Low-Light Image Enhancement Based on Generative Adversarial Network (GANs) Final Project Report

Mohammed Dishin

May 4, 2024

Abstract

Low-light image enhancement target to improve the perceptual quality of images captured under low-light conditions. Generative Adversarial Networks (GANs) have revolutionized the field of artificial intelligence by enabling the generation of realistic data samples from a given distribution. This paper proposes a novel generative adversarial network to enhance low-light image quality by designing a generator consisting of residual modules with hybrid attention modules and parallel dilated convolution modules. The residual module is designed to prevent gradient explosion during training and to avoid feature information loss. The hybrid attention module is designed to make the network pay more attention to useful features. A parallel dilated convolution module is designed to increase the receptive field and capture multi-scale information. Additionally, a skip connection is utilized to fuse shallow features with deep features to extract more effective features. Proposed solution consists of a discriminator to improve the discrimination ability. Improved loss function is proposed by incorporating pixel loss to effectively recover detailed information. The proposed method demonstrates superior performance in enhancing low-light images compared to seven other methods.

1 Introduction

Generative Adversarial Networks (GANs) have revolutionized the field of generative modeling, allowing machines to create realistic and diverse data like images, videos, and text. This report delves into the fascinating world of GANs, exploring their inner workings, architectures, training methods, formulations, applications, and challenges. GANs have achieved remarkable success in generating realistic data across various domains. They offer several advantages:

- Unsupervised Learning: GANs can learn without labeled data, making them suitable for situations where labeled data is scarce or expensive to obtain.
- High-Quality Generation: GANs can generate data that is often indistinguishable from real data, leading to various creative and practical applications.
- Flexibility: The GAN framework can be adapted to generate different types of data by modifying the architecture and training process.

2 Problem Statement

The problem statement addressed in this research is the enhancement of low-light images. The focus is on developing a method that can effectively improve the quality of images captured in low-light conditions by utilizing advanced network architectures, feature extraction techniques, attention mechanisms, and loss functions to outperform existing methods in terms of image quality metrics such as NIQE and BRISQUE This research aims to address the limitations in low-light image enhancement methods, such as noise interference, color information recovery, and overall image quality improvement. The proposed method focuses on enhancing low-light images by integrating a novel generative adversarial network to effectively improve the quality of the enhanced images.

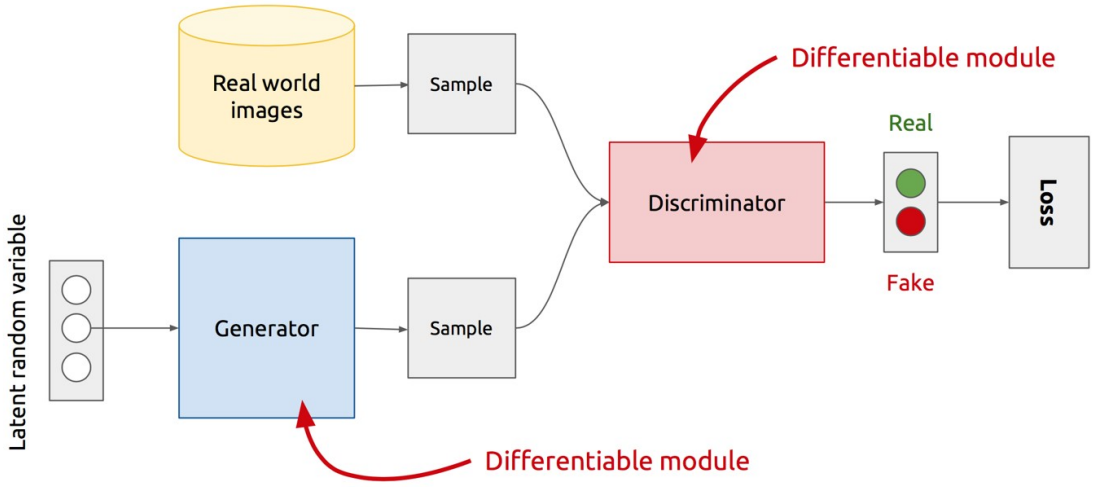

Figure 1: GANs Architecture.

