Low-Light Image Enhancement with U-Nets

Shayan Ravari [1] **Yuxi Luo** [1]

Abstract

Low-light images often suffer from poor visibility and high noise levels, hindering downstream computer vision tasks and human interpretation. In this work, we propose a U-Net-based convolutional neural network (CNN) for enhancing low-light images. A CNN is a class of deep neural networks particularly effective at extracting hierarchical features from image data, while the U-Net architecture extends this concept through an encoder-decoder design with skip connections. These skip connections preserve fine spatial details lost during downsampling, enabling more accurate reconstructions. We further integrate an attention mechanism to focus on noisy or underexposed regions and incorporate a combined L1-SSIM loss function to retain structural fidelity and color information.

1. Introduction

Low-light image enhancement (LLIE) has recently become a critical task in computer vision, as images captured in dim environments frequently suffer from poor visibility, high noise levels, and color distortions. Such degradation not only diminishes human perception but also weakens the performance of automated vision systems that rely on clear, high-contrast imagery. Consequently, improving image quality under low-light conditions can have a far-reaching impact on diverse applications, including surveillance, autonomous driving at night, and healthcare imaging in poorly lit operating rooms.

Over the years, traditional approaches to low-light enhancement have included techniques such as histogram equalization (Abdullah-Al-Wadud et al., 2007), gamma correction (Jeon et al., 2024), and Retinex-based algorithms (Ren et al., 2020). While these methods can boost brightness and contrast to some degree, they often rely on handcrafted assumptions about illumination and noise. As a result, they struggle with complex lighting variations, risk amplifying noise, and demand extensive tuning of hyperparameters. More importantly, many of these methods lack the computational efficiency required for real-time scenarios or large-scale video processing, limiting their practicality in modern applications.

In contrast, deep learning offers an alternative that can learn complex mappings between low-light inputs and well-exposed outputs without relying on extensive manual adjustments. Convolutional neural networks (CNNs), in particular, have shown a remarkable capacity to capture multi-scale features and model both global and local contrast relationships. By using large training sets of paired or unpaired low/normal-light images, CNN-based solutions have been shown to robustly handle diverse lighting conditions, adapt to scene complexity, and process images at realistic speeds (Tao et al., 2017). Recent architectures such as U-Net (Ronneberger et al., 2015), GAN-based frameworks (Jiang et al., 2019), and attention-enhanced models (Woo et al., 2018) further push the envelope by preserving fine textures and reducing noise in underexposed regions.

In this paper, we propose a U-Net based deep learning model for low-light image enhancement. U-Net's encoder-decoder architecture, originally designed for biomedical image segmentation, has proven highly effective in tasks requiring both spatial detail preservation and contextual feature extraction. By using skip connections, our approach retains fine-grained structural information lost in conventional CNN-based models during downsampling, leading to more accurate and visually coherent image reconstructions. Additionally, we integrate an attention mechanism to selectively enhance underexposed regions while mitigating noise amplification. To further improve perceptual quality, we employ a hybrid loss function combining L1 loss with Structural Similarity Index Measure (SSIM), ensuring both pixel-wise accuracy and structural fidelity. Through extensive experimentation on benchmark datasets, we demonstrate that our method significantly improves visual quality and quantitatively outperforms baseline approaches in terms of PSNR and MAE metrics, confirming its effectiveness and robustness in enhancing low-light images.

[1]Department of Mathematics, University of California - Los Angeles, California, US. Correspondence to: Tingwei Meng <tingwei@math.ucla.edu>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

