Perceptually Optimised Swin-Unet for Low-Light Image Enhancement

Tomasz M. Lehmann* and Przemysław Rokita
Warsaw University of Technology, Warsaw, Poland
*Corresponding author: Tomasz M. Lehmann (tomasz.lehmann.dokt@pw.edu.pl)

Submitted: May 27, 2025 **Accepted:** Oct 12, 2025 **Published:** Nov 12, 2025

Abstract In this paper we propose a novel approach to low-light image enhancement using a transformer-based Swin-Unet and a perceptually driven loss that incorporates Learned Perceptual Image Patch Similarity (LPIPS), a deep-feature distance aligned with human visual judgements.

Specifically, our U-shaped Swin-Unet applies shifted-window self-attention across scales with skip connections and multi-scale fusion, mapping a low-light RGB image to its enhanced version in one pass. Training uses a compact objective – Smooth-L_1, LPIPS (AlexNet), MS-SSIM (detached), inverted PSNR, channel-wise colour consistency, and Sobel-gradient terms – with a small LPIPS weight chosen via ablation.

Our work addresses the limits of purely pixel-wise losses by integrating perceptual and structural components to produce visually superior results. Experiments on LOL-v1, LOL-v2, and SID show that while our Swin-Unet does not surpass current state-of-the-art on standard metrics, the LPIPS-based loss significantly improves perceptual quality and visual fidelity.

These results confirm the viability of transformer-based U-Net architectures for low-light enhancement, particularly in resource-constrained settings, and suggest exploring larger variants and further tuning of loss parameters in future work.

Keywords: low-light image enhancement, U-Net, mean opinion score, LPIPS.

1. Introduction

As shown below, numerous software frameworks, models, and methodologies have been proposed for the low-light enhancement task. Nevertheless, we extend this research by examining three persistent gaps – architecture, efficiency, and perception. Pure transformer U-Nets such as Swin-Unet [3] have been scarcely explored in this context, yet their hierarchical shifted-window attention is well suited to the joint global–local reasoning required by complex illumination. Moreover, state-of-the-art models almost exclusively optimise pixel-level errors, which correlate poorly with human judgement; colour shifts and texture flattening therefore persist. A composite loss that blends classic terms with a perceptual metric (LPIPS) [48] is needed to align optimisation with visual quality. In addition, many high-performing pipelines rely on heavy diffusion stages or multi-branch designs, whereas a lightweight, single-stage Swin-Unet promises a superior accuracy-efficiency trade-off – crucial for real-time or mobile applications.

These observations motivate our investigation of a perceptually optimised Swin-Unet

that couples the representational power of hierarchical transformers with an LPIPS-augmented composite loss, aiming to reduce residual artefacts while retaining computational frugality.

1.1. Related Works

Enhancing photographs captured in severe darkness has matured from handcrafted tone-mappers to sophisticated learning pipelines, yet every generation still negotiates its own trade-offs between fidelity, robustness, and speed. Early grey-level transformations and Retinex-based formulations [9, 10, 13, 14, 17, 27, 44] adjust global brightness through fixed, analytical rules that remain attractive for real-time use but inevitably falter when illumination varies across a scene, leaving local noise and colour bias unresolved. Retinex theory itself – explicitly separating reflectance from illumination – continues to underpin most modern networks: Retinex-Net [37] dissects, corrects, and re-merges the two layers in three consecutive modules, achieving joint denoising and brightening, although its separate branches occasionally amplify artefacts if any module under-fits. Diff-Retinex [43] replaces convolutions with Transformer Decomposition Networks (TDN) and diffusion-style adjusters that offer smoother global illumination at the cost of substantial inference latency introduced by the diffusion iterations. Alternative encoder–decoder designs regress a coarse illumination map and refine it in a single pass; their simplicity improves throughput but risks oversmoothing high-frequency detail. Two-stream recurrent models mitigate this blur by letting a secondary branch track salient textures, yet the recurrent roll-out lengthens both memory use and training time.

To preserve the fine structure of the image, in the subsequent work the multi-scale processing and attention was introduced. Unrolled optimisation with residual blocks and parallel multi-resolution streams [19, 45] retains context over very large receptive fields, but the extra resolution hierarchy enlarges GPU memory consumption. CDAN [31] adds dense connectivity and channel-attention to a U-Net skeleton, improving colour consistency and perceptual sharpness while inflating parameter count. SNR-aware attention [40] and residual dense attention units [50] explicitly weight features by estimated noise statistics, reducing information loss on consumer cameras, yet the reliance on a reliable SNR estimate can degrade accuracy when sensor characteristics change. Laplacian-pyramid diffusion in PyDiff [52] progressively samples higher resolutions so as to suppress global RGB shifts with fewer parameters than classic diffusion; nevertheless, its iterative denoiser remains too heavy for battery-powered hardware.

The field is therefore witnessing a parallel push toward lightweight yet perceptually solid designs. LYT-Net [1] splits the Y and UV channels into separate paths with a Channel-Wise Denoiser and a ViT-based fusion block, reaching mobile-class throughput; its dependence on an explicit YUV conversion, however, complicates end-to-end RAW processing pipelines. Self-DACE [38] alternates Adaptive Adjustment Curves with a

CNN-based denoiser in a two-stage loop and learns solely from unpaired data, generalising across cameras while effectively doubling runtime. Other lightweight attempts compress feature maps aggressively but tend to underperform on real photographs where noise, colour cast, and motion blur co-occur.

Collectively, these developments yield a toolbox that can brighten images, suppress grain, and restore colour, yet three persistent challenges remain. First, colour distortion survives in regions where statistical priors deviate from the true illumination spectrum. Second, texture fidelity still drops whenever a network relies exclusively on pixel-wise losses such as L_1 or MSE, encouraging overly smooth outputs. Third, computational overhead – either from deep cascades, recurrent loops, or diffusion steps – prevents many state-of-the-art models from running interactively on edge devices.

Transformers equipped with windowed self-attention offer a plausible route toward closing these gaps. The Swin Transformer family [21] combines convolution-like locality with long-range context in a hierarchical fashion that scales linearly with image size, and thus promises a more favourable accuracy–efficiency balance than global-attention ViTs. Embedding Swin blocks in an encoder–decoder topology inherits the strong reconstruction ability of U-Nets while eliminating the multi-branch overhead common in Retinex cascades or the multi-step burden of diffusion. Such a design can devote its full capacity to suppressing colour shifts and preserving texture within a single pass, potentially delivering competitive perceptual quality at a fraction of the compute budget. The present work therefore positions a Swin-based U-Net at the centre of the low-light enhancement landscape, evaluating it against both heavyweight perceptual optimisers and recent lightweight specialists, and highlighting where transformer attention can bridge the longstanding trade-off between fidelity, robustness, and real-time performance.

2. Experimental setup

2.1. Datasets

To comprehensively evaluate our proposed method for low-light image enhancement, we utilized two prominent benchmark datasets specifically designed for addressing challenges associated with underexposed photography: the LOL and SID datasets. These datasets provide paired low-light and normal-light images, enabling supervised learning and detailed performance assessments. Additionally, to determine the most effective approach to data integration, we explored various dataset combinations, consistently using LOL for training, while systematically varying the inclusion and selection strategy of SID images (single darkest, three darkest, random selection, or none).