3 Proposed Method

Generative Adversarial Networks (GANs) have revolutionized the field of generative modeling, allowing machines to create realistic and diverse data like images, videos, and text. This report delves into the fascinating world of GANs, exploring their inner workings, architectures, training methods, formulations, applications, and challenges. GANs have achieved remarkable success in generating realistic data across various domains. They offer several advantages:

- Unsupervised Learning: GANs can learn without labeled data, making them suitable for situations where labeled data is scarce or expensive to obtain.
- High-Quality Generation: GANs can generate data that is often indistinguishable from real data, leading to various creative and practical applications.
- Flexibility: The GAN framework can be adapted to generate different types of data by modifying the architecture and training process.

3.1 Model Architecture

3.1.1 Generator

A neural network that takes random noise as input and transforms it into the desired data format (e.g., images, text).

3.1.2 Discriminator

Another neural network that analyzes data and outputs a probability of whether it's real or generated. See Figure 1 in this section for an example.

3.1.3 How GANs Work

Imagine a two-player game: the generator and the discriminator. The generator creates new data, while the discriminator tries to determine if it's real or fake. This adversarial training refines both networks: the generator gets better at fooling the discriminator, and the discriminator becomes more adept at identifying fakes. Over time, the generator learns the underlying distribution of the real data, enabling it to create high-quality, realistic samples.

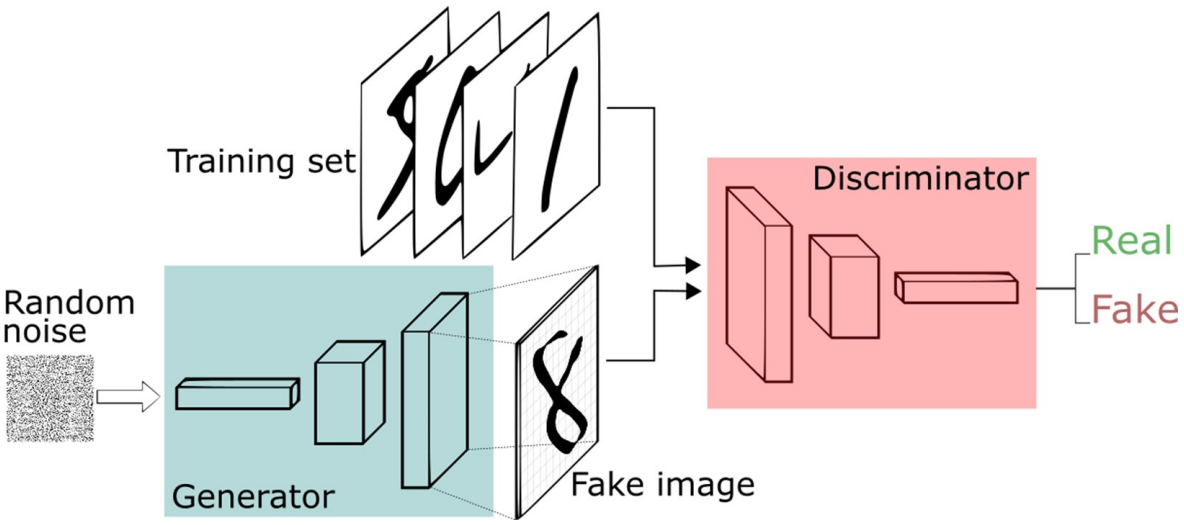

Figure 2: How GANs Works.

3.1.4 Model Training

Training GANs involves optimizing both the generator and discriminator simultaneously. Back-propagation is used to update the weights of each network based on the feedback received from the other. Various loss functions can be employed to measure the performance of each network and guide their individual training processes. See Figure 2. below are the steps for GANs model Training:

- Define the problem.
- Choose the architecture of GAN.
- Train Discriminator on the real data.
- Generate fake data for Generator.
- Train Discriminator on the fake data.
- Train Generator with the output of Discriminator.

