_l$  and  $\beta_r$  are auxiliary variables derived from the corresponding scale parameters, whereas  $\Gamma(\cdot)$  denotes the gamma function.

#### 4.3. Numerical Results

Table 2 summarizes quantitative comparisons between the proposed model and recent low-light image enhancement methods on the LOL-v1, LOL-v2-Real, and LOL-v2-Synthetic datasets, using PSNR, SSIM, and LPIPS. Although these comparisons provide an overview of the current literature, they should not be interpreted as a controlled benchmark under identical experimental conditions, as the source code, pretrained models, and computational environments were unavailable.

**Table 2.** Comparison between the proposed model and representative state-of-the-art low-light image enhancement methods on the LOL-v1, LOL-v2-Real, and LOL-v2-Synthetic datasets in terms of PSNR, SSIM, and LPIPS. For this study, the mean values are reported.

| Methods             | LOL-v1       |              |              | LOL-v2-Real  |               |              | LOL-v2-Synthetic |               |               |
|---------------------|--------------|--------------|--------------|--------------|---------------|--------------|------------------|---------------|---------------|
|                     | PSNR↑        | SSIM↑        | LPIPS↓       | PSNR↑        | SSIM↑         | LPIPS↓       | PSNR↑            | SSIM↑         | LPIPS↓        |
| Shin and Lee [38]   | 25.14        | 0.860        | 0.109        | <b>29.14</b> | <b>0.907</b>  | <b>0.085</b> | —                | —             | —             |
| Song [20]           | 22.66        | <b>0.882</b> | 0.149        | 20.23        | 0.844         | <u>0.163</u> | <b>25.99</b>     | <b>0.956</b>  | <b>0.055</b>  |
| Qi et al. [39]      | 23.62        | 0.805        | —            | <u>23.77</u> | 0.827         | —            | —                | —             | —             |
| Yin et al. [19]     | 19.95        | 0.755        | 0.217        | —            | —             | —            | 19.41            | 0.813         | 0.197         |
| Jiang et al. [17]   | 17.48        | 0.652        | 0.322        | —            | —             | —            | 18.64            | 0.677         | 0.309         |
| Wu et al. [14]      | <b>25.77</b> | 0.835        | 0.158        | —            | —             | —            | —                | —             | —             |
| He et al. [18]      | 22.20        | 0.846        | <b>0.084</b> | —            | —             | —            | —                | —             | —             |
| Liu et al. [40]     | 24.74        | <u>0.875</u> | <u>0.094</u> | —            | —             | —            | —                | —             | —             |
| Qian and Jiang [10] | 23.16        | 0.855        | 0.165        | —            | —             | —            | —                | —             | —             |
| Batziau et al. [41] | 22.65        | 0.783        | —            | —            | —             | —            | —                | —             | —             |
| Jeon and Kim [42]   | 22.47        | 0.825        | 0.156        | —            | —             | —            | —                | —             | —             |
| Cha and Tang [15]   | 19.08        | 0.756        | 0.248        | —            | —             | —            | —                | —             | —             |
| Zainab et al. [16]  | 17.48        | 0.802        | 0.210        | —            | —             | —            | —                | —             | —             |
| Zhao and Shi [22]   | 15.66        | 0.660        | 0.300        | —            | —             | —            | —                | —             | —             |
| Proposed model      | 20.15        | 0.8387       | 0.2158       | 19.62        | <u>0.8446</u> | 0.276        | <u>22.30</u>     | <u>0.9154</u> | <u>0.1022</u> |

**Bold** indicates the best value for each metric; *Underline* is the second-best value; (—) is a missing entry unavailable in the literature. (↑) Higher is better. (↓) Lower is better.

The proposed model exhibits image conditions- and dataset-dependent performance across the three benchmarks. On LOL-v1, the proposed model achieves a PSNR of

$20.15 \pm 4.20$  dB and an SSIM of  $0.8387 \pm 0.0587$ , indicating reliable structural similarity and remaining competitive with several recent methods while outperforming multiple GAN-based approaches. Although an LPIPS of  $0.2158 \pm 0.0642$  does not match the best reported performance [18], the results suggest that the proposed model preserves relevant structural information while maintaining moderate reconstruction quality and visual perceptual accuracy, as indicated by the complementary SSIM metric.

On LOL-v2-Real, the proposed model achieves a PSNR of  $19.62 \pm 3.46$  dB, an SSIM of  $0.8446 \pm 0.0816$ , and an LPIPS of  $0.2760 \pm 0.1061$ . Although the PSNR and LPIPS values remain below reported reconstruction-oriented methods, the SSIM is the second-highest among the reported results, indicating superior preservation of structural similarity despite a more conservative brightness reconstruction on real-world low-light images.

On the LOL-v2-Synthetic dataset, the proposed model exhibits favorable performance, obtaining a PSNR of  $22.30 \pm 3.82$ , an SSIM of  $0.9154 \pm 0.0503$ , and an LPIPS of  $0.1022 \pm 0.0478$ . These results outperform those reported by Jiang et al. [17], Yin et al. [19] across all three metrics and correspond to the second-best PSNR, SSIM, and LPIPS values among the analyzed methods, slightly behind those of Song [20].

Furthermore, the SSIM of  $0.8446 \pm 0.0816$  obtained on LOL-v2-Real indicates competitive structural similarity, whereas the consistently second-best results on LOL-v2-Synthetic suggest reliable performance under synthetic low-light conditions, with a high-quality perceptual reconstruction and structural similarity.

The no-reference perceptual quality metrics NIQE and BRISQUE provide complementary information about the visual perceptual quality of low-light enhanced images. Whereas BRISQUE analyzes whether local luminance statistics deviate from those expected in pristine natural images, NIQE assesses whether natural scene statistics deviate from those expected in pristine natural images, potentially indicating reduced perceptual quality.

Table 3 presents representative results for the NIQE and BRISQUE metrics. The proposed model achieved an NIQE score of  $4.4396 \pm 0.8180$ , which is near the best NIQE values [15,18]. By contrast, the proposed model achieved the lowest (best) BRISQUE score among the revised methods on the LOL-v1 dataset, with a value of  $10.4540 \pm 9.0297$ . These results suggest that the proposed framework presents well-balanced no-reference perceptual quality and low visual degradation in enhanced images.

**Table 3.** Comparison of no-reference perceptual quality metrics between the proposed model and representative low-light image enhancement methods across the LOL-v1, LOL-v2-Real, and LOL-v2-Synthetic datasets. Lower NIQE ( $\downarrow$ ) and BRISQUE ( $\downarrow$ ) values indicate improved perceptual quality.

| Methods           | LOL-v1            |                      | LOL-v2-Real       |                      | LOL-v2-Synthetic  |                      |
|-------------------|-------------------|----------------------|-------------------|----------------------|-------------------|----------------------|
|                   | NIQE $\downarrow$ | BRISQUE $\downarrow$ | NIQE $\downarrow$ | BRISQUE $\downarrow$ | NIQE $\downarrow$ | BRISQUE $\downarrow$ |
| Cha and Tang [15] | <b>3.882</b>      | —                    | —                 | —                    | —                 | —                    |
| He et al. [18]    | <u>3.906</u>      | —                    | —                 | —                    | —                 | —                    |
| Du et al. [8]     | 4.239             | 21.849               | —                 | —                    | —                 | —                    |
| Ren et al. [43]   | 7.340             | <u>14.880</u>        | —                 | —                    | —                 | —                    |
| Zhao and Shi [22] | 6.260             | —                    | —                 | —                    | —                 | —                    |
| Proposed model    | 4.4396            | <b>10.4540</b>       | 4.0308            | 16.7976              | 4.4361            | 17.1937              |

**Bold** indicates the best value for each metric; *Underline* is the second-best value; (—) is a missing entry unavailable in the literature. ( $\downarrow$ ) Lower is better.

On LOL-v2-Real, the proposed model yielded NIQE and BRISQUE scores of  $4.0308 \pm 0.8845$  and  $16.7976 \pm 9.4646$ , respectively. Meanwhile, on the LOL-v2 Synthetic dataset, it obtained NIQE and BRISQUE scores of  $4.4361 \pm 1.1281$  and  $17.1937 \pm 14.1628$ , respectively. Since comparable values from previous studies were unavailable for these

datasets, these results should be interpreted as reference values for the perceptual performance of the proposed model rather than as direct state-of-the-art comparisons.

Table 4 summarizes the efficiency comparison of the proposed model, which requires 171 GFLOPs and 26.3 M parameters, indicating high computational complexity and model capacity relative to most representative low-light image enhancement methods. Despite the higher computational complexity, the model achieves an inference time of 0.0366 s per image, which remains competitive with several existing approaches.

**Table 4.** Comparison of computational complexity and inference efficiency for representative low-light image enhancement methods, including floating-point operations, model size, inference time, hardware, and image size.

| Methods            | FLOPs [G] | Parameters [M] | Inference [s] per Image | Device NVIDIA      | Image Size |
|--------------------|-----------|----------------|-------------------------|--------------------|------------|
| Jeon and Kim [42]  | –         | 0.131          | –                       | A6000              | 256 × 256  |
| Zainab et al. [16] | 26.33     | 0.362          | 0.4900                  | GTX 1080Ti         | –          |
| Yin et al. [19]    | 22.70     | 0.450          | –                       | RTX 4090 (24 GB)   | 224 × 224  |
| Song [20]          | 15.60     | 1.610          | 0.3400                  | A6000              | 600 × 400  |
| He et al. [18]     | 130.25    | 18.34          | 0.0867                  | 3090Ti             | 128 × 128  |
| Qi et al. [39]     | 2.51      | 0.140          | 0.0040                  | RTX 4090 (2×)      | –          |
| Shin and Lee [38]  | 25.74     | 25.70          | 0.0500                  | GTX 3090           | 128 × 128  |
| Proposed model     | 171.00    | 26.30          | 0.0366                  | RTX 4070Ti (12 GB) | 600 × 400  |

Note: (–) is a missing entry unavailable in the literature.