2.1.1. LOL Dataset

The LOL dataset [37] consists of pairs of images captured under low-light and normal-light conditions, primarily designed to support research focused on image enhancement

techniques. It includes 500 image pairs, of which 485 are used for training and 15 for testing. Most images in this dataset depict indoor scenes and maintain a uniform resolution of 400×600 pixels. Additionally, we employed an expanded version, known as LOL-v2, which provides 689 training and 100 testing image pairs. LOL-v2 notably enhances dataset variability by incorporating both synthetic and real-world low-light scenarios, allowing for more robust evaluations of algorithmic performance under diverse conditions.

2.1.2. SID Dataset

The See-in-the-Dark (SID) dataset [4] is a comprehensive collection of raw, short-exposure images accompanied by corresponding long-exposure reference images, tailored specifically for low-light enhancement studies. It comprises 5094 image pairs captured under various illumination conditions using two different professional-grade camera systems. This dataset uniquely offers multiple exposure levels per scene, providing valuable insights into the effectiveness of enhancement methods across varying degrees of darkness. In our experiments, we specifically evaluated multiple strategies for incorporating SID data into the training process. These strategies included selecting only the darkest exposure per scene, the three darkest exposures, random exposure selection, and excluding SID data entirely. This allowed us to rigorously investigate the impact of different dataset configurations on model performance and generalizability.

2.2. Proposed method

The goal of this work is to investigate whether a carefully tuned and loss-optimised lightweight architecture based on Swin-Unet [3] can achieve performance competitive with current state-of-the-art models for low-light image enhancement. In contrast to many recent approaches that incorporate multiple complex modules or multi-stage designs [1, 31, 52], we focus on a streamlined and efficient model that leverages the global context modelling capabilities of Vision Transformers while maintaining the desirable properties of U-Net's encoder-decoder structure.

We hypothesize that, with the right combination of architectural design and a composite loss function tailored to perceptual and structural fidelity, a pure transformer-based model can deliver good results on both synthetic and real-world low-light datasets.

2.2.1. Model Architecture

Our proposed model builds upon Swin-Unet [3], a pure Transformer architecture originally developed for medical image segmentation. The architecture follows a symmetric U-shaped design composed entirely of Swin Transformer blocks [21], organized into an encoder, bottleneck, and decoder, interconnected through skip connections.

The encoder consists of a patch embedding layer followed by four hierarchical stages of Swin Transformer blocks and patch merging layers, progressively reducing spatial

resolution while increasing feature dimensionality. The bottleneck module operates at the lowest resolution, capturing deep contextual features.

The decoder mirrors the encoder structure, utilizing patch expanding layers and Swin Transformer blocks to restore spatial resolution and refine the feature representations. Skip connections are introduced at each level to recover fine-grained spatial information lost during downsampling.

Unlike traditional CNN-based U-Nets, Swin-Unet replaces convolutional layers with self-attention mechanisms using shifted windows. This allows the model to efficiently capture both local details and long-range dependencies without excessive computational overhead. A final upsampling module brings the output back to the original image resolution, followed by a 1×1 convolution to produce the enhanced image.

2.2.2. Loss function

The most commonly used loss functions in low-light image enhancement tasks are the Mean Absolute Error (MAE), often referred to as L_1-loss, and the Mean Squared Error (MSE), also known as L_2-loss. These functions have been widely adopted due to their simplicity and effectiveness in pixel-wise intensity comparison.

Recent top-tier works, such as [52] and [2], prominently utilize the L_1-loss, highlighting its continued relevance in state-of-the-art models. The formula for L_1-loss is given by:

$$L_1 = \frac{1}{N}\sum_{i=1}^{N} |\hat{y}_i - y_i| \ , \tag{1}$$

where $\hat{y}_i$ denotes the predicted pixel value, y_i is the corresponding ground-truth value, and N is the total number of pixels. For comparison, the L_2 loss (mean squared error, MSE) is defined as:

$$L_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \ . \tag{2}$$

While L_2-loss penalizes large deviations more heavily, leading to smoother outputs, L_1-loss is less sensitive to outliers and often results in sharper reconstructions. This distinction makes L_1-loss preferable in tasks requiring better preservation of image details.

In addition to pixel-wise losses, perceptual losses have gained popularity for improving the visual quality of enhanced images. In [31], the authors utilize a combination of MSE and perceptual loss based on a pre-trained VGG19 network. The perceptual loss compares feature maps from different layers of the VGG19 network for both generated and reference images, ensuring better high-level feature alignment. The perceptual loss is formulated as:

$$L_{\text{VGG}} = \frac{1}{N}\sum_{i=1}^{N} \|\text{VGG}(\hat{I}_i) - \text{VGG}(I_i)\|_2^2 \, , \tag{3}$$

where $\hat{I}_i$ and I_i represent the predicted and ground truth images, respectively, and VGG denotes the feature extraction function using the VGG19 network.

The composite loss function used in this work combines MSE and perceptual loss as follows:

$$L_{\text{composite}} = L_{\text{MSE}} + \lambda L_{\text{VGG}}\,, \tag{4}$$

where λ is a hyperparameter balancing the contributions of the two components. According to the authors, $\lambda = 0.25$ yields optimal results.

Similarly, [8] proposes a loss function designed for low-light image enhancement in both HVI and sRGB colour spaces; we will refer to it as *FN-loss* in the remainder of this paper to simplify the nomenclature. The total loss L is defined as:

$$L = \lambda_c \cdot l(\hat{I}_{\text{HVI}}, I_{\text{HVI}}) + l(\hat{I}, I)\,, \tag{5}$$

where $\hat{I}_{\text{HVI}}$ and I_{HVI} are the predicted and ground truth images in the HVI colour space, $\hat{I}$ and I are the predicted and ground truth images in the sRGB colour space, and λ_c is a weight balancing the two losses.

The loss function l for each colour space consists of multiple components:

$$l(\hat{X}, X) = \lambda_1 L_1(\hat{X}, X) + \lambda_e L_e(\hat{X}, X) + \lambda_p L_p(\hat{X}, X)\,, \tag{6}$$

where: L_1 loss denotes the pixel-wise L_1 loss, L_e is the edge loss encouraging edge preservation in the enhanced image, and L_p is the perceptual loss, ensuring perceptual similarity by comparing features extracted by a pre-trained network (e.g., VGG19). λ_1, λ_e, and λ_p are weights controlling the contributions of the respective loss components.

The proposed approaches demonstrate the efficacy of combining multiple loss components, including pixel-wise, edge, and perceptual losses, to achieve enhanced brightness, colour accuracy, and edge sharpness in low-light image enhancement tasks.