2. Background

2.1. Convolutional Neural Networks

A Convolutional Neural Network (CNN) can be viewed as a parametric function

$$\mathbf{\Phi}(\mathbf{x};\mathbf{\Theta}) = \mathcal{T}_L \circ \cdots \circ \mathcal{T}_1(\mathbf{x};\mathbf{\Theta}_1) \tag{1}$$

with parameters $\mathbf{\Theta} = \bigcup_{\ell=1}^{L} \mathbf{\Theta}_\ell$, where each layer $\mathcal{T}_\ell$, $\ell \in \{1,\dots,L\}$, may be a convolution followed by an activation σ or a pooling operation. Let $\mathbf{x}^{(0)} \in \mathbb{R}^{C_0 \times H_0 \times W_0}$ denote the input tensor, with height H_0, width W_0, and C_0 input channels. Suppose a convolutional layer $\mathcal{T}_\ell$ has C_{in} input channels and C_{out} output channels. Each filter has a kernel of size $k \times k$, so the learnable parameters are $\mathbf{w}_{\ell,k} \in \mathbb{R}^{C_{\text{in}} \times k \times k}$ for $k = 1,\dots,C_{\text{out}}$, plus biases $b_{\ell,k} \in \mathbb{R}$. The convolution operation with stride s and padding p is can be written as

$$\big(\mathcal{T}_\ell(\mathbf{x})\big)_{k,y,x} \;=\; \sigma\left(\sum_{c=1}^{C_{\text{in}}} \big(\mathbf{w}_{\ell,k,c} * \mathbf{x}_c\big)_{y,x} + b_{\ell,k}\right) \tag{2}$$

where $\sigma(\cdot)$ is a pointwise nonlinearity (*e.g.*, ReLU). Concretely, if $*$ denotes the discrete 2D convolution with stride s and padding p, then

$$\begin{aligned}(\mathbf{w}_{\ell,k,c} * \mathbf{x}_c)_{y,x} \;=\; &\sum_{u=0}^{k-1}\sum_{v=0}^{k-1} \mathbf{w}_{\ell,k,c}[u,v] \\ &\times\; \mathbf{x}_c\Big[s\,y - p + u,\; s\,x - p + v\Big]\end{aligned} \tag{3}$$

If indices fall outside the valid range, $\mathbf{x}_c[\cdots]$ is taken to be zero. This yields an output tensor of shape $C_{\text{out}} \times H_{\text{out}} \times W_{\text{out}}$, where

$$H_{\text{out}} = \left\lfloor \frac{H + 2p - k}{s} \right\rfloor + 1, \quad W_{\text{out}} = \left\lfloor \frac{W + 2p - k}{s} \right\rfloor + 1$$

To further downsample feature maps, many CNN architectures interleave convolutions with pooling operations. A common choice is max pooling with kernel size $k \times k$ and stride s, defined by

$$\big(\mathcal{T}_\ell(\mathbf{x})\big)_{c,y,x} \;=\; \max_{0 \le u,v < k} \mathbf{x}_c\big[s \cdot y + u,\; s \cdot x + v\big] \tag{4}$$

This reduces the spatial dimension, aggregating information over local neighborhoods. For average pooling, we replace the max with an arithmetic mean.

2.2. U-Nets

A U-Net is a specific encoder-decoder CNN designed for image-to-image tasks. Let $\mathbf{x}^{(0)} \in \mathbb{R}^{C_0 \times H_0 \times W_0}$ be the input, and let d denote the number of downsampling (encoder) stages. Each encoder stage $i \in \{1,\dots,d\}$ applies a convolution or a sequence of convolutions and a pooling operation, producing intermediate outputs

$$\mathbf{x}^{(i)} \;=\; \mathcal{T}_i^{\text{enc}}\big(\mathbf{x}^{(i-1)};\mathbf{\Theta}_i^{\text{enc}}\big) \;\in \mathbb{R}^{C_i \times H_i \times W_i}$$

where $H_i = \lfloor H_{i-1}/2 \rfloor$, $W_i = \lfloor W_{i-1}/2 \rfloor$ if pooling is stride 2. At the final encoder level, we obtain $\mathbf{x}^{(d)} \in \mathbb{R}^{C_d \times H_d \times W_d}$, which is refered to as the bottleneck. The decoder then reconstructs a full-resolution output through a sequence of upsampling operators. Define $\mathbf{z}^{(0)} = \mathbf{x}^{(d)}$. For each decoder stage $j \in \{1,\dots,d\}$, we upsample $\mathbf{z}^{(j-1)}$ and concatenate it with the corresponding encoder output called the skip connection $\mathbf{x}^{(d-j)}$:

$$\mathbf{z}^{(j)} \;=\; \mathcal{T}_j^{\text{dec}}\Big(\text{Up}\big(\mathbf{z}^{(j-1)}\big) \;\oplus\; \mathbf{x}^{(d-j)}\Big)$$