These results suggest that the proposed architecture maintains satisfactory inference performance despite its comparatively high model capacity and computational complexity. The model exhibits a trade-off among restoration capability, reduced inference times, and practical computational efficiency, particularly in scenarios where fast LLIE quality is prioritized over architectural compactness.

Table 4 shows that although the proposed model has the highest FLOPs and parameter count, its inference time remains comparatively short, partially due to the generator's fully convolutional architecture, whose dense convolutional operations are efficiently parallelized on GPUs, even for larger image dimensions. The high computational cost is associated with the residual UNet, the six bottleneck residual blocks, the multi-branch ASPP module, and the SE modules, rendering a more robust architecture. FLOPs and inference time per image were measured using RGB images of 600 × 400 with a batch size of 2. Here, inference time is not directly proportional to FLOPs, as it is multifactorial and depends on hardware, input resolution, batch size, numerical precision, memory bandwidth, and implementation details.

#### 4.4. Ablation Study

Tables 5–7 present the ablation study to evaluate the influence of the main architectural components and training configurations of the proposed framework, including residual connection (RC), spectral normalization (SN), the SE module, ASPP, the optimizer, and the normalization method.

Several configurations were analyzed by enabling (✓) and disabling (✗) components. The best-performing configuration, consisting of RC, SN, SE, ASPP, AdamW, and GN, is highlighted by a gray row in Tables 5–7. The results show that the effect of the normalization method depends on the dataset and the evaluated metric.

Regarding LOL-v1, the complete GN-based configurations outperform their BN-based counterparts for both optimizers, as shown in Table 5. Specifically, AdamW-GN improves PSNR and SSIM from  $18.94 \pm 3.29$  dB and  $0.8180 \pm 0.0957$  to  $20.15 \pm 4.20$  dB and  $0.8387 \pm 0.0587$ , respectively, compared with AdamW-BN. A similar increase in PSNR is observed with Adam, and GN raises PSNR from  $18.11 \pm 3.57$  to  $20.18 \pm 4.03$  dB.

**Table 5.** Ablation study to evaluate the contributions of the main architectural components and training configurations to LOL-v1. Results are reported as the mean  $\pm$  standard deviation across test images; ( $\checkmark$ ) and ( $\times$ ) indicate enabled and disabled components, respectively.

| Dataset | RC           | SN           | SE           | ASPP         | Optimizer | BN           | GN           | PSNR $\uparrow$  | SSIM $\uparrow$     | LPIPS $\downarrow$  | NIQE $\downarrow$   | BRISQUE $\downarrow$  |
|---------|--------------|--------------|--------------|--------------|-----------|--------------|--------------|------------------|---------------------|---------------------|---------------------|-----------------------|
| LOL-v1  | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 18.94 $\pm$ 3.29 | 0.8180 $\pm$ 0.0957 | 0.2239 $\pm$ 0.0931 | 4.1726 $\pm$ 0.6905 | 12.9899 $\pm$ 8.6817  |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | AdamW     | $\times$     | $\checkmark$ | 20.15 $\pm$ 4.20 | 0.8387 $\pm$ 0.0587 | 0.2158 $\pm$ 0.0642 | 4.4396 $\pm$ 0.8180 | 10.4540 $\pm$ 9.0297  |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\checkmark$ | AdamW     | $\times$     | $\checkmark$ | 19.26 $\pm$ 3.90 | 0.8209 $\pm$ 0.0757 | 0.2474 $\pm$ 0.0705 | 3.7786 $\pm$ 0.6350 | 13.3295 $\pm$ 7.7229  |
|         | $\times$     | $\checkmark$ | $\checkmark$ | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 15.05 $\pm$ 3.13 | 0.7081 $\pm$ 0.0851 | 0.4191 $\pm$ 0.0652 | 5.4888 $\pm$ 0.6994 | 35.9528 $\pm$ 6.6074  |
|         | $\times$     | $\checkmark$ | $\times$     | $\checkmark$ | AdamW     | $\times$     | $\checkmark$ | 16.65 $\pm$ 3.50 | 0.7373 $\pm$ 0.0865 | 0.3612 $\pm$ 0.1058 | 5.4929 $\pm$ 1.7137 | 40.8097 $\pm$ 16.0747 |
|         | $\times$     | $\checkmark$ | $\checkmark$ | $\times$     | AdamW     | $\checkmark$ | $\times$     | 15.76 $\pm$ 2.70 | 0.7047 $\pm$ 0.0937 | 0.4079 $\pm$ 0.1318 | 5.5622 $\pm$ 0.9079 | 25.5791 $\pm$ 9.5856  |
|         | $\times$     | $\checkmark$ | $\times$     | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 14.71 $\pm$ 2.53 | 0.6520 $\pm$ 0.0777 | 0.4997 $\pm$ 0.0604 | 5.6771 $\pm$ 0.5870 | 41.7269 $\pm$ 6.8356  |
|         | $\times$     | $\checkmark$ | $\times$     | $\times$     | AdamW     | $\checkmark$ | $\times$     | 16.04 $\pm$ 3.10 | 0.7202 $\pm$ 0.0632 | 0.3737 $\pm$ 0.0484 | 6.8820 $\pm$ 1.5219 | 35.2467 $\pm$ 22.9616 |
|         | $\times$     | $\checkmark$ | $\times$     | $\times$     | AdamW     | $\times$     | $\checkmark$ | 15.50 $\pm$ 2.70 | 0.6974 $\pm$ 0.0936 | 0.4834 $\pm$ 0.0842 | 6.2876 $\pm$ 0.7642 | 53.0968 $\pm$ 5.2556  |
|         | $\times$     | $\times$     | $\checkmark$ | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 14.55 $\pm$ 3.30 | 0.6227 $\pm$ 0.1894 | 0.4945 $\pm$ 0.1739 | 5.4070 $\pm$ 0.7932 | 56.2630 $\pm$ 6.8136  |
|         | $\times$     | $\times$     | $\times$     | $\checkmark$ | AdamW     | $\times$     | $\checkmark$ | 15.66 $\pm$ 2.82 | 0.7103 $\pm$ 0.0816 | 0.5023 $\pm$ 0.0844 | 7.0904 $\pm$ 0.7824 | 46.9516 $\pm$ 6.6904  |
|         | $\times$     | $\times$     | $\checkmark$ | $\times$     | AdamW     | $\checkmark$ | $\times$     | 16.36 $\pm$ 3.06 | 0.7289 $\pm$ 0.0638 | 0.3854 $\pm$ 0.0639 | 6.9010 $\pm$ 0.6614 | 32.3490 $\pm$ 10.8595 |
|         | $\times$     | $\times$     | $\checkmark$ | $\times$     | AdamW     | $\times$     | $\checkmark$ | 15.17 $\pm$ 3.01 | 0.6964 $\pm$ 0.0672 | 0.5270 $\pm$ 0.0766 | 6.5874 $\pm$ 0.6744 | 47.6592 $\pm$ 4.5203  |
|         | $\times$     | $\times$     | $\times$     | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 15.41 $\pm$ 3.22 | 0.6936 $\pm$ 0.0783 | 0.4481 $\pm$ 0.0761 | 5.1576 $\pm$ 0.7444 | 30.7012 $\pm$ 9.8658  |
|         | $\times$     | $\times$     | $\times$     | $\checkmark$ | AdamW     | $\times$     | $\checkmark$ | 15.97 $\pm$ 3.28 | 0.7295 $\pm$ 0.0694 | 0.4357 $\pm$ 0.0692 | 6.0855 $\pm$ 0.5975 | 35.8556 $\pm$ 5.5409  |
|         | $\times$     | $\times$     | $\times$     | $\times$     | AdamW     | $\checkmark$ | $\times$     | 14.65 $\pm$ 2.06 | 0.6485 $\pm$ 0.0797 | 0.5156 $\pm$ 0.0668 | 6.5169 $\pm$ 0.9581 | 52.6938 $\pm$ 7.7075  |
|         | $\times$     | $\times$     | $\times$     | $\times$     | AdamW     | $\times$     | $\checkmark$ | 15.61 $\pm$ 3.09 | 0.7143 $\pm$ 0.0778 | 0.4577 $\pm$ 0.0785 | 5.4051 $\pm$ 0.5221 | 51.8875 $\pm$ 8.0082  |
|         | $\checkmark$ | $\times$     | $\checkmark$ | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 16.20 $\pm$ 3.89 | 0.7442 $\pm$ 0.0877 | 0.3869 $\pm$ 0.0774 | 5.8535 $\pm$ 0.5980 | 21.7166 $\pm$ 10.2135 |
|         | $\checkmark$ | $\times$     | $\checkmark$ | $\checkmark$ | AdamW     | $\times$     | $\checkmark$ | 19.04 $\pm$ 3.65 | 0.8273 $\pm$ 0.0603 | 0.2500 $\pm$ 0.0826 | 3.6850 $\pm$ 0.6305 | 6.1131 $\pm$ 7.9572   |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\checkmark$ | AdamW     | $\checkmark$ | $\times$     | 18.71 $\pm$ 3.35 | 0.8158 $\pm$ 0.0804 | 0.2480 $\pm$ 0.0702 | 4.0237 $\pm$ 0.7262 | 15.2353 $\pm$ 10.2116 |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\times$     | AdamW     | $\checkmark$ | $\times$     | 17.52 $\pm$ 2.79 | 0.7633 $\pm$ 0.0810 | 0.3366 $\pm$ 0.0713 | 4.6693 $\pm$ 0.6529 | 15.1011 $\pm$ 8.6932  |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\times$     | AdamW     | $\times$     | $\checkmark$ | 19.92 $\pm$ 3.68 | 0.8338 $\pm$ 0.0583 | 0.2276 $\pm$ 0.0677 | 3.7071 $\pm$ 0.6479 | 12.5250 $\pm$ 7.0706  |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$     | AdamW     | $\times$     | $\checkmark$ | 18.73 $\pm$ 3.62 | 0.8170 $\pm$ 0.0733 | 0.2516 $\pm$ 0.0696 | 3.7060 $\pm$ 0.6171 | 12.9450 $\pm$ 7.6221  |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$     | AdamW     | $\checkmark$ | $\times$     | 17.02 $\pm$ 3.22 | 0.7731 $\pm$ 0.1029 | 0.2747 $\pm$ 0.0897 | 4.3492 $\pm$ 0.5820 | 7.9220 $\pm$ 7.2576   |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$     | Adam      | $\checkmark$ | $\times$     | 16.78 $\pm$ 4.17 | 0.7775 $\pm$ 0.1021 | 0.2528 $\pm$ 0.1050 | 4.5455 $\pm$ 0.7614 | 8.5076 $\pm$ 7.3161   |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$     | Adam      | $\times$     | $\checkmark$ | 19.10 $\pm$ 3.95 | 0.8173 $\pm$ 0.0758 | 0.2427 $\pm$ 0.0642 | 3.7047 $\pm$ 0.5948 | 10.2252 $\pm$ 7.9266  |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\checkmark$ | Adam      | $\times$     | $\checkmark$ | 19.71 $\pm$ 3.95 | 0.8326 $\pm$ 0.0707 | 0.2311 $\pm$ 0.0732 | 3.7793 $\pm$ 0.7042 | 12.0360 $\pm$ 7.8742  |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\checkmark$ | Adam      | $\checkmark$ | $\times$     | 18.02 $\pm$ 3.30 | 0.7921 $\pm$ 0.0993 | 0.2864 $\pm$ 0.1202 | 4.1105 $\pm$ 0.7292 | 13.8762 $\pm$ 9.8179  |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\times$     | Adam      | $\checkmark$ | $\times$     | 18.10 $\pm$ 3.75 | 0.8116 $\pm$ 0.0741 | 0.2687 $\pm$ 0.0660 | 4.1697 $\pm$ 0.9595 | 16.3517 $\pm$ 11.2824 |
|         | $\checkmark$ | $\checkmark$ | $\times$     | $\times$     | Adam      | $\times$     | $\checkmark$ | 18.85 $\pm$ 3.42 | 0.8156 $\pm$ 0.0639 | 0.2758 $\pm$ 0.0681 | 4.0178 $\pm$ 0.5480 | 10.0531 $\pm$ 8.1672  |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | Adam      | $\times$     | $\checkmark$ | 20.18 $\pm$ 4.03 | 0.8373 $\pm$ 0.0548 | 0.2163 $\pm$ 0.0733 | 4.5111 $\pm$ 0.8638 | 5.8779 $\pm$ 9.5455   |
|         | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | Adam      | $\checkmark$ | $\times$     | 18.11 $\pm$ 3.57 | 0.8065 $\pm$ 0.0889 | 0.2355 $\pm$ 0.0864 | 4.2658 $\pm$ 0.5735 | 11.6651 $\pm$ 9.5456  |
|         | $\checkmark$ | $\times$     | $\checkmark$ | $\checkmark$ | Adam      | $\checkmark$ | $\times$     | 17.66 $\pm$ 3.56 | 0.7790 $\pm$ 0.0861 | 0.3353 $\pm$ 0.1110 | 5.1755 $\pm$ 0.5600 | 19.2058 $\pm$ 9.2224  |
|         | $\checkmark$ | $\times$     | $\checkmark$ | $\checkmark$ | Adam      | $\times$     | $\checkmark$ | 19.53 $\pm$ 3.44 | 0.8246 $\pm$ 0.0734 | 0.2536 $\pm$ 0.0739 | 3.9827 $\pm$ 0.7300 | 9.0318 $\pm$ 8.0123   |
|         | $\times$     | $\checkmark$ | $\checkmark$ | $\checkmark$ | Adam      | $\checkmark$ | $\times$     | 16.14 $\pm$ 3.11 | 0.7146 $\pm$ 0.0849 | 0.3777 $\pm$ 0.0830 | 4.8331 $\pm$ 0.8883 | 29.0177 $\pm$ 8.8590  |
|         | $\times$     | $\checkmark$ | $\checkmark$ | $\checkmark$ | Adam      | $\times$     | $\checkmark$ | 16.25 $\pm$ 3.07 | 0.7075 $\pm$ 0.0758 | 0.4027 $\pm$ 0.0756 | 4.9854 $\pm$ 0.8422 | 24.8139 $\pm$ 8.0842  |
|         | $\times$     | $\checkmark$ | $\checkmark$ | $\times$     | Adam      | $\checkmark$ | $\times$     | 15.91 $\pm$ 3.01 | 0.7189 $\pm$ 0.1065 | 0.3753 $\pm$ 0.0934 | 4.8926 $\pm$ 0.5921 | 25.9936 $\pm$ 12.3055 |
|         | $\times$     | $\checkmark$ | $\checkmark$ | $\times$     | Adam      | $\times$     | $\checkmark$ | 16.37 $\pm$ 3.33 | 0.7511 $\pm$ 0.0638 | 0.3945 $\pm$ 0.0634 | 5.8193 $\pm$ 0.7199 | 37.5545 $\pm$ 10.8299 |

