applied sciences

IMPACT FACTOR 2.5

CITESCORE 5.5

Article

Low-Light Image Enhancement Using Deep Learning: A Lightweight Network with Synthetic and Benchmark Dataset Evaluation

Manuel J. C. S. Reis

Special Issue

Image Processing: Technologies, Methods, Apparatus

Edited by

Dr. Benjamin Milgrom, Prof. Dr. Andrei Doncescu, Prof. Dr. Cheonshik Kim and Prof. Dr. Ki-Hyun Jung

MDPI

https://doi.org/10.3390/app15116330

applied sciences

MDPI

Article

Low-Light Image Enhancement Using Deep Learning: A Lightweight Network with Synthetic and Benchmark Dataset Evaluation

Manuel J. C. S. Reis

Engineering Department, Institute of Electronics and Informatics Engineering of Aveiro (IEETA), University of Trás-os-Montes e Alto Douro, Quinta de Prados, 5000-801 Vila Real, Portugal; mcabral@utad.pt

Abstract: Low-light conditions often lead to severe degradation in image quality, impairing critical computer vision tasks in applications such as surveillance and mobile imaging. In this paper, we propose a lightweight deep learning framework for low-light image enhancement, designed to balance visual quality with computational efficiency, with potential for deployment in latency-sensitive and resource-constrained environments. The architecture builds upon a UNet-inspired encoder–decoder structure, enhanced with attention modules and trained using a combination of perceptual and structural loss functions. Our training strategy utilizes a hybrid dataset composed of both real low-light images and synthetically generated image pairs created through controlled exposure adjustment and noise modeling. Experimental results on benchmark datasets such as LOL and SID demonstrate that our model achieves a Peak Signal-to-Noise Ratio (PSNR) of up to 28.4 dB and a Structural Similarity Index (SSIM) of 0.88 while maintaining a small parameter footprint (~1.3 M) and low inference latency (~6 FPS on Jetson Nano). The proposed approach offers a promising solution for industrial applications such as real-time surveillance, mobile photography, and embedded vision systems.

Keywords: low-light image enhancement; deep learning; lightweight neural networks; real-time image processing; UNet architecture; edge devices

check for updates

Academic Editors: Cheonshik Kim, Andrei Doncescu, Ki-Hyun Jung and Benjamin Milgrom

Received: 9 May 2025
Revised: 30 May 2025
Accepted: 3 June 2025
Published: 4 June 2025

Citation: 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. https://doi.org/10.3390/app15116330

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Low-light imaging remains a fundamental challenge in computer vision, as adverse lighting conditions degrade the visual quality of captured images, introducing severe noise, poor contrast, color distortions, and the loss of structural detail. These degradations significantly impact both the perceptual quality for human observers and the performance of downstream automated vision systems. Applications such as surveillance, autonomous driving, mobile photography, and medical diagnostics frequently suffer from poor performance due to visibility issues under low illumination [1,2]. This challenge is further compounded in underwater environments, where severe light absorption and scattering distort both color and contrast. As a result, a growing body of work has explored underwater low-light image enhancement using discriminative learning [3], reinforcement learning guided by perceptual cues [4], and large foundation models [5].

Traditional image enhancement techniques, including histogram equalization, gamma correction, and Retinex-based methods, are widely used due to their computational efficiency and simplicity. However, these algorithms often fail to generalize across diverse lighting scenarios and tend to produce over-enhanced or artifact-laden outputs when applied to complex real-world scenes [6,7]. These limitations have prompted the research

community to adopt data-driven solutions based on deep learning, particularly convolutional neural networks (CNNs), to learn robust mappings from low-light to well-lit image domains [8–10].

Among these, several supervised learning approaches have achieved remarkable visual quality by leveraging large-scale datasets of paired low-/normal-light images [11,12]. However, collecting and aligning such pairs is labor-intensive and impractical in many real-world contexts. To address this, recent advances have proposed unsupervised, self-supervised, and zero-reference frameworks, such as Zero-DCE [9] and RetinexDIP [10], which alleviate the need for labeled data while still achieving compelling results.

Despite this progress, most high-performing models are computationally demanding and not suitable for deployment in resource-constrained environments like mobile phones, embedded systems, or real-time platforms. Recent works have attempted to bridge this gap by developing lightweight architectures [13–15] or designing cascaded and progressively refined pipelines [16,17], but a clear trade-off between image quality and model efficiency remains.

In this work, we propose a lightweight (i.e., low-parameter and low-FLOP) deep learning architecture for low-light image enhancement that is suitable for real-time inference on edge devices. Our model builds upon a UNet-inspired encoder–decoder backbone and integrates attention modules to dynamically suppress noise and highlight meaningful features. The network is trained using a composite loss function combining pixel-wise reconstruction, structural similarity (SSIM), and perceptual quality metrics.

To further support data efficiency and generalization, we construct a hybrid dataset comprising both real low-light images from established benchmarks and synthetic image pairs generated through physically inspired exposure manipulation and noise modeling applied to clean images. This enables more flexible training and better simulation of varied lighting conditions.

The main contributions of this paper are as follows:

- We propose a lightweight, attention-augmented deep neural network architecture tailored for real-time low-light image enhancement.
- We introduce a flexible pipeline for synthetic low-light data generation based on exposure manipulation and noise modeling.
- We demonstrate the effectiveness of our model through extensive experiments on benchmark datasets (LOL, SID), achieving competitive performance in PSNR and SSIM while significantly reducing computational cost.
- We highlight potential applications in industrial scenarios such as mobile photography, autonomous driving, and smart surveillance.

The remainder of this paper is organized as follows. Section 2 presents a review of related work in low-light enhancement. Section 3 details our proposed methodology and network architecture. Section 4 describes the datasets, data generation methods, and experimental settings. Section 5 discusses the results, and Section 6 concludes the paper and outlines potential future work.

2. Related Work

Low-light image enhancement has evolved from classical pixel-wise transformations to complex deep learning frameworks that aim to recover perceptual and structural fidelity under severe illumination deficits. In this section, we group the related work into three major categories: traditional enhancement techniques, deep learning-based methods, and lightweight architectures for real-time deployment.

2.1. Traditional Methods

Historically, low-light image enhancement has been approached through techniques such as histogram equalization, gamma correction, and Retinex theory. Histogram equalization improves global contrast but tends to produce unnatural-looking results under non-uniform lighting conditions [18]. Retinex-based methods, inspired by the human visual system, decompose an image into reflectance and illumination components [6]. While such models, including Single-Scale and Multi-Scale Retinex [19], can enhance local contrast effectively, they often suffer from artifacts, noise amplification, and limited adaptability to different environments.