A notable example of an advanced loss function design is presented in [1]. The authors of LYT-Net used a hybrid loss function that combines multiple components to jointly optimise image brightness, perceptual quality, structural similarity, and colour fidelity. Their loss function can be expressed as:

$$L_{\text{total}} = L_S + \alpha_1 L_{\text{Perc}} + \alpha_2 L_{\text{Hist}} + \alpha_3 L_{\text{PSNR}} + \alpha_4 L_{\text{colour}} + \alpha_5 L_{\text{MS-SSIM}}\,, \tag{7}$$

where: L_{S} denotes the Smooth L_1 loss, applying a linear or quadratic penalty depending on the error magnitude to handle outliers effectively, L_{Perc} is the perceptual loss enforcing high-level feature consistency via VGG feature maps, L_{Hist} is the histogram loss aligning intensity distributions of prediction and ground truth, L_{PSNR} is the PSNR-based loss penalizing deviations in peak signal-to-noise terms, L_{colour} is the colour fidelity loss minimizing channel-wise mean differences, and $L_{\text{MS-SSIM}}$ is the multiscale structural similarity loss preserving structure across scales.

Each component in this hybrid loss function addresses a specific aspect of the enhancement problem, ensuring a balanced optimization process. This approach demonstrates how combining multiple loss terms can lead to excellent results in low-light image enhancement.

Both methods, [1] and [20], achieve excellent performance, particularly on synthetic datasets like LOLv2. However, models trained with simpler loss functions, such as the L_1-loss used in [52], tend to perform better on real-world datasets. This suggests that while advanced hybrid loss functions can improve performance on controlled datasets, simpler losses might generalize better in real-world scenarios. The superior real-world performance of [52] is likely influenced by the entire network architecture and training optimization strategy, including the choice of loss function.

In [20], the authors employ a vector quantization-based method for low-light image enhancement and define separate loss functions across three stages:

Stage I Loss: The goal is to train a normal-light encoder, decoder, and codebook using a combination of:

$$L_{\text{Stage I}} = L_{\text{recon}} + \beta L_{\text{vq}} \,, \tag{8}$$

where L_{recon} is the L_2-loss (Mean Squared Error) ensuring pixel-wise reconstruction accuracy, and L_{vq} is the vector quantization loss, which penalizes differences between the encoded and quantized features.

Stage II Loss: To bridge the gap between low-light and normal-light feature spaces, a distillation loss is introduced, alongside a query loss that optimises the matching process:

$$L_{\text{Stage II}} = L_{\text{distill}} + L_{\text{query}} \,, \tag{9}$$

Here, L_{distill} minimizes the feature-level discrepancy using L_1-loss, while L_{query} ensures accurate codebook item selection by aligning distance maps between features and codebook/query items.

Stage III Loss: In the final stage, a fusion branch combines features from different scales, and a brightness-aware attention module is employed to refine the enhanced image. The total loss in this stage is an L_1-loss defined as:

$$L_{\text{Stage III}} = \|I_{\text{rec}} - I_N\|_1 \tag{10}$$

where I_{rec} is the reconstructed image, and I_N is the ground truth normal-light image. Influence when parameters change: Eq. (10) has no explicit hyperparameters; if weighted by λ_3 in the total loss, increasing λ_3 scales the gradient $\partial L/\partial I_{\text{rec}} = \lambda_3 \operatorname{sign}(I_{\text{rec}} - I_N)$ and enforces pixel fidelity (typically higher PSNR/SSIM, smoother textures), while decreasing λ_3 lets perceptual/structural terms dominate (often sharper appearance with slight PSNR/SSIM trade-off). Replacing $\|\cdot\|_1$ with $\|\cdot\|_2^2$ would penalize large residuals more (more denoising/smoothness, potential edge blurring); keeping L_1 preserves edges and is outlier-robust. Stronger brightness-aware attention

concentrates updates in dark regions (better shadow recovery, risk of halos if excessive); weaker attention spreads updates (fewer artifacts, possible residual shadow noise). We use plain L_1 ($\lambda_3 = 1$ unless stated) and control the overall balance via Eq. (11).

To better align the network output with human visual perception, we augment classic pixel-wise objectives with a deep-feature component based on LPIPS [48]. The total training signal is defined as:

$$L_{\text{total}} = \alpha_{\text{S}} L_S + \alpha_P L_{\text{LPIPS}} + \alpha_M L_{\text{MS-SSIM}} + \alpha_N L_{\text{PSNR}} + \alpha_C L_{\text{colour}} + \alpha_G L_{\text{Grad}}\,, \quad (11)$$

where L_S is the Smooth-L_1 loss, $L_{\text{MS-SSIM}}$ is the multi-scale structural similarity loss (computed with detached gradients), L_{PSNR} is the inverted PSNR loss, L_{colour} penalizes differences in channel-wise mean values, and L_{Grad} enforces edge consistency using Sobel-based gradients. The perceptual term L_{LPIPS} uses the metric introduced by Zhang et al. [48], based on a frozen AlexNet backbone [16]. During training, both prediction and ground-truth images are forwarded through the LPIPS network in `no_grad` mode, after being rescaled from $[0, 1]$ to $[-1, 1]$, as required by the implementation. The choice of the LPIPS loss weight α_P was also subject to ablation, as we evaluated different values to balance perceptual quality and training stability. A comprehensive comparison of alternative loss functions and weight configurations is presented later in the paper.

2.2.3. Training setup

The complete pipeline is implemented in PyTorch 2.3 [28] with native AMP (Automatic Mixed Precision), uDNN (CUDA Deep Neural Network library) [26], benchmarking enabled, weight-initialization utilities from *timm* [25], and tensor rearrangements from *einops* [29, 30]. The Swin-Unet backbone is realised as a pure-attention U-Net: a patch-embedding stem feeds four encoder stages that alternate shifted-window multi-head self-attention, MLPs and residual connections, each stage halving the spatial resolution through patch merging; a bottleneck attends at the coarsest scale; four symmetric decoder stages then perform patch expansion while concatenating the corresponding encoder activations; an expand-by-four layer followed by a 1×1 projection produces the RGB output. Three capacities are explored by crossing initial widths $256, 384, 512$ with depth patterns 2–4–6–2, 2–4–8–2 and 2–6–12–4, giving nine architectural variants.

Training uses the LOL-v1 split, both LOL-v2 subsets and the SID corpus; for SID only the darkest exposure of every scene is paired with its long-exposure reference and the official Part-1 / Part-2 division is kept for training and validation. All images are converted to linear $[0, 1]$, randomly flipped and rotated by multiples of 90°, then partitioned into non-overlapping 256×256 crops that serve as individual samples; evaluation runs on a single uncropped patch without test-time augmentation. Four supervision regimes are tested: the hybrid LYT objective, the six-term LPIPS-augmented loss of Eq. (11) with $\alpha_P \in 0.1, 0.2, 0.5$, pure MSE and the colour-space FN-loss of Feng et al [8]. In every

case AdamW starts at 1×10^{-4}, warms up linearly for five epochs, decays cosinely to 1×10^{-6}, applies weight-decay of 10^{-4}, clips the gradient norm to 1.0 and accumulates two mixed-precision micro-batches, yielding an effective batch of sixteen patches. Each run spans one hundred epochs and the checkpoint with the lowest mean validation loss over LOL-v1, LOL-v2-real and LOL-v2-synthetic is retained.