( $\uparrow$ ) Higher is better. ( $\downarrow$ ) Lower is better.

**Table 6.** Ablation study to evaluate the contributions of the main architectural components and training configurations to LOL-v2-Real. Results are reported as the mean  $\pm$  standard deviation across test images; ( $\checkmark$ ) and ( $\times$ ) indicate enabled and disabled components, respectively.

| Dataset     | RC | SN | SE | ASPP  | Optimizer | BN | GN           | PSNR↑           | SSIM↑           | LPIPS↓          | NIQE↓             | BRISQUE↓          |
|-------------|----|----|----|-------|-----------|----|--------------|-----------------|-----------------|-----------------|-------------------|-------------------|
| LOL-v2-Real | ✓  | ✓  | ✓  | ✓     | Adam      | ✓  | ✗            | 19.48 ± 2.47    | 0.8079 ± 0.0663 | 0.3744 ± 0.0986 | 5.2660 ± 0.6898   | 30.1917 ± 13.2257 |
|             | ✓  | ✓  | ✓  | ✓     | Adam      | ✗  | ✓            | 19.40 ± 3.78    | 0.8393 ± 0.0595 | 0.2613 ± 0.0913 | 4.0850 ± 0.9060   | 17.3043 ± 10.3199 |
|             | ✓  | ✓  | ✓  | ✗     | Adam      | ✗  | ✓            | 18.95 ± 3.71    | 0.8273 ± 0.0644 | 0.2848 ± 0.0898 | 4.0173 ± 0.7572   | 19.4439 ± 10.2766 |
|             | ✓  | ✓  | ✓  | ✗     | Adam      | ✓  | ✗            | 19.59 ± 3.30    | 0.8340 ± 0.0613 | 0.2910 ± 0.0809 | 4.0872 ± 1.0105   | 19.1183 ± 10.3964 |
|             | ✓  | ✓  | ✗  | ✓     | Adam      | ✓  | ✗            | 18.95 ± 3.01    | 0.7915 ± 0.0699 | 0.3778 ± 0.2740 | 5.1923 ± 0.7240   | 30.3855 ± 11.0321 |
|             | ✓  | ✓  | ✗  | ✓     | Adam      | ✗  | ✓            | 20.13 ± 3.41    | 0.8425 ± 0.0528 | 0.2754 ± 0.1019 | 3.7741 ± 0.7720   | 17.1557 ± 10.5991 |
|             | ✓  | ✓  | ✗  | ✗     | Adam      | ✓  | ✗            | 18.96 ± 3.49    | 0.7951 ± 0.0700 | 0.3837 ± 0.0965 | 5.2684 ± 0.9823   | 26.6922 ± 11.5055 |
|             | ✓  | ✓  | ✗  | ✗     | Adam      | ✗  | ✓            | 20.28 ± 3.91    | 0.8391 ± 0.0685 | 0.2821 ± 0.0998 | 4.0871 ± 0.8259   | 22.7821 ± 9.8895  |
|             | ✓  | ✗  | ✓  | ✓     | Adam      | ✓  | ✗            | 18.73 ± 3.06    | 0.7836 ± 0.0657 | 0.3657 ± 0.0863 | 5.7888 ± 0.8762   | 31.2463 ± 11.6339 |
|             | ✓  | ✗  | ✓  | ✓     | Adam      | ✗  | ✓            | 19.30 ± 3.52    | 0.8346 ± 0.0616 | 0.2671 ± 0.0848 | 3.9044 ± 0.9226   | 16.7702 ± 0.6841  |
|             | ✓  | ✓  | ✓  | ✓     | AdamW     | ✓  | ✗            | 19.64 ± 3.09    | 0.8286 ± 0.0450 | 0.3620 ± 0.0901 | 5.2265 ± 0.7573   | 29.2820 ± 10.7480 |
|             | ✓  | ✓  | ✓  | ✓     | AdamW     | ✗  | ✓            | 19.62 ± 3.46    | 0.8446 ± 0.0816 | 0.2760 ± 0.1061 | 4.0308 ± 0.8845   | 16.7976 ± 9.4646  |
|             | ✓  | ✗  | ✓  | ✓     | AdamW     | ✓  | ✗            | 19.05 ± 3.85    | 0.8332 ± 0.0678 | 0.2626 ± 0.0889 | 4.1617 ± 0.9994   | 17.0845 ± 10.3209 |
|             | ✓  | ✗  | ✓  | ✓     | AdamW     | ✗  | ✓            | 19.58 ± 2.66    | 0.8203 ± 0.0513 | 0.3686 ± 0.1017 | 5.0426 ± 0.5945   | 25.2409 ± 10.1352 |
|             | ✓  | ✓  | ✓  | ✗     | AdamW     | ✗  | ✓            | 19.89 ± 3.56    | 0.8393 ± 0.0624 | 0.2762 ± 0.0975 | 3.8565 ± 0.7791   | 20.0001 ± 7.3945  |
|             | ✓  | ✓  | ✓  | ✗     | AdamW     | ✓  | ✗            | 18.88 ± 3.37    | 0.7910 ± 0.0720 | 0.3666 ± 0.0982 | 5.1488 ± 0.6795   | 23.7434 ± 10.8802 |
|             | ✓  | ✓  | ✗  | ✓     | AdamW     | ✓  | ✗            | 19.12 ± 3.11    | 0.8090 ± 0.0555 | 0.3764 ± 0.0913 | 5.4340 ± 0.8243   | 33.0495 ± 11.8399 |
|             | ✓  | ✓  | ✗  | ✓     | AdamW     | ✗  | ✓            | 19.30 ± 3.37    | 0.8339 ± 0.0556 | 0.2722 ± 0.0887 | 3.8652 ± 0.7655   | 18.7551 ± 8.4204  |
|             | ✓  | ✓  | ✗  | ✗     | AdamW     | ✓  | ✗            | 19.61 ± 4.24    | 0.8280 ± 0.0815 | 0.2941 ± 0.0851 | 4.3999 ± 0.9044   | 18.5977 ± 10.2034 |
| ✓           | ✓  | ✗  | ✗  | AdamW | ✗         | ✓  | 20.09 ± 3.97 | 0.8490 ± 0.0598 | 0.2691 ± 0.0965 | 3.8845 ± 0.7511 | 20.0614 ± 10.7994 |                   |
| ✗           | ✓  | ✓  | ✓  | AdamW | ✓         | ✗  | 17.08 ± 1.94 | 0.7035 ± 0.0823 | 0.5036 ± 0.1045 | 6.4823 ± 0.8977 | 41.8862 ± 10.2765 |                   |

a marked degradation is observed when RC is disabled, supporting the relevance of residual connections on the architecture for this dataset. The results confirm that the normalization modules are very important in improving method performance for perceptual metrics. On LOL-v2-Real, the use of the AdamW optimizer and GN improves the LPIPS, NIQE, and BRISQUE metrics with respect to BN, reducing them from  $0.3620 \pm 0.0901$ ,  $5.2265 \pm 0.7573$ , and  $29.2820 \pm 10.7480$  to  $0.2760 \pm 0.1061$ ,  $4.0308 \pm 0.8845$ , and  $16.7976 \pm 9.4646$ , respectively.

In short, LOL-v2-Real shows a trade-off: AdamW-BN yields only a marginally higher PSNR, whereas AdamW-GN provides higher SSIM and improved LPIPS, NIQE, and BRISQUE values with RC, SN, SE, and ASPP, highlighting the metric-dependent behavior of the evaluated configurations. GN helps achieve more favorable visual-perceptual and no-reference metrics quality despite comparable full-reference performance measures. Moreover, GN provides more stable feature normalization during small-batch training, reducing fluctuations in feature statistics and improving structural reconstruction.