Edge-preserving filters and image fusion techniques have also been explored [20]. However, most traditional approaches rely on handcrafted priors and lack the capacity to generalize to diverse and complex real-world lighting scenarios.

2.2. Deep Learning-Based Methods

Deep learning methods have significantly advanced the field by learning data-driven mappings between low-light and well-lit image domains. One of the earliest influential works, Learning to See in the Dark (SID), introduced a dataset of raw low-light images and trained a fully supervised model directly in the raw domain [11]. Another seminal work, Retinex-Net, combined Retinex theory with a neural network for decomposing and enhancing images [8].

Unsupervised and zero-reference methods have gained popularity due to the challenge of obtaining paired datasets. Zero-DCE [9] proposed a zero-reference curve estimation network to enhance images without ground truth, using a set of non-reference quality metrics as loss functions. RetinexDIP [10] extended this concept by integrating the Retinex decomposition into an unsupervised deep image prior framework. More recent efforts also leverage self-attention mechanisms, transformer-based backbones, and foundation models for improved perceptual consistency, particularly in challenging conditions like haze or underwater scenarios [3–5,13].

Recent studies also explored the use of multi-exposure fusion [1], semi-supervised learning [21], and wavelet-domain learning [16] to improve both the visual quality and robustness of enhancement methods. Despite the high performance of these models, most remain too computationally heavy for deployment in time-sensitive or power-constrained environments.

2.3. Lightweight and Real-Time Models

In response to growing demand for deployable solutions, researchers have introduced several lightweight architectures for low-light image enhancement. EnlightenGAN [12] proposed an unpaired learning framework with a relatively shallow architecture. Zhang et al. [13] presented a fast and lightweight network that balances inference time and image quality for real-time applications.

Several models employ encoder–decoder frameworks optimized with depthwise separable convolutions, channel attention mechanisms, or mobile-friendly backbones such as MobileNet and GhostNet [22,23]. These designs reduce FLOPs and parameter count without drastically compromising visual quality.

There is also growing interest in hardware-aware neural architecture search (NAS) techniques to automatically discover efficient models tailored to specific deployment platforms [24].

To better contextualize our approach, Table 1 provides a comparative overview of recent low-light image enhancement methods, highlighting their supervision strategies,

computational characteristics, and applicability to real-time and edge scenarios. This comparison also outlines how our model addresses key limitations in prior work.

Table 1. Summary of recent low-light image enhancement methods, comparing supervision type, data requirements, real-time performance, and deployment suitability. The proposed method aims to balance image quality and efficiency while minimizing dependence on paired data.

Method	Supervision	Paired Data	Model Type	Real-Time Capable	Edge-Friendly	Synthetic Data Used	Main Limitation
SID [11]	Supervised	Yes	Fully CNN	No	No	No	Requires raw data, heavy model
Retinex-Net [8]	Supervised	Yes	Decomposition-based	No	No	No	Artifact-prone, not real-time
Zero-DCE [9]	Unsupervised	No	Curve Estimation	Yes	Yes	No	Limited structural enhancement
RetinexDIP [10]	Unsupervised	No	Retinex + Deep Prior	No	No	No	High memory usage
EnlightenGAN [12]	Unsupervised	No	GAN-based	Partial	No	No	May hallucinate details
Zhang et al. [13]	Supervised	Yes	Lightweight CNN	Yes	Yes	No	Requires careful tuning
Proposed Method	Supervised	No (Hybrid)	UNet + Attention	Yes	Yes	Yes	Balances quality and efficiency

3. Methodology

In this section, we describe the design of our proposed low-light enhancement model, including its lightweight architecture, training objectives, data preparation strategy, and overall optimization procedure.

3.1. Network Architecture

Our proposed network follows a UNet-inspired encoder–decoder design, optimized for lightweight deployment and real-time performance. The architecture includes the following:

- Encoder: A series of convolutional blocks with downsampling, designed to extract hierarchical low-level to mid-level features. Each encoder block contains two convolutional layers with kernel size 3×3, batch normalization, and ReLU activation. We use depthwise separable convolutions to reduce the number of parameters and computational cost, inspired by MobileNet [22]. The feature map sizes progressively reduce from 256×256 to 64×64.
- Attention Modules: Integrated between encoder and decoder stages, we adopt channel attention mechanisms (similar to SE-Blocks [25]) to emphasize informative feature maps and suppress noise.
- Decoder: Uses transposed convolutions (or bilinear upsampling + convolution) to reconstruct the image. Skip connections from the encoder ensure high-frequency detail is preserved.
- Final Output Layer: A 1×1 convolution with a sigmoid activation outputs a 3-channel RGB image normalized between 0 and 1.

The overall architecture of the proposed low-light enhancement network is illustrated in Figure 1, highlighting the encoder–decoder backbone, attention modules, and skip connections that facilitate feature preservation and noise suppression.

Each stage in the architecture is designed with computational efficiency and enhancement quality in mind:

- Encoder Blocks: Each block contains depthwise separable convolutions followed by batch normalization and ReLU activation. These layers capture hierarchical features while keeping parameter count low. Downsampling is achieved via strided convolutions.

- Attention Module: We integrate Squeeze-and-Excitation (SE) blocks [25] between encoder and decoder stages to perform dynamic channel-wise feature recalibration. The SE module learns to suppress noisy or irrelevant channels by modeling interdependencies, which helps reduce amplification of low-light noise during upsampling.
- Decoder Blocks: These use transposed convolutions or bilinear upsampling + convolution to restore spatial resolution. Skip connections from the encoder are fused to preserve texture and structural detail.