All experiments were run on a single NVIDIA RTX 4090. Mini-batch size was adjusted per model to saturate GPU memory; for the 512-channel backbone this meant a batch size of 1, which noticeably slowed iterative testing. Given the tight hardware and time budget – and the wish to cover nine capacities and four loss functions – some hyper-parameters (e.g. the LPIPS multiplier) were fixed to representative values instead of being exhaustively tuned. Access to stronger hardware would allow a broader sweep over embed width, window size and loss weights, leading to a more thoroughly optimised model.

3. Experimental results

In this section, we present extensive experimental validation of our proposed Swin-Unet-based method for low-light image enhancement. We systematically evaluated the performance impact of key architectural choices, different strategies for incorporating supplementary datasets, and various loss functions. To directly address the reviewer's concern and isolate sources of improvement, we conducted two complementary ablations: (i) with the architecture and data held fixed, we varied only the loss (MSE, FN-loss, LYT, and LPIPS-weighted variants); and (ii) with the loss and data held fixed, we varied only the architecture (embedding dimensions and transformer depths). The baseline for all comparisons was the original Swin-Unet model configuration with embedding dimension 512 and hierarchical depths of 2-4-8-2, which previously demonstrated promising results in similar vision tasks. The LOL-v1 and LOL-v2 datasets (both synthetic and real subsets) were utilized as primary benchmarks. We specifically investigated the impact of embedding dimensions and transformer depths, dataset integration strategies (particularly regarding the SID dataset), and diverse loss function formulations, including Mean Squared Error (MSE), FN-loss, LYT loss, and our proposed LPIPS-based perceptual loss function. The evaluation metrics used were Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR), commonly adopted standards for image enhancement assessment.

3.1. Comparative analysis

Initially, we focused on the effective use of the SID dataset within the training pipeline. Four distinct approaches were tested using the optimal Swin-Unet architecture (embedding dimension 512, depths 2-4-8-6) and LYT loss: (1) selecting the single darkest image per scene from SID, (2) selecting the three darkest images, (3) randomly choosing SID

Tab. 1. Comparison of SID dataset integration strategies using LYT loss.

SID Strategy	SSIM LOL-v1	PSNR LOL-v1	SSIM LOL-v2-real	PSNR LOL-v2-real	SSIM LOL-v2-synth	PSNR LOL-v2-synth
Single darkest	**0.829**	21.43	**0.834**	22.55	0.897	22.46
Three darkest	0.799	**23.16**	0.825	22.63	0.897	22.40
Random	0.772	22.53	0.810	22.58	**0.904**	**23.30**
No SID	0.780	22.13	0.826	**23.67**	0.902	22.61

Tab. 2. Effect of embedding dimensions and depths (LYT loss, single darkest SID).

Embed dim / depths	SSIM LOL-v1	PSNR LOL-v1	SSIM LOL-v2-real	PSNR LOL-v2-real	SSIM LOL-v2-synth	PSNR LOL-v2-synth
512 / 2-4-8-2	**0.829**	21.43	**0.834**	**22.55**	0.897	**22.46**
512 / 2-4-6-2	0.762	20.00	0.803	21.56	0.883	20.84
384 / 2-4-6-2	0.759	20.91	0.792	21.10	0.872	20.52
384 / 2-6-12-4	0.784	**21.47**	0.806	21.74	**0.895**	22.52

images, and (4) completely excluding SID. Table 1 summarizes these experiments, clearly indicating that leveraging the single darkest SID image achieved consistently superior results. This strategy yielded an SSIM = 0.829 and PSNR = 21.43 for LOL-v1, and SSIM = 0.834 and PSNR = 22.55 for LOL-v2-real, significantly outperforming alternative approaches.

The observed differences between SID usage strategies highlight that carefully selecting SID images based on luminance intensity notably improves performance and training stability. Because the loss and architecture were held fixed here, these gains are attributable to the data integration strategy rather than the perceptual loss choice. Random SID selection, although performing well on synthetic datasets, showed reduced consistency across real-world benchmarks.

We then explored varying model configurations by adjusting the embedding dimensions and transformer depths, again utilizing the optimal SID selection (single darkest image). We compared embedding dimensions of 384 and 512, and various depth configurations, specifically 2-4-6-2 and 2-6-12-4. As Table 2 demonstrates, significantly lower embedding dimensions (384) substantially decreased SSIM and PSNR values, indicating insufficient representational capacity. Thus, such configurations were excluded from further experiments.

Under a fixed loss (LYT) and data strategy, increasing architectural capacity from depths 2-4-6-2 to 2-4-8-2 at embed = 512 improved SSIM/PSNR by +0.067/+1.43 (LOL-v1), +0.031/+0.99 (LOL-v2-real), and +0.014/+1.62 (LOL-v2-synth). These deltas are larger than those observed when swapping perceptual losses under a fixed architecture (see below), indicating that most SSIM/PSNR gains stem from the architecture.

Next, we assessed several loss functions to determine their efficacy. Specifically, we compared MSE, FN-loss, LYT loss, and our perceptual LPIPS-based loss with varying LPIPS multipliers (0.1, 0.5, and 1.0). Results summarized in Table 3 illustrate that simpler loss functions such as MSE and FN-loss underperformed notably, with MSE consistently lowest due to its exclusive pixel-level error penalization, which leads to

Tab. 3. Performance comparison of different loss functions.

Loss Function	SSIM LOL-v1	PSNR LOL-v1	SSIM LOL-v2-real	PSNR LOL-v2-real	SSIM LOL-v2-synth	PSNR LOL-v2-synth
LYT	**0.829**	21.43	**0.834**	22.55	**0.897**	22.46
LPIPS (0.1)	0.827	**21.77**	0.826	**22.60**	**0.897**	22.42
LPIPS (0.5)	0.789	21.13	0.827	22.32	0.895	**22.58**
LPIPS (1.0)	0.789	21.05	0.799	20.46	0.871	21.72
FN-loss	0.798	21.41	0.809	21.09	0.882	22.11
MSE	0.675	19.27	0.722	18.12	0.832	19.00

Tab. 4. NIQE and BRISQUE scores for the four loss functions (lower is better).

Loss	Dataset	NIQE	BRISQUE
MSE	LOL-v1	5.20	19.26
MSE	LOL-v2-real	5.46	20.90
MSE	LOL-v2-synth	5.02	15.84
FN-Loss	LOL-v1	7.14	22.56
FN-Loss	LOL-v2-real	7.36	25.78
FN-Loss	LOL-v2-synth	6.30	17.36
LYT	LOL-v1	5.79	15.36
LYT	LOL-v2-real	6.16	18.00
LYT	LOL-v2-synth	5.85	16.42
LPIPS	LOL-v1	5.55	17.18
LPIPS	LOL-v2-real	5.97	19.23
LPIPS	LOL-v2-synth	5.58	16.08

overly smooth and detail-deficient images. Conversely, LYT and LPIPS-based losses yielded the highest results, largely attributed to their composite nature – incorporating pixel-wise accuracy, perceptual quality, structural similarity, and colour fidelity, thus better aligning with human visual preferences.