**Table 7.** Ablation study to evaluate the contributions of the main architectural components and training configurations to LOL-v2-Synthetic. Results are reported as the mean  $\pm$  standard deviation across test images; (✓) and (✗) indicate enabled and disabled components, respectively.

| Dataset          | RC | SN | SE | ASPP | Optimizer | BN | GN | PSNR↑            | SSIM↑               | LPIPS↓              | NIQE↓               | BRISQUE↓              |
|------------------|----|----|----|------|-----------|----|----|------------------|---------------------|---------------------|---------------------|-----------------------|
| LOL-v2-Synthetic | ✓  | ✓  | ✓  | ✓    | Adam      | ✓  | ✗  | $19.40 \pm 5.07$ | $0.8532 \pm 0.1070$ | $0.1399 \pm 0.0930$ | $4.2831 \pm 1.1603$ | $15.0514 \pm 13.5775$ |
|                  | ✓  | ✓  | ✓  | ✓    | AdamW     | ✗  | ✓  | $22.30 \pm 3.82$ | $0.9154 \pm 0.0503$ | $0.1022 \pm 0.0478$ | $4.4361 \pm 1.1281$ | $17.1937 \pm 14.1628$ |
|                  | ✓  | ✓  | ✓  | ✗    | AdamW     | ✗  | ✓  | $21.85 \pm 3.76$ | $0.9048 \pm 0.0563$ | $0.1168 \pm 0.0553$ | $4.7278 \pm 1.0794$ | $16.6779 \pm 14.5727$ |
|                  | ✓  | ✗  | ✓  | ✓    | AdamW     | ✓  | ✗  | $20.29 \pm 3.88$ | $0.8640 \pm 0.0867$ | $0.1527 \pm 0.0719$ | $4.3002 \pm 1.0083$ | $12.7551 \pm 13.4996$ |
|                  | ✓  | ✗  | ✓  | ✓    | AdamW     | ✗  | ✓  | $21.94 \pm 3.96$ | $0.8992 \pm 0.0621$ | $0.1225 \pm 0.0582$ | $4.2383 \pm 1.1052$ | $14.7605 \pm 14.0530$ |
|                  | ✓  | ✓  | ✓  | ✓    | AdamW     | ✓  | ✗  | $21.07 \pm 4.97$ | $0.8863 \pm 0.0865$ | $0.1147 \pm 0.0755$ | $4.2188 \pm 1.0629$ | $14.6204 \pm 14.0030$ |
|                  | ✓  | ✓  | ✓  | ✗    | AdamW     | ✓  | ✗  | $24.35 \pm 4.15$ | $0.8732 \pm 0.0889$ | $0.1552 \pm 0.0824$ | $4.5326 \pm 1.0483$ | $14.1528 \pm 13.2919$ |
|                  | ✓  | ✓  | ✗  | ✗    | AdamW     | ✓  | ✗  | $19.90 \pm 3.98$ | $0.8745 \pm 0.0793$ | $0.1493 \pm 0.0728$ | $4.4614 \pm 1.0516$ | $14.6740 \pm 14.8641$ |
|                  | ✓  | ✓  | ✗  | ✗    | AdamW     | ✗  | ✓  | $21.72 \pm 4.18$ | $0.9011 \pm 0.0581$ | $0.1164 \pm 0.0587$ | $4.3977 \pm 1.0928$ | $15.4346 \pm 14.6372$ |
|                  | ✓  | ✓  | ✗  | ✗    | AdamW     | ✓  | ✗  | $19.62 \pm 4.64$ | $0.8640 \pm 0.0875$ | $0.1445 \pm 0.0794$ | $4.3346 \pm 1.0984$ | $12.9048 \pm 14.6048$ |
|                  | ✓  | ✓  | ✗  | ✓    | AdamW     | ✗  | ✓  | $22.41 \pm 4.18$ | $0.9063 \pm 0.0569$ | $0.1142 \pm 0.0520$ | $4.4066 \pm 1.0281$ | $16.8379 \pm 14.2723$ |
|                  | ✓  | ✓  | ✓  | ✓    | Adam      | ✗  | ✓  | $22.42 \pm 4.07$ | $0.9181 \pm 0.0510$ | $0.0989 \pm 0.0482$ | $4.2546 \pm 1.0882$ | $15.1929 \pm 14.5105$ |
|                  | ✓  | ✓  | ✓  | ✗    | Adam      | ✗  | ✓  | $21.26 \pm 4.02$ | $0.9079 \pm 0.0544$ | $0.1126 \pm 0.0481$ | $4.1975 \pm 1.0284$ | $15.9878 \pm 13.7487$ |
|                  | ✓  | ✗  | ✓  | ✓    | Adam      | ✓  | ✗  | $20.76 \pm 4.05$ | $0.8809 \pm 0.0798$ | $0.1397 \pm 0.0716$ | $4.3226 \pm 1.0241$ | $12.6785 \pm 13.7055$ |
|                  | ✓  | ✗  | ✓  | ✓    | Adam      | ✗  | ✓  | $22.44 \pm 4.24$ | $0.9068 \pm 0.0669$ | $0.1145 \pm 0.0605$ | $4.3742 \pm 1.0881$ | $14.8300 \pm 13.7488$ |
|                  | ✓  | ✓  | ✓  | ✗    | Adam      | ✓  | ✗  | $20.77 \pm 4.22$ | $0.8680 \pm 0.1087$ | $0.1446 \pm 0.0912$ | $4.2896 \pm 1.1294$ | $13.8338 \pm 13.5970$ |
|                  | ✓  | ✓  | ✗  | ✓    | Adam      | ✓  | ✗  | $19.79 \pm 4.81$ | $0.8671 \pm 0.0884$ | $0.1450 \pm 0.0799$ | $4.4121 \pm 1.0914$ | $12.2281 \pm 14.1010$ |
|                  | ✓  | ✓  | ✗  | ✓    | Adam      | ✗  | ✓  | $22.24 \pm 3.95$ | $0.9122 \pm 0.0509$ | $0.1082 \pm 0.0502$ | $4.3465 \pm 1.0083$ | $15.4523 \pm 13.4907$ |
|                  | ✓  | ✓  | ✗  | ✗    | Adam      | ✓  | ✗  | $19.95 \pm 4.50$ | $0.8717 \pm 0.0801$ | $0.1371 \pm 0.0700$ | $4.4072 \pm 1.0013$ | $13.6705 \pm 13.7433$ |
|                  | ✓  | ✓  | ✗  | ✗    | Adam      | ✗  | ✓  | $21.53 \pm 4.28$ | $0.9010 \pm 0.0654$ | $0.1009 \pm 0.0531$ | $4.1401 \pm 1.0576$ | $15.0924 \pm 13.6068$ |
|                  | ✗  | ✓  | ✓  | ✓    | Adam      | ✓  | ✗  | $14.72 \pm 3.32$ | $0.7297 \pm 0.0997$ | $0.3411 \pm 0.1114$ | $6.5590 \pm 3.1753$ | $29.6098 \pm 17.7181$ |
|                  | ✗  | ✓  | ✓  | ✓    | Adam      | ✗  | ✓  | $15.02 \pm 3.20$ | $0.7247 \pm 0.0836$ | $0.3584 \pm 0.0992$ | $6.1124 \pm 2.6325$ | $31.2171 \pm 17.2909$ |
|                  | ✗  | ✗  | ✓  | ✓    | Adam      | ✓  | ✗  | $14.19 \pm 2.95$ | $0.6792 \pm 0.1019$ | $0.5331 \pm 0.1091$ | $7.5047 \pm 3.2193$ | $48.8548 \pm 15.7187$ |
|                  | ✗  | ✗  | ✓  | ✓    | Adam      | ✗  | ✓  | $14.20 \pm 2.81$ | $0.6506 \pm 0.0851$ | $0.6088 \pm 0.1000$ | $8.4137 \pm 1.9742$ | $51.8802 \pm 11.2674$ |
|                  | ✗  | ✗  | ✗  | ✓    | Adam      | ✓  | ✗  | $14.14 \pm 2.50$ | $0.6340 \pm 0.1128$ | $0.5030 \pm 0.1023$ | $8.1791 \pm 3.3655$ | $57.4090 \pm 16.0936$ |
|                  | ✗  | ✗  | ✗  | ✓    | Adam      | ✗  | ✓  | $14.33 \pm 2.67$ | $0.6490 \pm 0.0886$ | $0.5811 \pm 0.0979$ | $8.2708 \pm 1.9570$ | $52.0448 \pm 14.1018$ |
|                  | ✗  | ✗  | ✓  | ✗    | Adam      | ✓  | ✗  | $14.47 \pm 3.31$ | $0.7041 \pm 0.1055$ | $0.4796 \pm 0.1079$ | $7.3529 \pm 3.0150$ | $44.4237 \pm 13.1468$ |
|                  | ✗  | ✗  | ✓  | ✗    | Adam      | ✗  | ✓  | $14.28 \pm 3.09$ | $0.6580 \pm 0.0852$ | $0.5830 \pm 0.1008$ | $7.7004 \pm 1.4214$ | $51.0163 \pm 11.1786$ |
|                  | ✗  | ✗  | ✗  | ✗    | Adam      | ✓  | ✗  | $14.16 \pm 3.10$ | $0.6571 \pm 0.1136$ | $0.5409 \pm 0.1177$ | $8.2409 \pm 3.5374$ | $49.4186 \pm 17.6458$ |