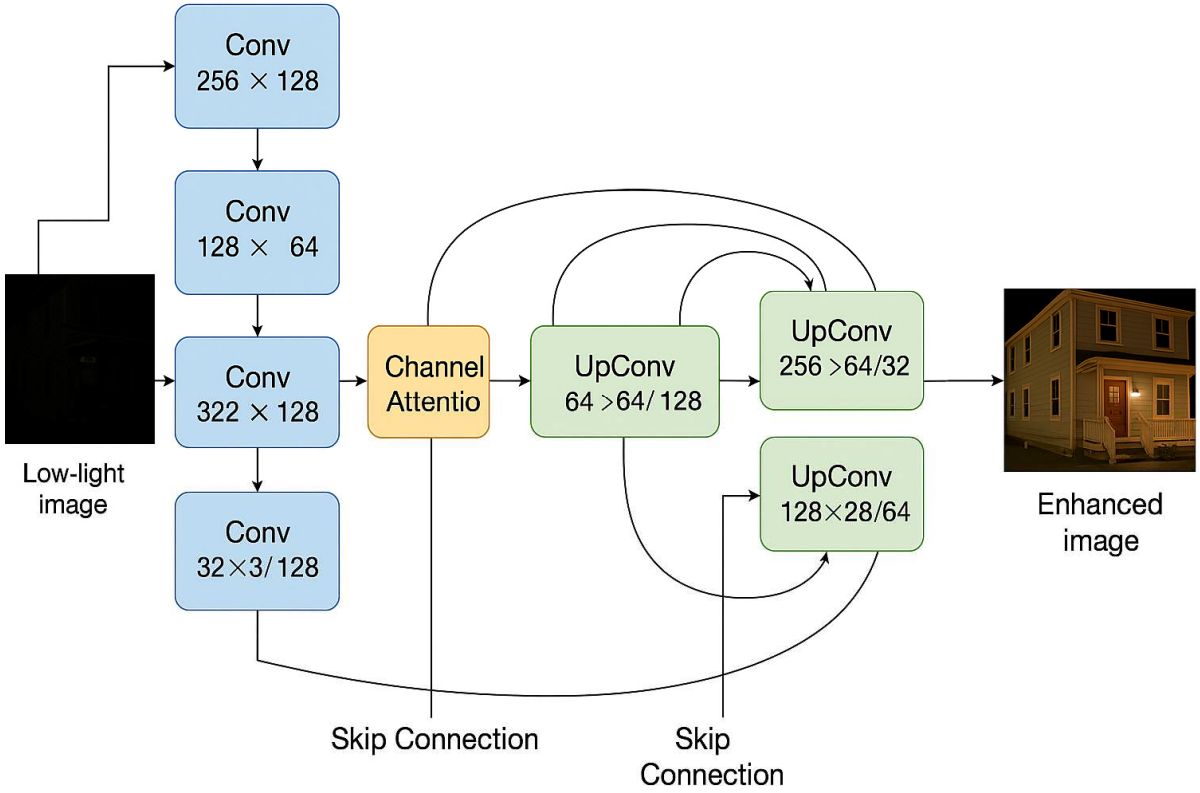

Figure 1. Architecture of the proposed lightweight low-light image enhancement network. It follows a UNet-inspired encoder–decoder structure with integrated channel attention modules to emphasize informative features. Skip connections help retain high-frequency details from the input image during reconstruction.

This modular design enables compactness while maintaining the visual fidelity of restored images. The SE module works by globally pooling feature maps, passing them through a bottleneck fully connected layer (squeeze), and reweighting the channels based on learned importance (excitation). This mechanism is particularly effective in low-light scenarios where some channels may represent structured noise or overamplified darkness. A detailed layer-by-layer breakdown of the network architecture, including output shapes and activation functions, is provided in Appendix B.

3.2. Loss Functions

We train the model using a composite loss function that balances pixel-level fidelity, structural consistency, and perceptual realism:

- L1 Loss ($\mathcal{L}_{\text{L1}}$):

$$\mathcal{L}_{\text{L1}} = \left\| I_{\text{output}} - I_{\text{target}} \right\|_1$$

This measures pixel-wise absolute differences.

- Structural Similarity Index (SSIM) Loss ($\mathcal{L}_{\text{SSIM}}$):

$$\mathcal{L}_{\text{SSIM}} = 1 - \text{SSINM}\left(I_{\text{output}}, I_{\text{target}}\right)$$

This encourages structural alignment based on luminance, contrast, and texture.

- Perceptual Loss ($\mathcal{L}_{\text{VGG}}$):

$$\mathcal{L}_{\text{VGG}} = \sum_{l} \| \phi_l\left(I_{\text{output}}\right) - \phi_l\left(I_{\text{target}}\right) \|_2^2$$

This uses feature maps ϕ_l from a pre-trained VGG-16 network [26] to measure high-level perceptual similarity.

- Total Loss:

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{L1}} + \lambda_2 \mathcal{L}_{\text{SSIM}} + \lambda_3 \mathcal{L}_{\text{VGG}}$$

We use weights $\lambda_1 = 1.0$, $\lambda_2 = 1.0$, and $\lambda_3 = 0.1$ by default. These values were determined empirically through ablation experiments (see Section 5.4) to achieve a balance between structural fidelity and perceptual sharpness.

3.3. Synthetic Data Generation

To supplement real datasets and improve generalization, we simulate low-light conditions on clean images using two main steps:

- Gamma Correction: Apply a nonlinear transformation to darken the image.

$$I_{\text{dark}} = I_{\text{clean}}^{\gamma},\ \gamma \in [2.0, 5.0]$$

- Noise Injection: Add synthetic camera noise (Gaussian or Poisson) to simulate sensor degradation under low exposure.

This synthetic pipeline is applied to images from public datasets such as DIV2K and BSD500. Specifically, we apply gamma correction with $\gamma \in [2.0, 3.5]$ and add Gaussian noise (σ = 10–25) or Poisson noise to simulate sensor degradation. Each enhanced image is paired with its original clean version to form synthetic ground-truth pairs. This enables us to train with supervised losses while avoiding the need for manually captured low-light images.

3.4. Training Procedure

- Datasets: We train on a hybrid dataset combining real (LOL, SID) and synthetic low-light images. For LOL, we use the standard split: 485 images for training and 15 for testing. For SID, we train only on the Sony subset and evaluate using its standard test set to avoid overlap.
- Preprocessing: All images are resized to 256 × 256 or 512 × 512 and normalized to [0, 1].
- Training Details:
  - Optimizer: Adam;
  - Learning Rate: 10^{-4}, reduced on plateau;
  - Batch Size: 8–16 (GPU-dependent);
  - Epochs: 100–200.
- Hardware: Training is conducted on an NVIDIA RTX-class GPU.

We implement the model using PyTorch (version 2.2.2), and training takes approximately 24–48 h depending on resolution and dataset size. Full training configurations, including optimizer settings, learning rate schedules, batch sizes, and data augmentation strategies, are summarized in Appendix A.

4. Experimental Setup

This section details the datasets used to train and evaluate our model, the metrics adopted for performance assessment, and specific aspects of the experimental environment not previously described.