With the architecture held constant (embed = 512, depths 2-4-8-2) and the same data strategy, LPIPS at a small weight (0.1) slightly increased PSNR relative to LYT while keeping SSIM essentially unchanged: +0.34 dB / −0.002 (LOL-v1) and +0.05 dB / −0.008 (LOL-v2-real); results on LOL-v2-synth were virtually tied (−0.04 dB / 0.000). Heavier LPIPS weights (0.5–1.0) reduced effectiveness, emphasizing the importance of balancing perceptual and pixel-level constraints. These comparisons show that while architectural capacity dominates fidelity (SSIM/PSNR), a lightly weighted LPIPS term can nudge optimization toward slightly better PSNR without sacrificing SSIM.

The four representative checkpoints were re-evaluated with the no-reference perceptual metrics NIQE [23] and BRISQUE [22] (Tab. 4). NIQE measures the deviation of an image's natural-scene statistics from a model learned on pristine photographs, whereas BRISQUE regresses locally normalized luminance and contrast statistics to subjective quality scores. Lower values in both cases correspond to higher perceptual quality.

Across the entire evaluation spectrum, LYT and LPIPS deliver noticeably better NIQE and BRISQUE scores than the multi-component FN-Loss of Feng et al., combining L_1, edge, and perceptual terms in both sRGB and HVI colour spaces. LPIPS attains the lowest NIQE values among the perceptual objectives, whereas LYT secures the best BRISQUE on LOL-v1 and LOL-v2-real, with LPIPS edging ahead on the synthetic subset. Because the architecture was fixed in these comparisons, these perceptual gains can be attributed primarily to the loss design.

Surprisingly, the plain MSE loss performs very competitively – particularly on LOL-v2-synth, where it records the overall best NIQE of 5.02. This suggests that strict pixel fidelity can suppress subtle non-linear artefacts sometimes introduced by perceptual losses; such artefacts are often imperceptible to the human eye yet penalised by statistical quality metrics. In summary, perceptually driven losses (LYT and LPIPS) still provide clear gains over FN-Loss, but a well-tuned MSE baseline remains a strong contender when judged solely by no-reference measures.

Detailed training convergence (Fig. 1) shows that, under the same architecture, the LYT loss and LPIPS with weight 0.1 both stabilize training and maintain superior PSNR/SSIM across epochs, with LPIPS slightly stronger in later epochs. Increasing the LPIPS weight reduces effectiveness, underscoring the need to balance perceptual and pixel-level terms. FN-Loss converges more gradually but remains competitive, whereas MSE lags throughout. Convergence plateaus appear around epoch 90.

Taken together, the ablations make the source of possible improvements explicit: most SSIM/PSNR gains come from scaling the Swin-Unet architecture (e.g., up to +1.62 dB PSNR when increasing depth at embed = 512), while perceptual gains (NIQE) are predominantly induced by the LPIPS-based loss when the architecture is fixed. The best

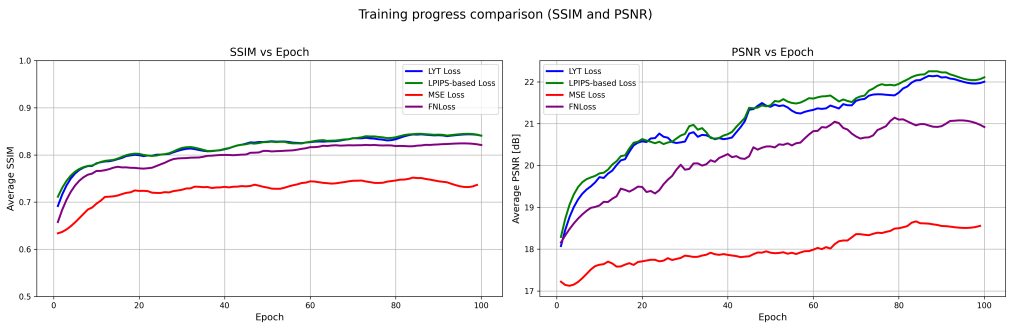

Fig. 1. Validation PSNR and SSIM versus training epochs. Legend: LYT Loss (blue), LPIPS-based loss with weight 0.1 (green), FN-Loss (purple), and MSE Loss (red). Curves are smoothed; metrics are computed after each epoch on the full validation set comprising SID (darkest exposure), LOL-v1, and the real and synthetic subsets of LOL-v2.

results arise from their combination – adequate model capacity paired with a modest LPIPS weight – yielding images that are both faithful and perceptually convincing.

These comprehensive results underscore the importance of model capacity, appropriate dataset integration, and carefully chosen composite loss functions in achieving high-quality, perceptually convincing low-light image enhancement; a visual comparison of our model's outputs with the reference images is provided in Figure 2.

On a per-dataset basis, holding the loss fixed (LYT) and increasing capacity from depths 2-4-6-2 to 2-4-8-2 at embed = 512 yields ΔPSNR/ΔSSIM of $+1.43/+0.067$ (LOL-v1), $+0.99/+0.031$ (LOL-v2-real), and $+1.62/+0.014$ (LOL-v2-synth). With the architecture fixed, LPIPS(0.1) improves NIQE vs. LYT by 0.24 (5.55 vs. 5.79, LOL-v1), 0.19 (5.97 vs. 6.16, LOL-v2-real), and 0.27 (5.58 vs. 5.85, LOL-v2-synth); BRISQUE favors LYT on real images (15.36 vs. 17.18; 18.00 vs. 19.23), while LPIPS is slightly better on synthetic (16.08 vs. 16.42). Although MSE attains a strong NIQE on LOL-v2-synth (5.02), it lags markedly in SSIM/PSNR across datasets. For data integration, selecting the single darkest SID exposure per scene is the most consistent strategy on real benchmarks; random selection can score higher on synthetic data but is less stable overall.

In practice, a compact recipe emerges: embed = 512 with depths 2-4-8-2, training on SID (single darkest) and a light LPIPS weight (0.1). Heavier LPIPS weights (0.5–1.0) reduce fidelity and stability, and convergence plateaus around epoch 90, after which early stopping is beneficial. Qualitatively (Fig. 2), this setting mitigates colour shifts and preserves edges, with only minor brightness deviations relative to ground truth.

3.2. Comparison with other algorithms

The quantitative comparison of our best-performing model – Swin-Unet trained with the proposed LPIPS-based loss function – is presented in Table 5. Although the model employing the LYT loss achieved similar performance, we prioritize the LPIPS-based approach as it introduces a novel perceptual component specifically tailored to low-light image enhancement. Furthermore, since the LPIPS-based loss was explicitly designed and proposed within this work, it more clearly represents our contributions.

From the results, it is evident that our Swin-Unet architecture achieves competitive but somewhat lower quantitative performance compared to state-of-the-art methods on all considered LOL datasets. Specifically, our best model achieved PSNR and SSIM of 21.77 dB and 0.827 on LOL-v1, 22.60 dB and 0.826 on LOL-v2-real, and 22.42 dB and 0.897 on LOL-v2-synthetic. In contrast, leading architectures such as CIDNet-oP [8], RetinexFormer [2], and LYT-Net [1] consistently surpass these metrics across all benchmarks, reaching PSNR values around 28 dB and SSIM over 0.88 in many cases.