(↑) Higher is better. (↓) Lower is better.

For LOL-v2-Synthetic, the complete GN-based configurations also outperform their BN-based counterparts in the full-reference and perceptual metrics. When RC, SN, SE, and ASPP are all enabled, the AdamW-GN configuration achieves  $22.30 \pm 3.82$  dB in PSNR,  $0.9154 \pm 0.0503$  in SSIM, and  $0.1022 \pm 0.0478$  in LPIPS. In contrast, the corresponding AdamW-BN configuration obtains  $21.07 \pm 4.97$  dB,  $0.8863 \pm 0.0865$ , and  $0.1147 \pm 0.0755$ , respectively.

The same trend is observed with Adam, where the complete Adam-GN configuration reaches  $22.42 \pm 4.07$  dB,  $0.9181 \pm 0.0510$  SSIM, and  $0.0989 \pm 0.0482$  LPIPS, compared with  $19.40 \pm 5.07$  dB,  $0.8532 \pm 0.1070$  SSIM, and  $0.1399 \pm 0.0930$  LPIPS for the complete Adam-BN configuration. Notably, the AdamW-BN variant without ASPP achieves the highest PSNR on the LOL-v2-Synthetic dataset, with  $24.35 \pm 4.15$  dB. Regardless, this configuration yields lower SSIM and higher LPIPS than the complete GN-based

frameworks, demonstrating that maximizing pixel-level fidelity does not guarantee optimal structural and perceptual performance.

Overall, the ablation study supports selecting AdamW-GN as a balanced configuration for the proposed model. Although this setting does not achieve the best value for every individual metric, it provides a favorable compromise among fidelity, structural consistency, perceptual similarity, and no-reference image quality across the evaluated datasets.

The comparison between BN and GN shows that the effect of the normalization strategy is dataset- and metric-dependent. By using small-batch training, GN exhibits more stable and consistent behavior across several configurations. For example, on LOL-v1 with AdamW, replacing BN with GN improves PSNR from  $18.94 \pm 3.29$  dB to  $20.15 \pm 4.20$  dB and SSIM from  $0.8180 \pm 0.0957$  to  $0.8387 \pm 0.0587$ .

The ablation results also indicate that the contributions of the RC, SN, SE, and ASPP modules should be interpreted jointly rather than independently. In particular, configurations with RC enabled generally preserve the image structure more effectively during illumination correction. At the same time, SE and ASPP contribute to channel-wise feature recalibration and multi-scale contextual aggregation. However, the highest PSNR does not always correspond to the best perceptual behavior. For instance, on LOL-v2-Synthetic, an AdamW-BN configuration without ASPP achieves the highest PSNR, but it reports lower SSIM and higher LPIPS than the complete GN-based configurations.

The final configuration was selected based on multiple metrics rather than solely on the maximum PSNR value. Overall, AdamW-GN with RC, SN, SE, and ASPP provides a balanced trade-off among reconstruction fidelity, structural preservation, perceptual similarity, and no-reference quality. In contrast, isolated configurations may improve a single metric at the expense of other image-quality criteria.