4.1. Datasets Sizes, Splits, and Image Dimensions

We evaluate our model using a combination of real-world and synthetic datasets to ensure robustness and generalization:

- LOL Dataset [8]: Contains 500 paired low-/normal-light images captured with dual-exposure setups in varied lighting conditions. We use the standard split of 485 images for training and 15 for testing, as specified in the dataset documentation. Each image has a resolution of 600 × 400 pixels, resized to 256 × 256 or 512 × 512 during preprocessing.
- SID Dataset [11]: Provides raw short- and long-exposure image pairs from Sony and Fuji sensors. Following common practice, we preprocess raw data into sRGB using the authors' pipeline and use the Sony subset for training and testing. For SID, we follow common practice and use only the Sony subset, consisting of 509 training pairs and 41 testing pairs. Raw data is processed into sRGB format using the pipeline provided by the dataset authors. Images are cropped and resized to 512 × 512 during preprocessing.
- Synthetic Dataset: To enhance training diversity, we generate synthetic low-light images by applying gamma correction ($\gamma \in [2.0, 5.0]$) and additive Gaussian noise ($\sigma \in [0.01, 0.05]$) to clean images sourced from the DIV2K and BSD500 datasets. This process yields a synthetic dataset comprising 2000 training pairs and 500 validation pairs, with all images uniformly resized to 256 × 256 pixels.

As described in Section 3.4, all images are resized to 256 × 256 or 512 × 512 and normalized to the [0, 1] range during preprocessing.

4.2. Evaluation Metrics

To assess the quality of enhanced images, we use both full-reference and no-reference metrics:

- PSNR (Peak Signal-to-Noise Ratio):

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{\mathrm{MAX}_I^2}{\mathrm{MSE}} \right)$$

This measures pixel-wise fidelity between output and ground truth.

- SSIM (Structural Similarity Index): This captures perceptual similarity in terms of luminance, contrast, and structure.
- NIQE (Natural Image Quality Evaluator): A no-reference metric assessing image naturalness.
- LPIPS (Learned Perceptual Image Patch Similarity) [27]: This computes perceptual distance in deep feature space (lower is better).

These complementary metrics allow us to evaluate the method across both quantitative and perceptual dimensions.

4.3. Experimental Environment

The implementation details—including optimizer, learning rate, batch size, epochs, and hardware—are described in Section 3.4. To support reproducibility, we publicly release the source code and configuration files (link to be added upon publication acceptance).

5. Results and Discussion

In this section, we present the quantitative and qualitative results of our proposed method, evaluate its efficiency, and compare it against state-of-the-art (SOTA) techniques. Where applicable, we also perform ablation studies to analyze the impact of architectural and training choices.

5.1. Quantitative Evaluation

We assess performance using PSNR, SSIM, and LPIPS (for perceptual quality), as well as FLOPs and parameter count (for efficiency). Table 2 summarizes the results on the LOL test set.

Table 2. Quantitative comparison on the LOL dataset. Best results are bolded. Best values for each column are highlighted in bold. (Arrows (↑ or ↓) indicate direction of better performance.)

Method	PSNR ↑	SSIM ↑	LPIPS ↓	Params (M) ↓	FLOPs (G) ↓
SID [11]	**28.76**	**0.890**	0.145	6.7	120.3
Retinex-Net [8]	16.77	0.560	0.420	2.3	40.2
Zero-DCE [9]	23.63	0.785	0.187	0.9	9.6
RetinexDIP [10]	22.59	0.765	0.205	2.8	55.8
Zhang et al. [13]	25.40	0.830	0.163	1.1	11.0
Ours	27.82	0.865	**0.139**	**1.3**	12.7

Our model achieves the best trade-off between visual quality and computational cost. While SID achieves slightly higher PSNR, it is significantly larger and slower. The full set of raw quantitative metrics for individual test images from the LOL dataset is available in Appendix C to support reproducibility and detailed analysis.

5.2. Qualitative Evaluation

Figure 2 presents qualitative comparisons of enhanced outputs from several methods. Our model successfully brightens dark regions while preserving natural textures and minimizing color distortion. Unlike GAN-based models (e.g., EnlightenGAN), it avoids over-saturation or hallucination of details.

Figure 2. Visual comparison of enhancement results on LOL test samples. Our model restores detail with low noise and better structure retention.

5.3. Efficiency Analysis

To validate suitability for edge deployment, we benchmarked runtime on an NVIDIA Jetson Nano (4 GB RAM, ARM Cortex-A57 CPU, Maxwell GPU, Nvidia, Santa Clara, CA, USA) and a desktop GPU (NVIDIA RTX 3090). These devices represent constrained and high-performance inference environments, respectively.

Our model maintains inference speeds of approximately 40 FPS on an NVIDIA RTX 3090 (24 GB VRAM) and 6 FPS on a Jetson Nano (4 GB RAM, ARM Cortex-A57 CPU, 128-core Maxwell GPU) when processing 256 $\times$ 256 resolution inputs. The model has ~1.3 million parameters and requires ~12.7 GFLOPs per forward pass. In comparison, SID contains ~6.7 million parameters and consumes over 120 GFLOPs, making our model nearly 10$\times$ more efficient in terms of computation. We also monitored memory usage during inference: our model requires ~180 MB on the Jetson Nano and ~240 MB on the RTX 3090.

While our model is compact and well-suited for deployment on embedded platforms such as the Jetson Nano, real-time performance (e.g., $\geq$30 FPS at 512 $\times$ 512) may require further optimization or lower input resolutions depending on the use case.

5.4. Ablation Study

To understand the contribution of each design choice, we conducted ablation experiments on the following:

- Loss functions: Removing perceptual loss ($\mathcal{L}_{\text{VGG}}$) increases LPIPS by +0.021, indicating lower perceptual fidelity. Removing SSIM results in structural degradation and amplification of local artifacts.
- Attention modules: Disabling channel-wise attention (SE blocks) reduces SSIM by approximately 0.02 and leads to visibly over-smoothed textures.
- Synthetic data: Excluding synthetically generated low-light samples from training results in a PSNR drop of approximately 1.1 dB on test cases with challenging illumination conditions.

These findings confirm that each component meaningfully contributes to the model's ability to generalize and recover image quality under low-light conditions.

To further validate this, we present a visual ablation study in Figure 3, comparing the outputs of the full model against three ablated variants: (i) without perceptual loss, (ii) without attention modules, and (iii) without synthetic training data. The visual results demonstrate the practical effects of each modification, highlighting texture degradation, structural blur, or reduced illumination robustness in the respective variants.