3.1.5 GANs Formulation

The core idea behind GANs can be mathematically formulated using game theory concepts. The objective is to find a generator that minimizes the cost of fooling the discriminator, while the discriminator aims to maximize the cost of being fooled. This formulation allows for rigorous analysis and development of new GAN architectures. below are the formulation steps:

- Generator tries to fool Discriminator.
- Discriminator tries not to be fooled.
- Models are trained simultaneously.
- As Generator gets better, Discriminator has a more challenging task.
- As Discriminator gets better, Generator has a more challenging task.
- Discriminator role is to force Generator to work harder.

See Figure 3 for training of Discriminator and Figure 4 for training of Generator conceptual modeling.

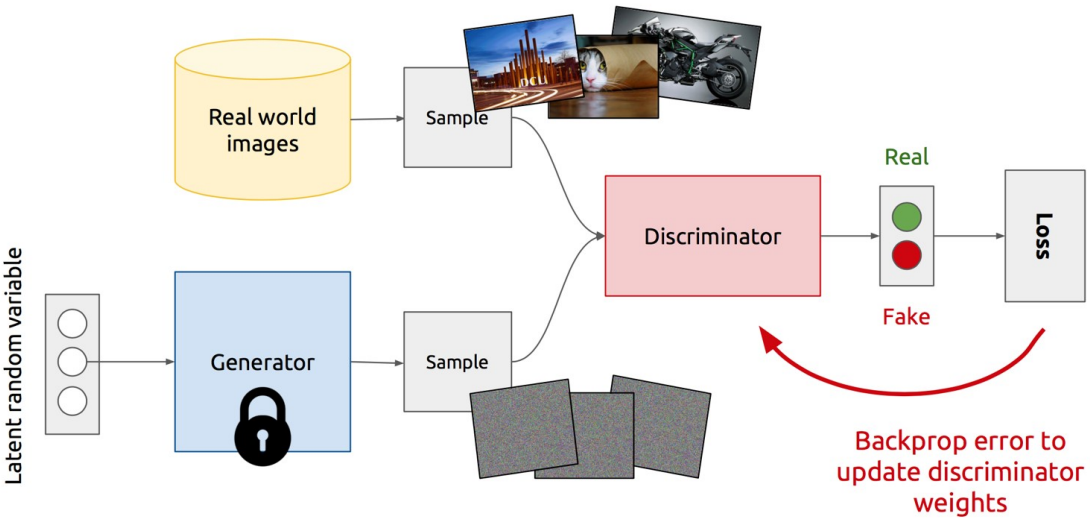

Figure 3: Training of Discriminator.