Here, $\oplus$ denotes channel-wise concatenation and Up is the upsampling operation used. In most cases, upsampling is done using transpose convolution; mathematically, a transpose convolution can be seen as the adjoint of a standard convolution. It expands resolution by inserting zeros between input positions (when stride > 1) and convolving with a flipped kernel, thereby "undoing" a lower-resolution convolution. By the final stage, $\mathbf{z}^{(d)} \in \mathbb{R}^{C_{\text{out}} \times H_0 \times W_0}$ has the same spatial shape as the original input assuming appropriate padding or alignment. The parameters $\mathbf{\Theta}_{1:d}^{\text{enc}} \cup \mathbf{\Theta}_{1:d}^{\text{dec}}$ comprise the entire U-Net. Skip connections preserve high-resolution features that might otherwise be lost due to repeated pooling, enabling the U-Net to accurately recover spatial details in segmentation, denoising, or enhancement tasks.

3. Dataset

We conduct our experiments using the LOL dataset (Chen Wei, 2018), a widely recognized benchmark for low-light image enhancement. The dataset contains 500 pairs of low-light images and their corresponding normal-light references. Each image pair is aligned, ensuring that illumination level is the principal difference between the two images.

This dataset is split into a testing set comprising of 15 low-light and normal-light images and a training set with 485 low-light and normal-light images where each low-light and normal light image $\mathbf{x}_i^{\text{low}}, \mathbf{x}_i^{\text{normal}} \in \mathbb{R}^{3 \times 400 \times 600}$. Such a configuration simplifies the evaluation of enhancement quality through pixel-wise metrics (e.g., mean squared error, SSIM, PSNR) between the predicted enhancements and the ground-truth normal-light references. Moreover, the strict alignment of image pairs removes extraneous variables related to misalignment or scene changes, allowing the learning task to focus on illumination correction.

4. Model

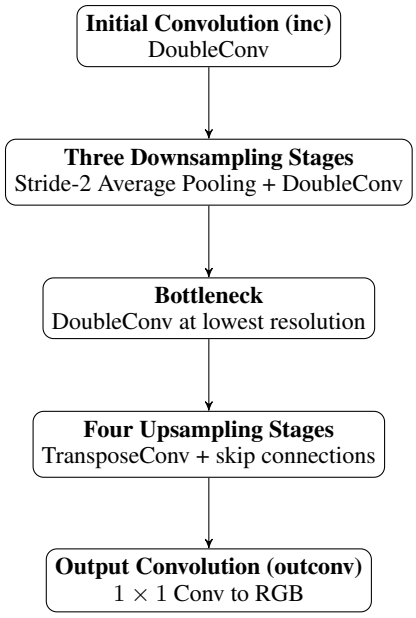
In this section, we provide a high-level description of our U-Net-based architecture for low-light image enhancement. While the background covers core concepts of convolutional networks and U-Nets, we now emphasize the novel aspects of our model design and the specific loss function choices that led to our best results.

4.1. Architecture

Our network follows a standard U-Net design but integrates double convolution (DoubleConv) blocks with Squeeze-and-Excitation (SE) modules adopted from (Hu et al., 2017) for channel-wise attention. Let $\mathbf{x} \in \mathbb{R}^{3\times H\times W}$ denote the input low-light image with three channels for RGB, and let

$$f_{\boldsymbol{\Theta}} : \mathbb{R}^{3\times H\times W} \;\to\; \mathbb{R}^{3\times H\times W}$$

be our parametric mapping, where $\boldsymbol{\Theta}$ collects all trainable parameters. The overall network has:



Concretely, if $\mathbf{x}^{(0)} = \mathbf{x}$ denotes the network input, we encode it into lower-resolution representations

$$\mathbf{x}^{(i)} = \mathcal{T}_i^{\text{enc}}\big(\mathbf{x}^{(i-1)};\, \boldsymbol{\Theta}_i^{\text{enc}}\big) \quad (i = 1, 2, 3)$$

pass through a bottleneck $\mathbf{x}^{(4)} = \mathbf{z}^{(0)}$, then decode with skip connections

$$\mathbf{z}^{(j)} = \mathcal{T}_j^{\text{dec}}\Big(\text{Up}(\mathbf{z}^{(j-1)}) \oplus \mathbf{x}^{(4-j)};\, \boldsymbol{\Theta}_j^{\text{dec}}\Big) \quad (j = 1, \ldots, 4)$$

where $\oplus$ denotes concatenation along the channel dimension. The final output $\mathbf{y} \in \mathbb{R}^{3\times H\times W}$ is obtained by applying a 1×1 convolution to $\mathbf{z}^{(4)}$ in order to reduce the channels to 3 for RGB output. In each DoubleConv block, we first apply two 3×3 convolutions with ReLU:

$$\mathbf{u} = \text{ReLU}\big(\mathbf{W}_2 * \text{ReLU}(\mathbf{W}_1 * \mathbf{z} + \mathbf{b}_1) + \mathbf{b}_2\big),$$

where $*$ is the 2D convolution operator. We then feed $\mathbf{u}$ into a Squeeze-and-Excitation mechanism that computes channel-wise scaling factors based on global average pooling. This process adaptively reweighs each channel before sending the output to the next layer. The inclusion of the SE block is particularly useful in our low-light enhancement task because it enables the network to focus on informative features and suppress less useful ones. By modeling the interdependencies among channels, the SE module improves the representational capacity of the network while adding only a modest increase in computational cost. This attention mechanism differentiates our approach from standard U-Net architectures and is one of the key modifications that lead to enhanced performance.

4.2. Loss Function and Training

We initially experimented with a linear combination of L1 and SSIM losses,

$$\mathcal{L} = \alpha \,\|\mathbf{y} - \mathbf{x}^{(\text{normal})}\|_1 + \beta\,\big[1 - \text{SSIM}(\mathbf{y}, \mathbf{x}^{(\text{normal})})\big],$$

where $\mathbf{x}^{(\text{normal})}$ is the ground-truth normal-light image, and $\text{SSIM}(\cdot,\cdot)$ is the structural similarity index (Wang et al., 2004). Over multiple trials, we observed that setting $\alpha = 0$ yielded the highest-quality enhancements, consistent with subjective visual inspection. Thus, our final loss became

$$\mathcal{L}_{\text{SSIM}} = 1 - \text{SSIM}\big(f_{\boldsymbol{\Theta}}(\mathbf{x}^{(\text{low})}),\, \mathbf{x}^{(\text{normal})}\big).$$

We minimize $\mathcal{L}_{\text{SSIM}}$ over mini-batches of size 3 using the Adam optimizer with a learning rate of 10^{-4}. Training proceeds for 10 epochs. During each epoch, the model sees only images in the training split; the separate test set remains completely withheld. We measure model performance via the peak signal-to-noise ratio (PSNR) and SSIM on the held-out set. Because the test set is never used for hyperparameter selection, these scores reflect a genuine estimation of the model's generalization ability to unseen low-light images.

5. Results

5.1. Qualitative/Visual Analysis

For illustration purposes, only one of the 15 images is chosen to be included in this report below since other enhanced images perform similarly in terms of visual enhancement effect. Figure 1 depicts an input image captured under low-light conditions, characterized by limited visibility and high noise levels. Figure 2 shows the enhanced output image generated by our model, clearly illustrating significant improvements in visibility, brightness, and structural clarity

compared to the low-light input image in Figure 1. The enhanced image closely resembles the normal-light ground truth, shown in Figure 3, which indicates the model's effectiveness in accurately restoring details and brightness.

Figure 1. Low Light Image

Figure 2. Enhanced Image

Figure 3. Normal Light Image

5.2. Quantitative/Numerical Analysis

Quantitatively, we evaluated our model's performance using the Training Loss, Mean Absolute Error (MAE, reported as Test L1), and Peak Signal-to-Noise Ratio (PSNR). Table 1 below summarizes the model's performance across training epochs on the training and validation datasets.