These observed discrepancies may suggest that the Swin-Unet architecture – originally proposed for medical image segmentation – might not be optimal in capturing the

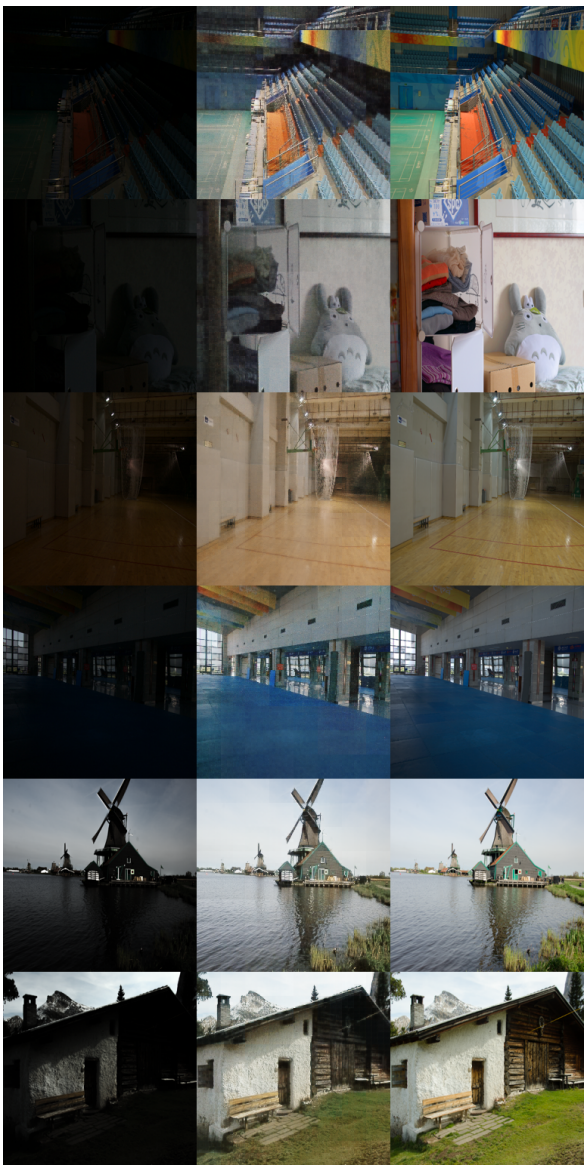

Fig. 2. Qualitative comparison layout and data sources. Columns: left – low-light inputs; centre – outputs from the model trained with an LPIPS-weighted loss; right – corresponding well-exposed ground-truth images. Rows: 1–2 from LOL-v1; 3–4 from LOL-v2-real; 5–6 from LOL-v2-synth. Images are randomly selected examples from the LOL family.

Tab. 5. Quantitative results on LOL datasets.

Methods	PSNR (LOL-v1)	SSIM (LOL-v1)	PSNR (LOL-v2-real)	SSIM (LOL-v2-real)	PSNR (LOL-v2-syn)	SSIM (LOL-v2-syn)
SID [4]	14.35	0.436	13.24	0.442	15.04	0.610
3DLUT [47]	21.35	0.585	20.19	0.745	22.17	0.854
Zero-DCE [11]	14.86	0.540	13.65	0.246	21.46	0.848
EnlightenGAN [15]	17.48	0.650	18.23	0.617	—	—
KinD [51]	20.87	0.800	20.40	0.652	16.26	0.591
KinD++ [49]	21.30	0.820	20.15	0.678	19.44	0.830
Bread [12]	22.96	0.840	22.54	0.762	19.28	0.831
IAT [6]	23.38	0.810	21.43	0.638	19.18	0.813
HWMNet [7]	24.24	0.850	22.40	0.622	18.79	0.817
LLFlow [35]	24.99	0.920	21.60	0.643	19.15	0.860
DeepUPE [33]	14.38	0.446	13.27	0.452	15.08	0.623
DeepLPF [24]	15.28	0.473	14.10	0.480	16.02	0.587
UFormer [36]	16.36	0.771	18.82	0.771	19.66	0.871
RetinexNet [37]	18.92	0.427	18.32	0.447	19.09	0.774
Sparse [42]	17.20	0.640	20.06	0.816	22.05	0.905
EnGAN [15]	20.00	0.691	18.23	0.617	16.57	0.734
FIDE [39]	18.27	0.665	16.85	0.678	15.20	0.612
Restormer [46]	26.68	0.853	26.12	0.853	25.43	0.859
LEDNet [53]	25.47	0.846	27.81	0.870	27.37	0.928
SNR-Aware [40]	26.72	0.851	27.21	0.871	27.79	0.941
LLFormer [34]	25.76	0.823	26.20	0.819	28.01	0.927
RetinexFormer [2]	27.14	0.850	27.69	0.856	28.99	0.939
CIDNet-wP [8]	27.72	0.876	**28.13**	**0.892**	29.37	**0.950**
CIDNet-oP [8]	**28.14**	0.889	27.76	0.881	**29.57**	**0.950**
A3DLUT [32]	14.77	0.458	18.19	0.745	18.92	0.838
IPT [5]	16.27	0.504	19.80	0.813	18.30	0.811
Band [41]	20.13	0.830	20.29	0.831	23.22	0.927
LPNet [18]	21.46	0.802	17.80	0.792	19.51	0.846
SNR [40]	24.61	0.842	21.48	0.849	24.14	0.928
LLIE [20]	25.24	0.855	25.94	0.854	27.79	0.941
PyDiff [52]	27.09	**0.930**	24.01	0.876	19.60	0.878
MIRNet [45]	26.52	0.856	27.17	0.865	25.96	0.898
LYT-Net [1]	27.23	0.853	27.80	0.873	29.38	0.940
Ours Swin-Unet (LPIPS-based)	21.77	0.827	22.60	0.826	22.42	0.897

specific features necessary for low-light image enhancement. However, despite somewhat lower quantitative results, the Swin-Unet architecture presents certain distinct advantages. Its pure transformer-based design effectively leverages global context modelling through self-attention mechanisms, enabling a strong representation of both local details and long-range dependencies simultaneously. Moreover, the architecture is relatively straightforward, highly modular, and significantly easier to train and fine-tune compared to more complex multi-stage architectures, such as those incorporating diffusion models or hybrid convolution-transformer networks.

Another key advantage of our model is computational efficiency and flexibility. While it is plausible that utilizing a larger-scale Swin-Unet network (e.g., deeper or wider variants) could potentially yield better quantitative performance, our experimental investigation was limited by available computational resources and time constraints. Therefore, an extensive exploration of larger models was beyond the scope of this work.

Nonetheless, the performance achieved demonstrates the viability and potential of the Swin-Unet approach – especially when paired with novel perceptual losses such as our LPIPS-based formulation. Given its favorable balance between complexity, computational efficiency, and respectable image enhancement quality, Swin-Unet remains an attractive candidate for further exploration, potentially yielding improved performance if scaled appropriately.