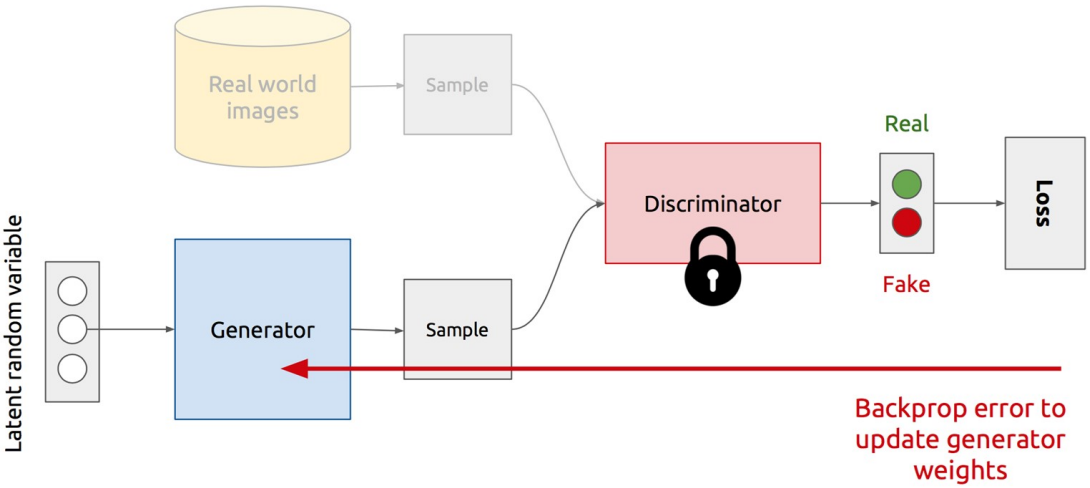

Figure 4: Training of Generator.

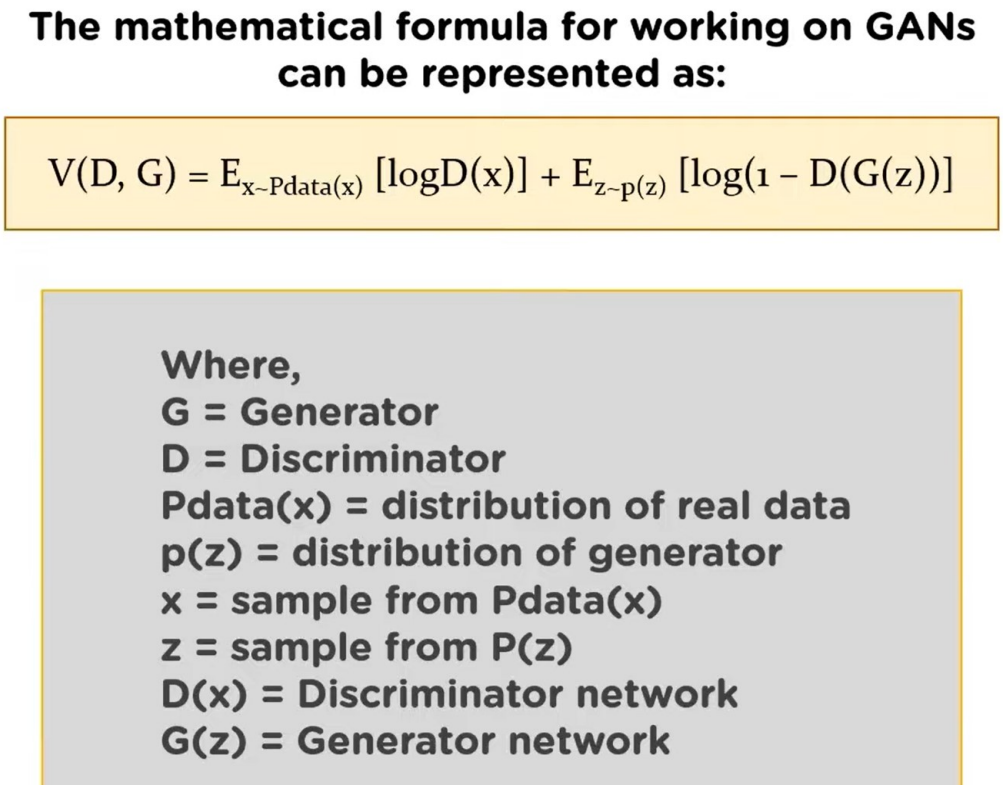

Figure 5: GANs Mathematical Representation.

3.1.6 GANs Mathematics Representation

See Figure 5 for mathematical representation showing how model works.

3.2 Types of GANs

The basic GAN architecture has been extended to address various challenges and applications. Some popular GAN variants include:

- Conditional GANs: These models incorporate additional information (e.g., labels, text descriptions) to guide the generation process.
- Wasserstein GANs: These GANs use a different loss function that improves training stability and addresses mode collapse issues.
- Generative Adversarial Autoencoders (GAEs): These models combine generative and autoencoder architectures, allowing for efficient data reconstruction and generation.