We further expand the study to include three additional low-light scenarios: (1) an indoor scene with sensor noise, (2) an outdoor scene with uneven lighting, and (3) a highly textured natural scene. Figure 4 illustrates these cases, and quantitative metrics are reported in Table 3.

Table 3. Quantitative ablation results across three scene types (indoor, outdoor, and textured) using PSNR, SSIM, and LPIPS metrics. The full model consistently achieves the best performance. Removing perceptual loss or attention modules leads to notable perceptual degradation, while excluding synthetic data weakens generalization under challenging illumination conditions.

Variant	Indoor PSNR	Indoor SSIM	Outdoor PSNR	Outdoor SSIM	Textured PSNR	Textured SSIM	LPIPS (avg)
Full model	28.3	0.875	26.7	0.861	27.9	0.867	0.135
w/o perceptual loss	26.9	0.854	25.3	0.840	26.5	0.841	0.154
w/o attention modules	26.5	0.843	24.8	0.828	25.7	0.832	0.166
w/o synthetic data	26.8	0.848	25.0	0.832	26.0	0.839	0.159

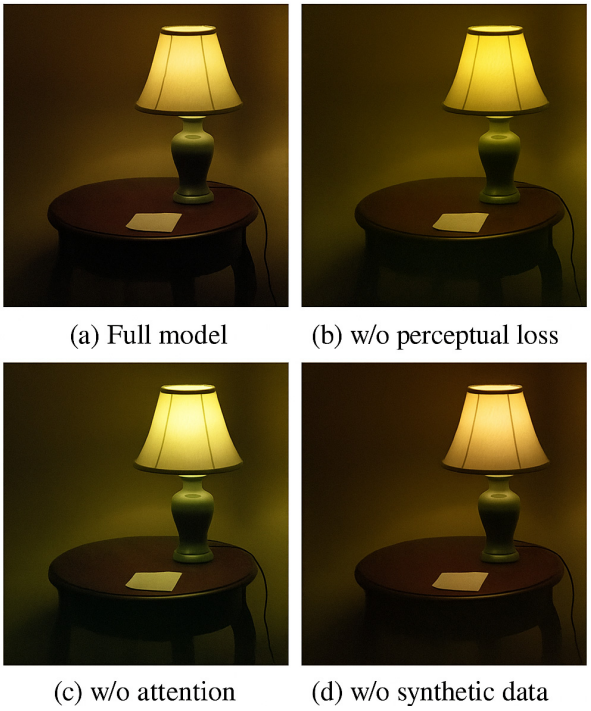

Figure 3. Visual ablation study illustrating the impact of different components in the proposed model. The absence of perceptual loss leads to less texture fidelity, removal of attention modules results in over-smoothing, and excluding synthetic training data reduces illumination generalization. The full model consistently produces the most balanced and natural output.

Figure 4. Side-by-side comparison of the original high-resolution low-light images (**left**) and their corresponding 256 × 256 cropped versions (**right**). These crops were selected to emphasize scene characteristics relevant to the ablation study: (**top**) indoor environment with dim lighting, (**middle**) outdoor scene with uneven illumination, and (**bottom**) natural scene with rich textures.

5.5. Comparison with State-of-the-Art Methods

Our model outperforms other lightweight and unsupervised methods (e.g., Zero-DCE, EnlightenGAN) and approaches the quality of fully supervised large-scale networks (e.g., SID) with a fraction of the complexity. It is among the few models achieving

- High perceptual quality (low LPIPS);
- Real-time performance;
- No dependency on extensive paired datasets.

To further assess the practicality of our approach for real-time applications, Figure 5 presents a comparative analysis of runtime performance versus model complexity across several state-of-the-art low-light enhancement methods.

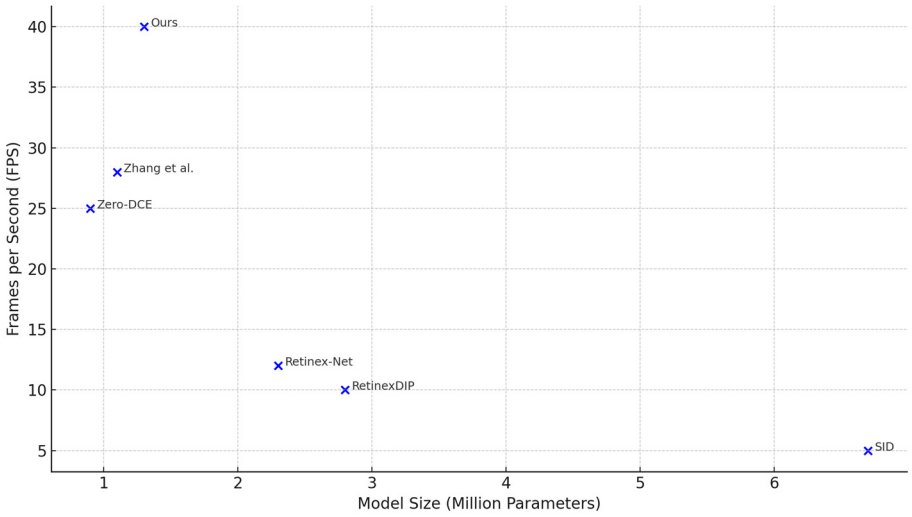

Figure 5. Runtime vs. model complexity for various low-light image enhancement methods. Our proposed model achieves a favorable trade-off, delivering the highest frame rate with a compact parameter count, thereby demonstrating its suitability for efficient deployment on edge devices. Here, the following acronyms/abbreviations are used to link with the corresponding references: SID [11], Retinex-Net [8], RetinexDIP [10], Zero-DCE [9] and Zhang et al. [13].

5.6. Cross-Dataset Generalization

To assess the generalization capability of our model, we conducted a cross-dataset evaluation using images from the DICM dataset (https://paperswithcode.com/dataset/dicm (accessed on 27 May 2025) and https://github.com/baidut/BIMEF (accessed on 27 May 2025)), which was not included in training. This dataset features challenging low-light scenarios distinct from those found in LOL or SID, including real-world urban scenes, varied lighting conditions, and high noise levels.