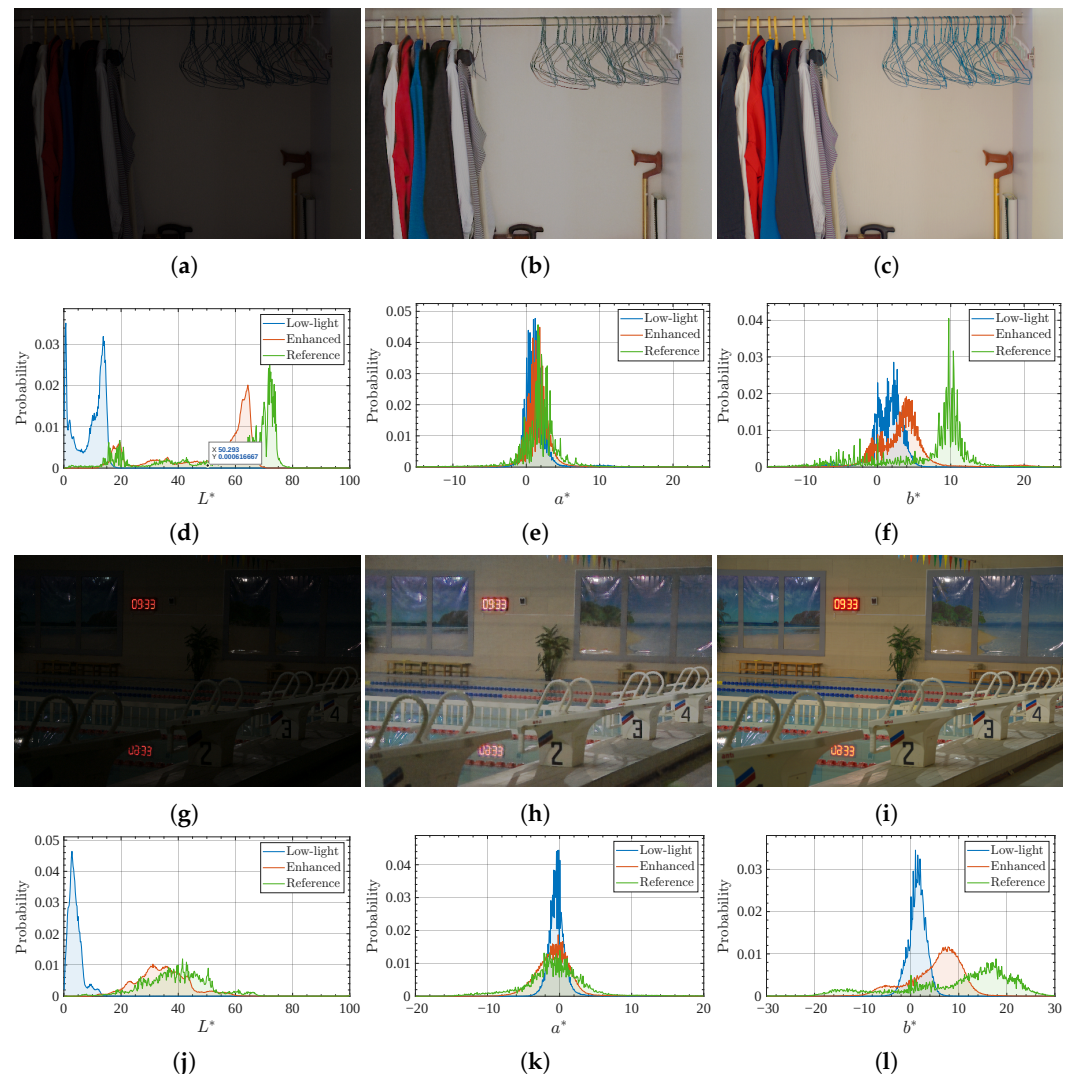
#### 4.5. Qualitative Results

Figure 7 presents qualitative enhancement results and normalized CIE-LAB histogram distributions for two representative samples from the LOL-v1 dataset.

Figure 7a,g show the original low-light images, which suffer from severe underexposure, where most structural details are barely visible. After enhancement, Figure 7b,h present a clear recovery of brightness and contrast, making objects such as clothes, hangers, pool edges, chairs, and background elements more distinguishable. Compared with the ground-truth images in Figure 7c,i, the enhanced results preserve the main scene structure and recover relevant visual details without evident overexposure. Particularly, Figure 7d,j present the luminance  $L^*$  histograms, whereas Figure 7e,k show the  $a^*$  channel histograms and Figure 7f,l present the  $b^*$  channel histograms.

Luminance distributions ( $L^*$ ) indicate that low-light pixels are concentrated toward low luminance values. In contrast, the enhanced images shift these values toward the ground-truth distributions, suggesting improved luminance recovery. The chromatic  $a^*$  and  $b^*$  histograms provide complementary evidence about chromatic coherence and consistency. The near-overlap between the enhanced and ground-truth distributions suggests that the proposed model highly preserves the chromatic balance along the green–red and blue–yellow axes.

Overall, the visual comparisons and CIE-LAB histograms indicate that the proposed model improves perceptual visibility, enhances luminance, and maintains a color distribution consistent with the corresponding ground-truth images. Exact recovery of the original scene color perception is algorithmically feasible from single low-light observations because image acquisition in low-light conditions generally results in the loss of rich original gamma color dynamics.



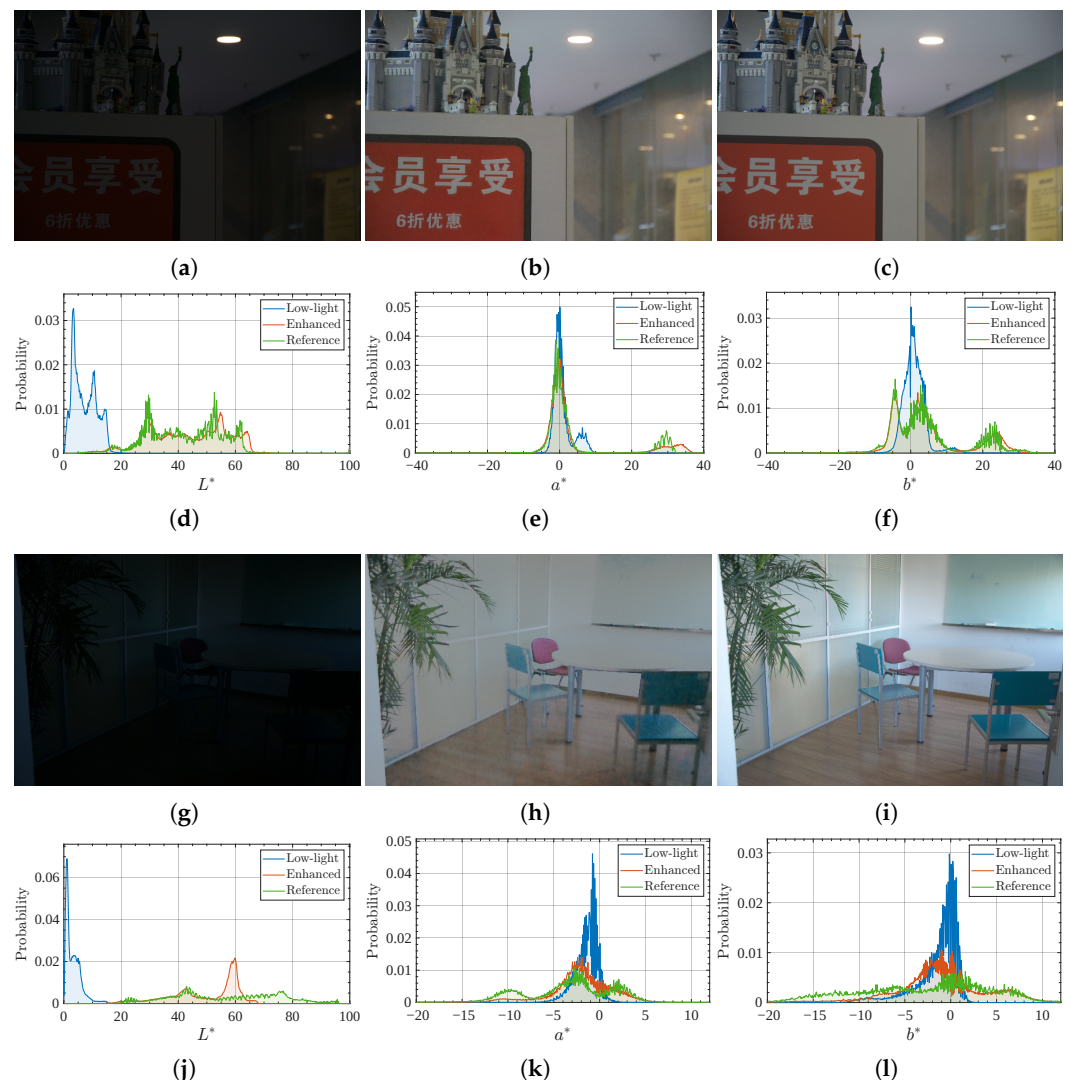
**Figure 7.** Qualitative enhancement results and normalized CIE-LAB histogram distributions for two representative low-light samples from the LOL-v1 dataset. (a–c) and (g–i) show the low-light inputs, enhanced outputs generated by the proposed model, and ground-truth images, respectively; (d–f,j–l) show the corresponding normalized CIE-LAB histogram distributions for each sample. Specifically, (d,j), (e,k), and (f,l) correspond to the luminance  $L^*$ , green–red  $a^*$ , and blue–yellow  $b^*$  channels, respectively.

Figure 8 shows representative visual results and CIE-LAB histogram comparisons from the LOL-v2-Real dataset.

Figure 8a,g show the original low-light images, in which severe underexposure limits the visibility of scene structures and objects. Figure 8b,h show the enhanced images generated by the proposed model, displaying a clear improvement in brightness, contrast, and overall visibility. In particular, details in background regions, indoor objects, furniture, and structural elements become more distinguishable after enhancement.

Figure 8c,i present the corresponding ground-truth images. Compared with these references, the enhanced results preserve the main spatial content and recover relevant visual details without evident overexposure. Figure 8d–f,j–l present the normalized CIE-LAB histograms associated with each visual comparison. The  $L^*$ ,  $a^*$ , and  $b^*$  channels correspond to subfigures (d,j), (e,k), and (f,l), respectively. The  $L^*$  histograms show the luminance recovery achieved by the model. In contrast, the  $a^*$  and  $b^*$  histograms indicate that the enhanced images maintain a chromatic distribution consistent with the

ground-truth images. Overall, the visual results and CIE-LAB histograms indicate improved visibility, enhanced luminance, and consistent color preservation.



**Figure 8.** Qualitative enhancement results and normalized CIE-LAB histogram distributions for two representative low-light samples from the LOL-v2-Real dataset. (a–c) and (g–i) show the low-light inputs, enhanced outputs generated by the proposed model, and ground-truth images, respectively; (d–f,j–l) show the corresponding normalized CIE-LAB histogram distributions for each sample. Specifically, (d,j), (e,k), and (f,l) correspond to the luminance  $L^*$ , green–red  $a^*$ , and blue–yellow  $b^*$  channels, respectively.

Figure 9 shows qualitative enhancement results and normalized CIE-LAB histogram distributions for two representative samples from the LOL-v2-Synthetic dataset. Figure 9a–c and Figure 9g–i show the low-light inputs, the enhanced outputs, and the corresponding ground-truth images, respectively. The original low-light images exhibit severe underexposure, reducing scene visibility and obscuring relevant architectural and background details.

After enhancement, the proposed model increases global brightness and improves local contrast, making structures such as buildings, windows, vegetation, sky regions, and shadowed areas more distinguishable.