These metrics show a clear progression in training, with decreasing loss values and increasing PSNR, indicating effective learning and improved image reconstruction quality over epochs.

Figure 4 presents the evolution of training and validation losses during the training process. The convergence pattern indicates that the model successfully minimizes the combined loss function without significant overfitting, demonstrating stable learning.

Table 1. Training and testing metrics per epoch.

Epoch	Train Loss	Test L1	Test PSNR (dB)
1	0.3898	0.1473	16.11
2	0.2548	0.1256	17.57
3	0.2316	0.1131	18.28
4	0.2146	0.1129	18.24
5	0.2013	0.1176	17.84
6	0.1959	0.1160	17.97
7	0.1946	0.1251	17.45
8	0.1912	0.1170	18.04
9	0.1897	0.1158	18.18
10	0.1896	0.1102	18.68

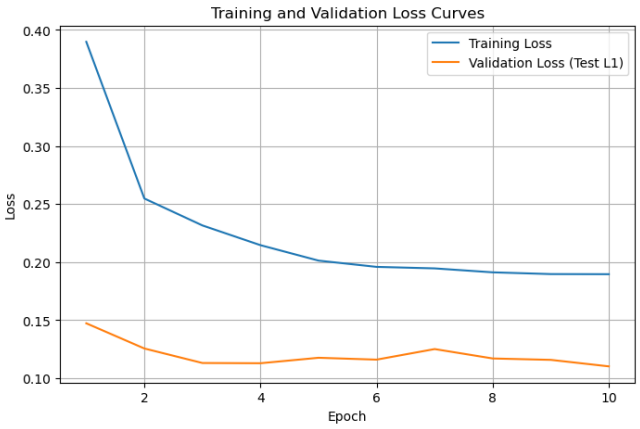


Figure 4. Training and validation loss curves over epochs

The code used for this project is available at the github link: github.com/shayanravari/MATH-156-Final-Project

6. Conclusion

In this project, we presented a U-Net-based model for low-light image enhancement that achieves promising results on most samples from our dataset. However, due to the relatively small dataset and the constraints of training 400×600 images over ten epochs, the model required significant time (around 20 minutes) to train and was not exhaustively tested on extreme low-light conditions. Future work could focus on increasing color saturation to address the somewhat dull hues in the output images. Additionally, extending the approach to low-light video enhancement would be a valuable practical direction, allowing real-world applications such as surveillance or mobile photography in challenging lighting.

References

Abdullah-Al-Wadud, M., Kabir, M. H., Dewan, M. A. A., and Chae, O. A dynamic histogram equalization for image contrast enhancement. *IEEE transactions on consumer electronics*, 53(2):593–600, 2007.

Chen Wei, Wenjing Wang, W. Y. J. L. Deep retinex decomposition for low-light enhancement. In *British Machine Vision Conference*, 2018.

Hu, J., Shen, L., and Sun, G. Squeeze-and-excitation networks. *CoRR*, abs/1709.01507, 2017.

Jeon, J. J., Park, J. Y., and Eom, I. K. Low-light image enhancement using gamma correction prior in mixed color spaces. *Pattern Recognition*, 146:110001, 2024. ISSN 0031-3203.

Jiang, Y., Gong, X., Liu, D., Cheng, Y., Fang, C., Shen, X., Yang, J., Zhou, P., and Wang, Z. Enlightengan: Deep light enhancement without paired supervision. *CoRR*, abs/1906.06972, 2019.

Ren, X., Yang, W., Cheng, W.-H., and Liu, J. Lr3m: Robust low-light enhancement via low-rank regularized retinex model. *IEEE Transactions on Image Processing*, 29: 5862–5876, 2020.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

Tao, L., Zhu, C., Xiang, G., Li, Y., Jia, H., and Xie, X. Llcnn: A convolutional neural network for low-light image enhancement. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–4, 2017.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.

Woo, S., Park, J., Lee, J., and Kweon, I. S. CBAM: convolutional block attention module. *CoRR*, abs/1807.06521, 2018.