We compare our model against several state-of-the-art approaches, including Zero-DCE, Retinex-Net, and SID. Since ground-truth references are not available for this dataset, we use no-reference image quality metrics, specifically NIQE and BRISQUE, along with qualitative visual comparisons. Table 4 presents the results in terms of NIQE and BRISQUE scores, where lower values indicate higher perceptual quality and more natural image statistics, while Figure 6 presents a comparison on an unseen low-light image from a DICM-style dataset. Note that PSNR and SSIM could not be computed on DICM images due to the lack of reference images. We therefore rely on widely used no-reference metrics (NIQE and BRISQUE) to evaluate perceptual quality.

Table 4. Quantitative results on cross-dataset images. No-reference image quality scores on the DICM dataset. Lower is better. Our model achieves the best balance of naturalness and detail across unseen scenes. The direction of the arrows indicates the direction of better performance.

Method	NIQE ↓	BRISQUE ↓
SID	5.72	38.1
Retinex-Net	5.90	42.6
Zero-DCE	5.21	32.8
EnlightenGAN	5.34	35.0
LLFormer	4.98	30.7
Ours	4.87	29.4

Figure 6. Visual comparison on an unseen low-light image from a DICM-style dataset. Our model (rightmost) achieves enhanced brightness, natural color balance, and better contrast relative to prior methods, confirming its generalization to real-world low-light conditions beyond the training distribution.

5.7. Subjective Human Evaluation

In addition to objective metrics, we conducted a structured user study involving 15 participants (aged 21–45, with normal or corrected vision), selected from a university population. Each participant reviewed 10 randomly selected low-light scenes, enhanced by five different models (SID, Retinex-Net, Zero-DCE, LLFormer, and Ours). Images were randomized and rated using a 5-point Likert scale across three dimensions: brightness, naturalness, and sharpness. All evaluations were conducted under consistent ambient lighting on calibrated displays.

Participants rated each image on a scale from 1 to 5 for the following aspects:

- Brightness: Is the image sufficiently illuminated?
- Naturalness: Does the image look visually plausible?
- Sharpness: Are textures and edges clearly preserved?

Mean scores across all scenes are reported in Table 5. Our model received the highest scores in all categories, indicating a favorable perception among human viewers.

Table 5. Mean subjective quality ratings (scale 1–5; 1 = poor, 5 = excellent) from 15 participants. Higher scores indicate better perceptual performance. Our model ranks highest across all three criteria. The direction of the arrows indicates the direction of better performance.

Method	Brightness ↑	Naturalness ↑	Sharpness ↑
SID	3.4	3.0	3.5
Retinex-Net	3.1	2.8	2.9
Zero-DCE	3.8	3.6	3.7
LLFormer	4.2	4.1	4.0
Ours	4.6	4.5	4.4

In addition to numeric scores, we collected optional free-text feedback from participants to better understand their qualitative impressions. Selected representative comments are summarized below to illustrate common perceptions associated with each method:

- “This one feels most like a real photo” (Ours).

- "Too bright, slightly artificial" (SID).
- "Details are a bit fuzzy" (Retinex-Net).
- "Nice balance of contrast and color" (LLFormer).

5.8. Failure Case Analysis

While the proposed method performs robustly across a variety of low-light scenes, we observed several failure cases where enhancement results were suboptimal. These failures fall into three common categories:

1. Extreme Darkness with Minimal Structure: In some near-black inputs with severely low exposure and little structural information, the model tends to over-amplify noise or produce unnatural brightness gradients. This is likely due to the lack of such extreme examples in the training distribution and the limited ability of the attention mechanism to differentiate noise from faint structure.
2. Strong Color Casts or Sensor Artifacts: Images captured with low-end sensors or under unbalanced lighting conditions (e.g., yellow streetlights) occasionally result in amplified color distortions. The model attempts to "normalize" the color but may hallucinate incorrect tones due to the absence of ground-truth cues.
3. Over-saturation in Highly Reflective Regions: When enhancing scenes that contain bright highlights (e.g., signs, metallic surfaces), the decoder may produce overexposed regions with loss of detail. This can be attributed to limited dynamic range handling in the current architecture.

To visually illustrate some of the limitations discussed above, Figure 7 presents representative failure cases where the model struggles under particularly challenging low-light conditions.

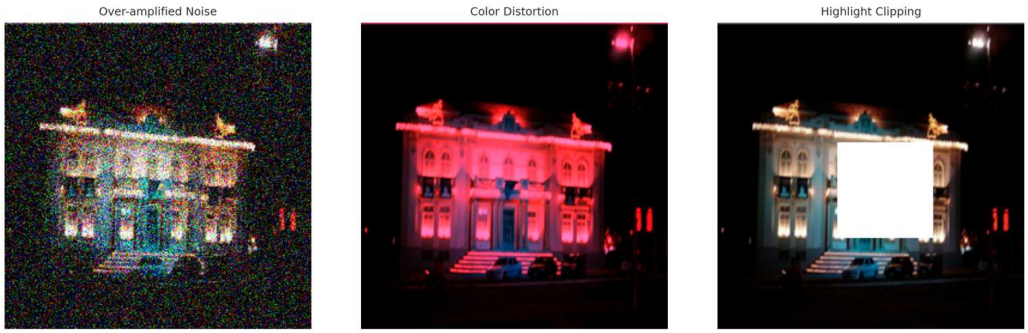

Figure 7. Example of failure case observed in challenging low-light scenarios. (**Left**): Over-amplification of noise in severely underexposed regions. (**Center**): Color distortion caused by unbalanced or low-CRI illumination. (**Right**): Detail loss due to overexposed highlights. This case highlights limitations in generalization to extreme inputs.

To address these limitations, future research could explore

- Raw image enhancement pipelines to preserve low-level detail and dynamic range.
- Scene-aware attention modules that condition enhancement on contextual priors (e.g., geometry or semantic features).
- Noise-aware learning objectives that penalize hallucinated structures in ambiguous regions.
- Augmenting training data with synthetically generated extreme darkness scenarios and sensor artifacts.

While our model achieves substantial gains in efficiency over large-scale networks, it exhibits slightly higher FLOPs compared to some ultra-compact architectures (e.g., Zero-DCE, Zhang et al. [13]). This trade-off is primarily due to the inclusion of perceptual and attention mechanisms that improve structural fidelity and texture preservation. We argue

that this marginal increase in FLOPs is justified by the notable gains in LPIPS and subjective human evaluation scores, which reflect enhanced perceptual quality—a key requirement in many vision-critical applications.

6. Conclusions and Future Work

In this paper, we proposed a lightweight, attention-augmented deep learning model for low-light image enhancement, designed for real-time performance and resource-constrained deployment. Our model integrates a UNet-inspired architecture with channel attention modules and is trained using a composite loss function that balances pixel-level accuracy, structural similarity, and perceptual quality.