**Figure 9.** Qualitative enhancement results and normalized CIE-LAB histogram distributions for two representative low-light samples from the LOL-v2-Synthetic dataset. (a–c) and (g–i) show the low-light inputs, enhanced outputs generated by the proposed model, and ground-truth images, respectively; (d–f,j–l) show the corresponding normalized CIE-LAB histogram distributions for each sample. Specifically, (d,j), (e,k), and (f,l) correspond to the luminance  $L^*$ , green–red  $a^*$ , and blue–yellow  $b^*$  channels, respectively.

The corresponding normalized CIE-LAB histograms are presented in Figure 9d–f for the first sample and Figure 9j–l for the second sample. These distributions provide complementary evidence of the enhancement behavior across luminance and chromatic components. In this case, the enhanced  $L^*$  distributions shift from darker regions toward brighter intensity ranges, while the  $a^*$  and  $b^*$  histograms show closer alignment with the ground-truth chromatic distributions.

Although the enhanced images preserve the main spatial structures and recover relevant visual details, minor deviations in color tone, saturation, and illumination distribution remain visible compared with the ground-truth images. This behavior indicates

that the proposed model substantially improves visibility and structural perception while still showing small differences relative to the reference images in highly illuminated or color-sensitive regions.

## 5. Discussion

The experimental results indicate that the proposed architecture offers a satisfactory and feasible GAN-based alternative for low-light image enhancement, particularly under real low-light conditions. In this case, the illumination degradation, contrast loss, and color distortion must be corrected simultaneously. In particular, Group Normalization is consistent with the adopted small-batch training regime. Similarly, spectral normalization applied to the PatchGAN discriminator was a good design choice; it helps restrict the network's Lipschitz constant to  $\leq 1$ , which prevents gradient divergence and also prevents the discriminator from overpowering the generator. In contrast, the multi-scale bottleneck and the channel-wise recalibration mechanism enrich the generator's contextual and discriminative capacities. Thus, the contextual feature enhancement utilizes an intermediate residual bottleneck ( $n_{res} = 6$ ) with transitioning dilation factors ( $d = 1$  to  $d = 2$ ) combined with ASPP to elegantly scale up the receptive field. This prevents the loss of fine structural data that typically occurs with excessive spatial downsampling.

The quantitative results indicate effective performance on LOL-v1, good results on LOL-v2-Real (second best SSIM of 0.8446), and the strongest results of the proposed model on LOL-v2-Synthetic, where it attains the highest PSNR (22.30 dB), SSIM (0.9154), and the lowest LPIPS (0.1022) among the three benchmarks, ranking second behind [20] among the compared methods. By contrast, on LOL-v2-Real, the promising margin narrows, since dedicated methods achieve markedly higher full-reference fidelity; Shin and Lee [38] reported a PSNR of 29.14 dB, compared with 19.62 dB for the proposed model, although the latter retained a comparatively high SSIM of 0.8446.

A plausible explanation is that the more regular and regular-consistent degradations of the synthetic subset, together with its larger training partition, are reconstructed more accurately, whereas a reduced dataset with real-world low-light heterogeneous sensor noise and richer spatially varying illumination conditions is harder to enhance. In this sense, the discrepancy between reference-based and no-reference metrics also indicates that perceptual improvement does not always translate into a uniformly strong structural reconstruction.

Compared with the representative LLIE methods reported in Tables 2 and 3, the proposed model does not uniformly dominate all quality criteria, but it occupies a satisfactory trade-off region across a rich diversity of metrics and has lower inference time than other high-capacity LLIE architectures. These results demonstrate efficient GPU utilization despite the increased number of parameters and floating-point operations.

Its best performance across the evaluated datasets is observed on LOL-v2-Synthetic, where the residual GAN formulation effectively reconstructs the synthetic degradations. Nevertheless, beyond the results of Song [20], this study suggests that model architectural modifications should be incorporated to improve reconstruction fidelity and visual-perceptual similarity across a broader range of conditions.

On LOL-v2-Real, the model preserves a reasonable structural similarity but does not match the full-reference fidelity of the leading methods, where illumination, contrast, and color degradations are naturally coupled.

The main implication of the proposed model is not superiority in universal metrics but balanced and robust enhancement behavior across the evaluated benchmarks, supported by competitive no-reference quality and low inference times for real-time applications.

Structural trade-offs of the proposed architecture are highlighted because it offers high model capacity and low inference latency. The system requires 171 GFLOPs and utilizes 26.3 M parameters. Nevertheless, it achieves a fast inference time of 0.0366 s per image. This behavior is a characteristic advantage of fully convolutional architectures: by avoiding dense fully connected layers and non-parallelizable operations (such as coordinate attention variants or heavy transformer windows). In this way, GPUs can efficiently exploit tensor-core parallelism.

Although the ablation study evaluated the main architectural and training components of the proposed framework, the interaction among RC, SN, SE, ASPP, BN/GN, and the optimizer remains dataset- and metric-dependent. Consequently, the reported behavior should be interpreted as the result of the complete configuration and its component interactions, rather than as an isolated effect of a single module. In addition, although an efficiency comparison is provided, inference-time values across methods remain affected by differences in hardware, input resolution, batch size, numerical precision, and implementation details. These aspects will be addressed in future work through broader cross-dataset validation, hardware-aware efficiency assessment, and exploration of additional objective functions to improve robustness under diverse real-world low-light conditions.

## 6. Conclusions

This work presents a GAN-based framework for low-light image enhancement built upon a ResUNet-like generator and a conditional PatchGAN discriminator. The proposed generator combines Group Normalization, reflection-padded convolutions, dilated residual blocks, Atrous Spatial Pyramid Pooling, and SE attention to improve training stability, capture multi-scale contextual information, and preserve relevant structural and chromatic details. The experimental evaluation on LOL-v1, LOL-v2-Real, and LOL-v2-Synthetic showed that the proposed model achieves competitive performance on LOL-v1 and competitive structural similarity on LOL-v2-Real, whereas its most favorable reference-based results are obtained on LOL-v2-Synthetic, as indicated by higher PSNR and SSIM values and a lower LPIPS value than those obtained on the two former datasets.

Overall, these results suggest that the proposed architecture is a viable alternative for low-light image enhancement by integrating stable optimization with multi-scale feature modeling.

From a practical perspective, the proposed framework is relevant to low-light imaging scenarios in which visual reliability is critical, including surveillance, autonomous perception, nighttime imaging, and computer-aided visual analysis. However, the slightly reduced robustness observed under more challenging real low-light conditions indicates that further improvements are required to enhance generalization across complex illumination distributions. Future work will focus on improving robustness to severe and spatially non-uniform degradations, extending the evaluation to broader real-world low-light datasets, and exploring more efficient implementations for resource-constrained deployment.

The ablation results further support the final design choice by showing that ResGASP-GAN is better justified by the joint behavior of RC, SN, SE, ASPP, and AdamW-GN than by any isolated component or single metric. This analysis reinforces the selected configuration as a balanced solution for illumination correction, structural preservation, and perceptual/no-reference quality across heterogeneous low-light conditions.

**Author Contributions:** Data curation, J.R.-P. and P.D.B.-A.; formal analysis, E.O.-M., M.A.I.-M., and J.G.A.-C.; investigation, F.D.H.-G., J.R.-P., and J.G.A.-C.; methodology, E.O.-M. and F.D.H.-G.; software, F.D.H.-G.; validation, M.A.I.-M., E.O.-M., and J.G.A.-C.; writing—original draft, F.D.H.-G.

and P.D.B.-A.; writing—review and editing, J.R.-P. and J.G.A.-C. All authors have read and agreed to the published version of the manuscript.

**Funding:** The Mexican Secretariat of Science, Humanities, Technology, and Innovation (SECIHTI) under scholarship CVU 1080385, and the University of Guanajuato through the Institutional Call for Scientific Research: CIIC-318/2026.