4. Conclusions and contributions

This study set out to verify whether a compact, single-stage Swin-Unet can remain competitive in extremely low-light conditions once supervision is shifted from purely pixel-based criteria to a perceptually oriented objective. The network we employed – an off-the-shelf Swin-Unet restricted to an embedding width of 512 and an encoder–decoder depth pattern of 2-4-8-2 – was purposefully kept small: with batch size one it already saturates the memory of a single RTX 4090, and shortening turnaround times was essential for running the nine-by-four grid of capacity-and-loss experiments reported throughout the paper. Within these resource limits several contributions emerge.

First, the composite loss that blends LPIPS, Smooth-L_1, MS-SSIM, inverted PSNR, colour mean and gradient consistency proves almost as effective as the far more elaborate LYT objective when both are applied to the same Swin-Unet backbone; on LOL-v1 and LOL-v2-real the two formulations reach virtually identical SSIM, while the LPIPS variant shows a slight PSNR advantage on two of the three benchmark splits. This confirms that loss design can close much of the perceptual gap even when architectural capacity is modest.

Second, the paper offers what is, to our knowledge, the first transformer-only baseline that covers LOL-v1, LOL-v2-real, LOL-v2-synthetic and SID under a single, fully documented training protocol; future work can therefore compare new transformer variants against numbers that are not confounded by convolutional extras or multi-branch tricks.

Third, the SID ablation confirms that keeping only the darkest exposure of each scene yields more dependable generalisation than either random or multi-exposure sampling – an observation that simplifies dataset preparation and, to our knowledge, had not been quantified before. The study also clarifies the limitations of our approach. Even the strongest configuration trails recent diffusion or multi-branch systems by roughly 5–6 dB in PSNR and a few hundredths in SSIM; visual inspection further reveals occasional smoothing of fine texture, most notably in areas dominated by read-noise. These deficits likely stem from choices that remained arbitrary because of limited time and compute – for example, the fixed LPIPS weight, the 7×7 shifted-window size, and the cap on embedding width. A wider sweep over those hyper-parameters, combined with experiments on deeper or broader Swin backbones, appears the most direct route to closing the remaining performance gap.

In short, although the model remains below the current state of the art, the study shows that a judiciously balanced perceptual loss can bring a compact Swin-Unet within striking distance of results obtained with far more elaborate objectives, establishes a clean transformer-only benchmark for future scaling studies, and uncovers a simple luminance-based strategy for sampling SID that reliably improves generalisation – insights that will help subsequent research allocate computational resources where they matter most.

References

[1] A. Brateanu, R. Balmez, A. Avram, C. Orhei, and C. Ancuti. LYT-NET: Lightweight YUV transformer-based network for low-light image enhancement. *IEEE Signal Processing Letters* 32:2065–2069, 2025. doi:10.1109/LSP.2025.3563125.

[2] Y. Cai, H. Bian, J. Lin, H. Wang, R. Timofte, et al. Retinexformer: One-stage Retinex-based transformer for low-light image enhancement. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12504–12513, 2023. doi:10.1109/ICCV51070.2023.01149.

[3] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, et al. Swin-Unet: Unet-like pure transformer for medical image segmentation. In: *Computer Vision – ECCV 2022 Workshops*, vol. 13803 of *Lecture Notes in Computer Science*, 2023. doi:10.1007/978-3-031-25066-8_9.

[4] C. Chen, Q. Chen, J. Xu, and V. Koltun. Learning to see in the dark. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pp. 3291–3300, 2018. doi:10.1109/CVPR.2018.00347.

[5] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, et al. Pre-trained image processing transformer. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, pp. 12299–12310, 2021. doi:10.1109/CVPR46437.2021.01212.

[6] Z. Cui, K. Li, L. Gu, S. Su, P. Gao, et al. You only need 90k parameters to adapt light: a light weight transformer for image enhancement and exposure correction. In: *33rd British Machine Vision Conference (BMVC 2022)*, 2022. https://bmvc2022.mpi-inf.mpg.de/238/.

[7] C.-M. Fan, T.-J. Liu, and K.-H. Liu. Half wavelet attention on M-Net+ for low-light image enhancement. In: *2022 IEEE International Conference on Image Processing (ICIP 2022)*, pp. 3878–3882, 2022. doi:10.1109/ICIP46576.2022.9897503.

[8] Y. Feng, C. Zhang, P. Wang, P. Wu, Q. Yan, et al. You only need one color space: An efficient network for low-light image enhancement. arXiv, arXiv:2402.05809, 2024. doi:10.48550/arXiv.2402.05809.

[9] X. Fu, Y. Liao, D. Zeng, Y. Huang, X.-P. Zhang, et al. A probabilistic method for image enhancement with simultaneous illumination and reflectance estimation. *IEEE Transactions on Image Processing* 24(12):4965–4977, 2015. doi:10.1109/TIP.2015.2474701.

[10] Z. Gu, F. Li, F. Fang, and G. Zhang. A novel Retinex-based fractional-order variational model for images with severely low light. *IEEE Transactions on Image Processing* 29:7233–7247, 2020. doi:10.1109/TIP.2019.2958144.

[11] C. Guo, C. Li, J. Guo, C. C. Loy, J. Hou, et al. Zero-reference deep curve estimation for low-light image enhancement. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pp. 1780–1789, 2020. doi:10.1109/CVPR42600.2020.00185.

[12] X. Guo and Q. Hu. Low-light image enhancement via breaking down the darkness. *International Journal of Computer Vision* 131:48–66, 2023. doi:10.1007/s11263-022-01667-9.

[13] H. Hou, Y. Hou, Y. Shi, B. Wei, and J. Xu. NLHD: A pixel-level non-local Retinex model for low-light image enhancement. arXiv, arXiv:2106.06971, 2021. doi:10.48550/arXiv.2106.06971.

[14] J. H. Jang, Y. Bae, and J. B. Ra. Contrast-enhanced fusion of multisensor images using subband-decomposed multiscale Retinex. *IEEE Transactions on Image Processing* 21(8):3479–3490, 2012. doi:10.1109/TIP.2012.2197014.

[15] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, et al. EnlightenGAN: Deep light enhancement without paired supervision. *IEEE Transactions on Image Processing* 30:2340–2349, 2021. doi:10.1109/TIP.2021.3051462.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, vol. 25, pp. 1097–1105, 2012. https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

[17] E. H. Land. The Retinex theory of color vision. *Scientific American* 237(6):108–128, 1977. doi:10.1038/scientificamerican1277-108.

[18] J. Li, J. Li, F. Fang, F. Li, and G. Zhang. Luminance-aware pyramid network for low-light image enhancement. *IEEE Transactions on Multimedia* 23:3153–3165, 2021. doi:10.1109/TMM.2020.3021243.

[19] R. Liu, L. Ma, J. Zhang, X. Fan, and Z. Luo. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, pp. 10561–10570, 2021. doi:10.1109/CVPR46437.2021.01042.

[20] Y. Liu, T. Huang, W. Dong, F. Wu, X. Li, et al. Low-light image enhancement with multi-stage residue quantization and brightness-aware attention. In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV 2023)*, pp. 12106–12115, 2023. doi:10.1109/ICCV51070.2023.01115.