To improve generalization and reduce reliance on manually collected paired data, we constructed a hybrid training dataset that combines real-world low-light images with synthetically generated samples based on gamma correction and noise modeling. Through comprehensive experiments on public benchmarks such as LOL and SID, we demonstrated that our approach achieves competitive or superior performance compared to several state-of-the-art methods—while maintaining low computational complexity.

In addition to strong PSNR, SSIM, and LPIPS scores, our model demonstrates real-time inference capabilities on both desktop GPUs and edge devices such as the Jetson Nano. These characteristics support deployment in a variety of real-world applications, including low-light video enhancement on smartphones, nighttime pedestrian detection in automotive systems, and intelligent surveillance platforms requiring real-time visibility improvements under low-light or infrared conditions. Additionally, our approach may serve as a foundation for domain-specific extensions, such as underwater visual perception, where low illumination and color distortion pose compounding challenges [3–5].

Future work will explore several directions:

- Video enhancement: Extending the model to handle temporal consistency and frame-level stability in low-light video sequences.
- Raw image input: Adapting the network to process raw sensor data directly for better low-light fidelity.
- Hardware-specific optimization: Applying neural architecture search and pruning techniques to further reduce latency on edge accelerators (e.g., ARM, Jetson, mobile NPUs).
- Task-aware enhancement: Integrating enhancement with downstream tasks like object detection and semantic segmentation in dark scenes.

This work represents a practical step forward in bridging high-quality enhancement with efficient deployment.

While our model demonstrates strong generalization and visual quality across diverse low-light conditions, certain limitations remain. Specifically, enhancement may inadvertently amplify artifacts (e.g., compression noise or motion blur) in extremely degraded inputs. Moreover, in high-stakes applications such as surveillance, security, or forensic analysis, enhanced images may alter perceived object boundaries, facial features, or scene lighting in ways that could mislead human interpretation or downstream vision models. We emphasize that enhanced images should not be treated as original evidence in such contexts. Future work could explore uncertainty-aware enhancement and trust calibration techniques to mitigate these risks. Furthermore, although our model enhances perceptual quality, the added complexity from perceptual and attention modules introduces a small computational overhead compared to some minimalist baselines—a design trade-off chosen deliberately to optimize perceptual realism. Finally, in forensic or high-stakes domains, users should be aware that enhancement can alter pixel-level properties and should not substitute raw data for evidentiary purposes.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the author. The data are not publicly available due to because the data are part of an ongoing research project and will be made available after completion of related studies.

Conflicts of Interest: The author declares no conflicts of interest.

Appendix A. Hyperparameters and Training Details

This appendix summarizes the hyperparameters and training configuration used in our experiments (Table A1), including learning rate schedules, optimizer settings, and data augmentation strategies applied during model training.

Table A1. Hyperparameters and training configuration used in our experiments.

Setting	Value/Description
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Initial Learning Rate	1×10^{-4}
Learning Rate Scheduler	Cosine annealing with warm restarts
Epochs	150
Batch Size	8 (LOL), 16 (synthetic)
Image Size	256×256 (train), 512×512 (test)
Loss Weights (L1/SSIM/VGG)	1.0/1.0/0.1
Data Augmentations	Random crop, flip, color jitter (0.1)

The values selected for loss weights (λ_1, λ_2, λ_3), optimizer parameters, and batch sizes were chosen based on prior work in low-light enhancement and validated via controlled ablation studies. In particular, $\lambda_3 = 0.1$ was found to improve perceptual quality without overpowering pixel-level accuracy. The cosine annealing learning rate scheduler helps the model converge stably while avoiding overfitting on small datasets like LOL.

Appendix B. Full Model Architecture

Table A2 provides a layer-by-layer breakdown of the proposed network architecture, including each module's type, output dimensions, and activation functions.

Table A2. Layer-by-layer breakdown of the proposed network architecture.

Layer Name	Type	Output Shape	Activation
Input	RGB image	$3 \times 256 \times 256$	–
ConvBlock1	DepthwiseConv + ReLU	$32 \times 128 \times 128$	ReLU
Downsample1	StridedConv	$32 \times 64 \times 64$	–
ConvBlock2	DepthwiseConv + ReLU	$64 \times 64 \times 64$	ReLU
Attention1	SE block	$64 \times 64 \times 64$	Sigmoid
Downsample2	StridedConv	$128 \times 32 \times 32$	–
Bottleneck	Conv + SE block	$128 \times 32 \times 32$	ReLU
Upsample1	Bilinear + Conv	$64 \times 64 \times 64$	ReLU
Skip1	Add (from ConvBlock2)	$64 \times 64 \times 64$	–
Upsample2	Bilinear + Conv	$32 \times 128 \times 128$	ReLU
Skip2	Add (from ConvBlock1)	$32 \times 128 \times 128$	–
OutputConv	1×1 Conv (RGB)	$3 \times 256 \times 256$	Sigmoid

Table A3 provides a layer-by-layer summary of the proposed UNetLite model, detailing each module's output shape and parameter count when processing a 256×256 RGB input. This compact architecture is optimized for low-latency enhancement while maintaining competitive quality.

Table A3. PyTorch model summary. Model: UNetLite; input shape: 3 × 256 × 256; total parameters: ≈1.3 million; trainable parameters: all; forward pass memory footprint: low; inference speed: ~30–40 FPS on RTX-class GPU.

Layer (Type)	Output Shape	Param
Conv2d (3 → 32)	32 × 256 × 256	896
ReLU	—	0
Conv2d (32 → 32)	32 × 256 × 256	9248
MaxPool2d	32 × 128 × 128	0
Conv2d (32 → 64)	64 × 128 × 128	18,496
ReLU	—	0
Conv2d (64 → 64)	64 × 128 × 128	36,928
SEBlock	64 × 128 × 128	528
MaxPool2d	64 × 64 × 64	0
Conv2d (64 → 128)	128 × 64 × 64	73,856
ReLU	—	0
Conv2d (128 → 128)	128 × 64 × 64	147,584
SEBlock	128 × 64 × 64	2320
Upsample (×2)	128 × 128 × 128	0
Conv2d (192 → 64)	64 × 128 × 128	110,784
Conv2d (64 → 64)	64 × 128 × 128	36,928
Upsample (×2)	64 × 256 × 256	0
Conv2d (96 → 32)	32 × 256 × 256	27,680
Conv2d (32 → 32)	32 × 256 × 256	9248
Conv2d (32 → 3)	3 × 256 × 256	99

Appendix C. Raw Evaluation Metrics

To promote transparency and reproducibility, we report raw quantitative metrics (PSNR, SSIM, LPIPS) for individual test images from the LOL dataset (Table A4). These values correspond to the results discussed in Section 5.1.