**Data Availability Statement:** The data presented in this study are openly available in the repository: <https://github.com/fdhernandezgutierrez/ResGASP-GAN> (accessed on 6 June 2026).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Mo, T.; Zheng, S.; Chan, W.Y.; Yang, R. Review of AI Image Enhancement Techniques for In-Vehicle Vision Systems Under Adverse Weather Conditions. *World Electr. Veh. J.* **2025**, *16*, 72. [\[CrossRef\]](#)
2. Ai, S.; Kwon, J. Extreme Low-Light Image Enhancement for Surveillance Cameras Using Attention U-Net. *Sensors* **2020**, *20*, 495. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Chen, T.; Lyu, Q.; Bai, L.; Guo, E.; Gao, H.; Yang, X.; Ren, H.; Zhou, L. LightTDiff: Surgical Endoscopic Image Low-Light Enhancement with T-Diffusion. In *Medical Image Computing and Computer Assisted Intervention, Proceedings of the MICCAI 2024, Marrakesh, Morocco, 6–10 October 2024*; Linguraru, M.G., Dou, Q., Feragen, A., Giannarou, S., Glocker, B., Lekadir, K., Schnabel, J.A., Eds.; Springer: Cham, Switzerland, 2024; pp. 369–379. [\[CrossRef\]](#)
4. Lu, P.; Huang, Q. Robotic Weld Image Enhancement Based on Improved Bilateral Filtering and CLAHE Algorithm. *Electronics* **2022**, *11*, 3629. [\[CrossRef\]](#)
5. Liu, X.; Pedersen, M.; Wang, R. Survey of natural image enhancement techniques: Classification, evaluation, challenges, and perspectives. *Digit. Signal Process.* **2022**, *127*, 103547. [\[CrossRef\]](#)
6. Liu, F.; Fan, L. A review of advancements in low-light image enhancement using deep learning. *Neurocomputing* **2025**, *652*, 131052. [\[CrossRef\]](#)
7. Jingchun, Z.; Eg Su, G.; Shahrizal Sunar, M. Low-light image enhancement: A comprehensive review on methods, datasets and evaluation metrics. *J. King Saud Univ.-Comput. Inf. Sci.* **2024**, *36*, 102234. [\[CrossRef\]](#)
8. Du, S.; Wen, Y.; Zhao, M.; Li, J.; Shi, Z.; Liu, Y. A decoupled framework for low-light image enhancement. *Expert Syst. Appl.* **2026**, *297*, 129436. [\[CrossRef\]](#)
9. Bai, G.; Yan, H.; Liu, W.; Deng, Y.; Dong, E. Towards lightest low-light image enhancement architecture for mobile devices. *Expert Syst. Appl.* **2026**, *296*, 129125. [\[CrossRef\]](#)
10. Qian, L.; Jiang, L. LIENet: A low-light image enhancement network for extreme darkness. *J. King Saud Univ. Comput. Inf. Sci.* **2025**, *37*, 344. [\[CrossRef\]](#)
11. Yang, R.; Chen, H.; Luo, S.; Wang, Z. DELIA-Net: A Detail-Enhanced and Local Illumination Adjustment Network for nighttime road image enhancement. *Displays* **2025**, *90*, 103174. [\[CrossRef\]](#)
12. Zhao, A.; Zheng, Q.; Li, L. Application of CycleGAN-Based Low-Light Image Enhancement Algorithm in Foreign Object Detection on Belt Conveyors in Underground Mines. *Sci. Rep.* **2025**, *15*, 27897. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Reis, M.J.C.S. Low-Light Image Enhancement Using Deep Learning: A Lightweight Network with Synthetic and Benchmark Dataset Evaluation. *Appl. Sci.* **2025**, *15*, 6330. [\[CrossRef\]](#)
14. Wu, J.; Ai, H.; Zhou, P.; Wang, H.; Zhang, H.; Zhang, G.; Chen, W. Low-Light Image Dehazing and Enhancement via Multi-Feature Domain Fusion. *Remote Sens.* **2025**, *17*, 2944. [\[CrossRef\]](#)
15. Cha, Z.; Tang, Y. Zero-shot low-light image enhancement via scattering-based retinex decomposition. *Digit. Signal Process.* **2026**, *170*, 105772. [\[CrossRef\]](#)
16. Zainab, L.; Afzal, H.; Mahmood, K.; Arif, O. A light-weight deep learning framework for Low Light Image Enhancement. *Neurocomputing* **2025**, *621*, 129236. [\[CrossRef\]](#)
17. Jiang, B.; Wang, X.; Yang, N.; Liu, Y.; Chen, X.; Wu, Q. Semantic-Aware Low-Light Image Enhancement by Learning from Multiple Color Spaces. *Appl. Sci.* **2025**, *15*, 5556. [\[CrossRef\]](#)
18. He, M.; Wang, R.; Zhang, M.; Lv, F.; Wang, Y.; Zhou, F.; Bian, X. SwinLightGAN a study of low-light image enhancement algorithms using depth residuals and transformer techniques. *Sci. Rep.* **2025**, *15*, 12151. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Yin, L.; Liu, P.; Zhang, T.; Zhao, H.; Zhou, Q. Differentiable histogram-guided unsupervised Retinex enhancement for paired low-light images. *Expert Syst. Appl.* **2026**, *298*, 129782. [\[CrossRef\]](#)
20. Song, S. Noise-Resilient Low-Light Image Enhancement with CLIP Guidance and Pixel-Reordering Subsampling. *Electronics* **2025**, *14*, 4839. [\[CrossRef\]](#)

21. Zeng, T.; Wang, T.; Zhang, M.; Yin, J.; Zeng, Z.; Zhang, F.; Wang, Y.; Jiao, J.; Wang, Y.; He, Y.; et al. TCSTNet: A text-driven color style transfer network for low-light image enhancement. *Expert Syst. Appl.* **2026**, *299*, 130012. [[CrossRef](#)]
22. Zhao, Z.; Shi, D. Retinex-guided generative diffusion prior for low-light image enhancement. *Pattern Recognit.* **2026**, *172*, 112421. [[CrossRef](#)]
23. Takase, R.; Yoshikawa, N.; Mariyama, T.; Tsuchiya, T. Stability-certified reinforcement learning control via spectral normalization. *Mach. Learn. Appl.* **2022**, *10*, 100409. [[CrossRef](#)]
24. Shu, Z.; Zhang, K. Spectral Normalization for Generative Adversarial Networks for Artistic Image Transformation. *Int. J. Digit. Multimed. Broadcast.* **2024**, *2024*, 6644706. [[CrossRef](#)]
25. Faye, B.; Azzag, H.; Lebbah, M.; Feng, F. Context normalization: A new approach for the stability and improvement of neural network performance. *Data Knowl. Eng.* **2025**, *155*, 102371. [[CrossRef](#)]
26. Boongasame, L.O.; Ketsamorn, J.; Thammarak, K. Laor Initialization: A New Weight Initialization Method for the Backpropagation of Deep Learning. *AI* **2025**, *9*, 181. [[CrossRef](#)]
27. Wu, Y.; He, K. Group Normalization. *Int. J. Comput. Vis.* **2020**, *128*, 742–755. [[CrossRef](#)]
28. Söylemez, Ö.F. SkinAttn-Net: A multi-level attention-based network for skin lesion segmentation. *Sci. Rep.* **2025**, *16*, 3700. [[CrossRef](#)] [[PubMed](#)]
29. Hettiarachchi, P.; Nawaratne, R.; Alahakoon, D.; De Silva, D.; Chilamkurti, N. Rain Streak Removal for Single Images Using Conditional Generative Adversarial Networks. *Appl. Sci.* **2021**, *11*, 2214. [[CrossRef](#)]
30. Liang, H.; Yu, A.; Shao, M.; Tian, Y. Multi-Feature Guided Low-Light Image Enhancement. *Appl. Sci.* **2021**, *11*, 5055. [[CrossRef](#)]
31. Zhao, R.; Xie, M.; Feng, X.; Su, X.; Zhang, H.; Yang, W. Content-illumination coupling guided low-light image enhancement network. *Sci. Rep.* **2024**, *14*, 8456. [[CrossRef](#)] [[PubMed](#)]
32. Hernandez-Gutierrez, F.D.; Bravo-Aguilar, P.D.; Ovalle-Magallanes, E.; Ibarra-Manzano, M.A.; Ruiz-Pinales, J.; Avina-Cervantes, J.G. ResGASP-GAN: Source Code for Low-Light Image Enhancement. GitHub Repository. 2026. Available online: <https://github.com/fdhernandezgutierrez/ResGASP-GAN> (accessed on 25 May 2026).
33. Sun, Z.; Shen, Z.; Chen, N.; Pang, S.; Liu, H.; You, Y.; Wang, H.; Zhu, Y. MEFormer: Enhancing Low-Light Images While Preserving Image Authenticity in Mining Environments. *Remote Sens.* **2025**, *17*, 1165. [[CrossRef](#)]
34. 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](#)]
35. Wang, Z.; Bovik, A.; Sheikh, H.; Simoncelli, E. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)] [[PubMed](#)]
36. Gunawan, R.; Tran, Y.; Zheng, J.; Nguyen, H.; Chai, R. Optimizing Natural Image Quality Evaluators for Quality Measurement in CT Scan Denoising. *Computers* **2025**, *14*, 18. [[CrossRef](#)]
37. Mittal, A.; Moorthy, A.K.; Bovik, A.C. Blind/Referenceless Image Spatial Quality Evaluator. In Proceedings of the 2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Pacific Grove, CA, USA, 6–9 November 2011; pp. 723–727. [[CrossRef](#)]
38. Shin, J.S.; Lee, H.S. A multi-scale U-shaped transformer neural network for low-light image enhancement. *Image Vis. Comput.* **2026**, *166*, 105801. [[CrossRef](#)]
39. Qi, Z.; Sun, S.; Liu, X.; Zhou, L.; Qiao, J.; Zhu, L. LFT-Net: A lightweight frequency-based transformer for low-light enhancement and exposure correction. *Pattern Recognit.* **2026**, *173*, 112887. [[CrossRef](#)]
40. Liu, Z.; Fan, T.; Zhang, W.; Wang, W.; Huang, J.; Lan, R.; Lu, H. Illuminating the Shadows: Enhanced Low-Light Image via a Retinex-based Model with Color Equalization. *Expert Syst. Appl.* **2026**, *296*, 129039. [[CrossRef](#)]
41. Batziou, E.; Ioannidis, K.; Patras, I.; Vrochidis, S.; Kompatsiaris, I. HDD-Unet: A Unet-based architecture for low-light image enhancement. *Image Vis. Comput.* **2026**, *167*, 105889. [[CrossRef](#)]
42. Jeon, Y.; Kim, H. Efficient Image Enhancement via Representative Color Transform. *IEEE Access* **2024**, *12*, 76458–76468. [[CrossRef](#)]
43. Ren, Z.; Zhu, H.; Yuan, R.; Zhang, J.; Zhao, L. An unsupervised low-light image enhancement method for underground scenes using low- and high-frequency enhancement calibration. *Measurement* **2026**, *263*, 120176. [[CrossRef](#)]

**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.