[21] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, et al. Swin transformer V2: Scaling up capacity and resolution. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 11999–12009, 2022. doi:10.1109/CVPR52688.2022.01170.

[22] A. Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing* 21(12):4695–4708, 2012. doi:10.1109/TIP.2012.2214050.

[23] A. Mittal, R. Soundararajan, and A. C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters* 20(3):209–212, 2013. doi:10.1109/LSP.2012.2227726.

[24] S. Moran, P. Marza, S. McDonagh, S. Parisot, and G. Slabaugh. DeepLPF: Deep Local Parametric Filters for image enhancement. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pp. 12823–12832, 2020. doi:10.1109/CVPR42600.2020.01284.

[25] M. Noyan, A. R. Gostipathy, R. Wightman, and P. Cuenca. timm PyTorch Image Models. In: Hugging Face, 2025. https://huggingface.co/timm.

[26] NVIDIA Corporation. NVIDIA cuDNN. In: NVIDIA DEVELOPER, 2025. https://developer.nvidia.com/cudnn.

[27] S. Park, S. Yu, B. Moon, S. Ko, and J.-I. Paik. Low-light image enhancement using variational optimization-based Retinex model. *IEEE Transactions on Consumer Electronics* 63(2):178–184, 2017. doi:10.1109/TCE.2017.014847.

[28] PyTorch. Previous PyTorch Versions, 2025. https://pytorch.org/get-started/previous-versions/.

[29] A. Rogozhnikov. Einops: Clear and reliable tensor manipulations with Einstein-like notation. In: *International Conference on Learning Representations (ICLR 2022)*, 2022. https://openreview.net/forum?id=oapKSVM2bcj.

[30] A. Rogozhnikov. einops, 2025. https://einops.rocks/.

[31] H. Shakibania, S. Raoufi, and H. Khotanlou. CDAN: Convolutional dense attention-guided network for low-light image enhancement. *Digital Signal Processing* 156:104802, 2025. doi:10.1016/j.dsp.2024.104802.

[32] A. Wang, Y. Li, J. Peng, Y. Ma, X. Wang, et al. Real-time image enhancer via learnable spatial-aware 3D lookup tables. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV 2021)*, pp. 2451–2460, 2021. doi:10.1109/ICCV48922.2021.00247.

[33] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, et al. Underexposed photo enhancement using deep illumination estimation. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, pp. 6842–6850, 2019. doi:10.1109/CVPR.2019.00701.

[34] T. Wang, K. Zhang, T. Shen, W. Luo, B. Stenger, et al. Ultra-high-definition low-light image enhancement: A benchmark and transformer-based method. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2023)*, vol. 37 no. 3, pp. 2654–2662, 2023. doi:10.1609/aaai.v37i3.25364.

[35] Y. Wang, R. Wan, W. Yang, H. Li, L.-P. Chau, et al. Low-light image enhancement with normalizing flow. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2022)*, vol. 36 no. 3, pp. 2604–2612, 2022. doi:10.1609/aaai.v36i3.20162.

[36] Z. Wang, X. Cun, J. Bao, W. Zhou, J. Liu, et al. Uformer: A general U-shaped transformer for image restoration. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 17662–17672, 2022. doi:10.1109/CVPR52688.2022.01716.

[37] C. Wei, W. Wang, W. Yang, and J. Liu. Deep Retinex decomposition for low-light enhancement. In: *Proceedings of the British Machine Vision Conference (BMVC 2018)*, 2018. https://bmva-archive.org.uk/bmvc/2018/contents/papers/0451.pdf.

[38] J. Wen, C. Wu, T. Zhang, Y. Yu, and P. Swierczynski. Self-reference deep adaptive curve estimation for low-light image enhancement. arXiv, arXiv:2308.08197, 2023. doi:10.48550/arXiv.2308.08197.

[39] K. Xu, X. Yang, B. Yin, and R. W. H. Lau. Learning to restore low-light images via decomposition-and-enhancement. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pp. 2278–2287, 2020. doi:10.1109/CVPR42600.2020.00235.

[40] X. Xu, R. Wang, C.-W. Fu, and J. Jia. Snr-aware low-light image enhancement. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 17737–17747, 2022. doi:10.1109/CVPR52688.2022.01719.

[41] W. Yang, S. Wang, Y. Fang, Y. Wang, and J. Liu. Band representation-based semi-supervised low-light image enhancement: Bridging the gap between signal fidelity and perceptual quality. *IEEE Transactions on Image Processing* 30:3461–3473, 2021. doi:10.1109/TIP.2021.3062184.

[42] W. Yang, W. Wang, H. Huang, S. Wang, and J. Liu. Sparse gradient regularized deep Retinex network for robust low-light image enhancement. *IEEE Transactions on Image Processing* 30:2072–2086, 2021. doi:10.1109/TIP.2021.3050850.

[43] X. Yi, H. Xu, H. Zhang, L. Tang, and J. Ma. Diff-Retinex: Rethinking low-light image enhancement with a generative diffusion model. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6725–6735, 2023. doi:10.1109/ICCV51070.2023.01130.

[44] D. You, J. Tao, Y. Zhang, and M. Zhang. Low-light image enhancement based on gray scale transformation and improved Retinex. *Infrared Technology (Hongwai Jishu)* 45(2):161–170, 2023.

[45] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, et al. Learning enriched features for real image restoration and enhancement. In: *European Conference on Computer Vision (ECCV 2020)*, vol. 12370 of *Lecture Notes in Computer Science*, pp. 492–511, 2020. doi:10.1007/978-3-030-58595-2_30.

[46] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, et al. Restormer: Efficient transformer for high-resolution image restoration. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 5718–5729, 2022. doi:10.1109/CVPR52688.2022.00564.

[47] H. Zeng, J. Cai, L. Li, Z. Cao, and L. Zhang. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44(4):2058–2073, 2022. doi:10.1109/TPAMI.2020.3005590.

[48] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pp. 586–595, 2018. doi:10.1109/CVPR.2018.00068.

[49] Y. Zhang, X. Guo, J. Ma, W. Liu, and J. Zhang. Beyond brightening low-light images. *International Journal of Computer Vision* 129(4):1013–1037, 2021. doi:10.1007/s11263-020-01407-x.

[50] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43(7):2480–2495, 2021. doi:arXiv:1812.10477.

[51] Y. Zhang, J. Zhang, and X. Guo. Kindling the darkness: A practical low-light image enhancer. In: *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pp. 1632–1640, 2019. doi:10.1145/3343031.3350926.

[52] D. Zhou, Z. Yang, and Y. Yang. Pyramid diffusion models for low-light image enhancement. arXiv, arXiv:2305.10028, 2023. doi:10.48550/arXiv.2305.10028.

[53] S. Zhou, C. Li, and C. C. Loy. LEDNet: Joint low-light enhancement and deblurring in the dark. In: *European Conference on Computer Vision (ECCV)*, vol. 13666 of *Lecture Notes in Computer Science*, pp. 573–589, 2022. doi:10.1007/978-3-031-20068-7_33.