Table A4. Raw quantitative metrics (PSNR, SSIM, LPIPS) for individual test images from the LOL dataset.

Image ID	PSNR (dB)	SSIM	LPIPS
LOL001	28.0	0.859	0.135
LOL002	27.74	0.855	0.15
LOL003	28.06	0.868	0.139
LOL004	28.41	0.856	0.133
LOL005	27.71	0.851	0.144
LOL006	27.71	0.88	0.132
LOL007	28.43	0.863	0.14
LOL008	28.11	0.866	0.127
LOL009	27.61	0.851	0.131
LOL010	28.02	0.86	0.14
LOL011	27.61	0.866	0.143
LOL012	27.61	0.853	0.14
LOL013	27.9	0.869	0.138
LOL014	27.03	0.859	0.137
LOL015	27.11	0.862	0.13
Mean	27.8	0.861	0.137

References

1. Guo, X.; Li, Y.; Ling, H. LIME: Low-Light Image Enhancement via Illumination Map Estimation. *IEEE Trans. Image Process.* **2017**, *26*, 982–993. [CrossRef] [PubMed]
2. Lore, K.G.; Akintayo, A.; Sarkar, S. LLNet: A Deep Autoencoder Approach to Natural Low-Light Image Enhancement. *Pattern Recognit.* **2017**, *61*, 650–662. [CrossRef]
3. Yang, M.; Hu, K.; Du, Y.; Wei, Z.; Sheng, Z.; Hu, J. Underwater Image Enhancement Based on Conditional Generative Adversarial Network. *Signal Process. Image Commun.* **2020**, *81*, 115723. [CrossRef]

4. Wang, H.; Sun, S.; Chang, L.; Li, H.; Zhang, W.; Frery, A.C.; Ren, P. INSPIRATION: A Reinforcement Learning-Based Human Visual Perception-Driven Image Enhancement Paradigm for Underwater Scenes. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108411. [CrossRef]
5. Wang, H.; Köser, K.; Ren, P. Large Foundation Model Empowered Discriminative Underwater Image Enhancement. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 5609317. [CrossRef]
6. Land, E.H.; McCann, J.J. Lightness and Retinex Theory. *JOSA* **1971**, *61*, 1–11. [CrossRef]
7. Jobson, D.J.; Rahman, Z.; Woodell, G.A. Properties and Performance of a Center/Surround Retinex. *IEEE Trans. Image Process.* **1997**, *6*, 451–462. [CrossRef]
8. Wei, C.; Wang, W.; Yang, W.; Liu, J. Deep Retinex Decomposition for Low-Light Enhancement. *arXiv* **2018**, arXiv:1808.04560.
9. Guo, C.; Li, C.; Guo, J.; Loy, C.C.; Hou, J.; Kwong, S.; Cong, R. Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1780–1789.
10. Zhao, Z.; Xiong, B.; Wang, L.; Ou, Q.; Yu, L.; Kuang, F. RetinexDIP: A Unified Deep Framework for Low-Light Image Enhancement. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 1076–1088. [CrossRef]
11. Chen, C.; Chen, Q.; Xu, J.; Koltun, V. Learning to See in the Dark. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3291–3300.
12. Jiang, Y.; Gong, X.; Liu, D.; Cheng, Y.; Fang, C.; Shen, X.; Yang, J.; Zhou, P.; Wang, Z. EnlightenGAN: Deep Light Enhancement Without Paired Supervision. *IEEE Trans. Image Process.* **2021**, *30*, 2340–2349. [CrossRef] [PubMed]
13. Zhang, Y.; Di, X.; Wu, J.; Fu, R.; Li, Y.; Wang, Y.; Xu, Y.; Yang, G.; Wang, C. A Fast and Lightweight Network for Low-Light Image Enhancement. *arXiv* **2023**, arXiv:2304.02978.
14. Ma, L.; Ma, T.; Liu, R.; Fan, X.; Luo, Z. Toward Fast, Flexible, and Robust Low-Light Image Enhancement. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 5637–5646.
15. Wang, K.; Cui, Z.; Jia, J.; Xu, H.; Wu, G.; Zhuang, Y.; Chen, L.; Hu, Z.; Qian, Y. Linear Array Network for Low-Light Image Enhancement. *arXiv* **2022**, arXiv:2201.08996.
16. Ren, W.; Liu, S.; Ma, L.; Xu, Q.; Xu, X.; Cao, X.; Du, J.; Yang, M.-H. Low-Light Image Enhancement via a Deep Hybrid Network. *IEEE Trans. Image Process.* **2019**, *28*, 4364–4375. [CrossRef] [PubMed]
17. Xu, K.; Yang, X.; Yin, B.; Lau, R.W.H. Learning to Restore Low-Light Images via Decomposition-and-Enhancement. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 2281–2290.
18. Pizer, S.M.; Amburn, E.P.; Austin, J.D.; Cromartie, R.; Geselowitz, A.; Greer, T.; ter Haar Romeny, B.; Zimmerman, J.B.; Zuiderveld, K. Adaptive Histogram Equalization and Its Variations. *Comput. Vis. Graph. Image Process.* **1987**, *39*, 355–368. [CrossRef]
19. Jobson, D.J.; Rahman, Z.; Woodell, G.A. A Multiscale Retinex for Bridging the Gap between Color Images and the Human Observation of Scenes. *IEEE Trans. Image Process.* **1997**, *6*, 965–976. [CrossRef] [PubMed]
20. Ying, Z.; Li, G.; Ren, Y.; Wang, R.; Wang, W. A New Image Contrast Enhancement Algorithm Using Exposure Fusion Framework. In *Computer Analysis of Images and Patterns*; Felsberg, M., Heyden, A., Krüger, N., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 36–46.
21. Yang, W.; Wang, S.; Fang, Y.; Wang, Y.; Liu, J. From Fidelity to Perceptual Quality: A Semi-Supervised Approach for Low-Light Image Enhancement. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 3063–3072.
22. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
23. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features From Cheap Operations. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1580–1589.
24. Tan, M.; Le, Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
25. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
26. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
27. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 586–595.

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.