3.3 Applications of GANs

GANs have found applications in diverse fields such as:

- Image and Video Generation: Creating realistic images, editing existing ones, generating high-resolution versions of low-quality images.
- Natural Language Processing: Generating realistic text, translating languages, writing different kinds of creative content.
- Drug Discovery: Generating new drug molecules with desired properties.
- Material Science: Designing novel materials with specific characteristics.

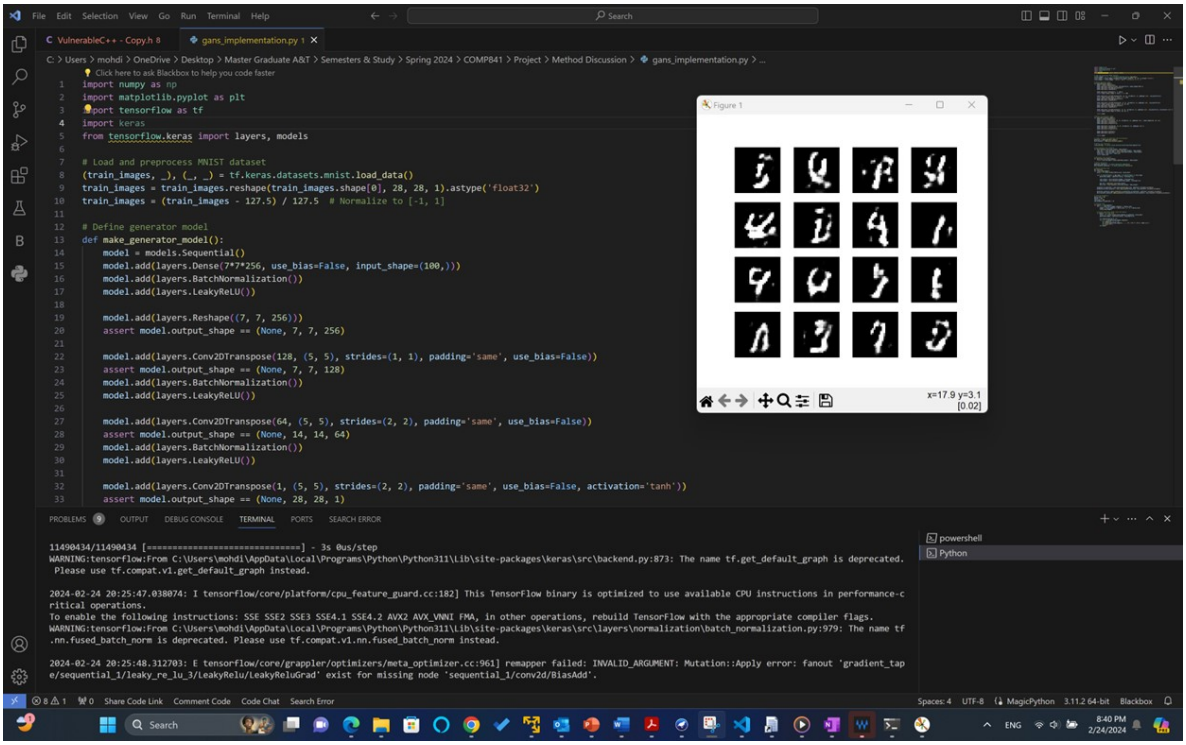

Figure 6: GANs Implementation on Local Machine.

3.4 Challenges with GANs

Despite their success, GANs face some challenges:

- Training Instability: Training GANs can be unstable, with the generator and discriminator potentially getting stuck in suboptimal states.
- Mode Collapse: The generator may converge to generating a single mode of data, neglecting the diversity present in the real data distribution.
- Evaluation Difficulties: Measuring the quality of generated data remains an open problem, making it harder to assess the true performance of GANs.

3.5 GANs Implementation and working Code

I implemented GANs model using Python, TensorFlow and Keras and below some results as following:

- I deployed the code on my local machine and training took around 5 hours to get some generated images, , see Figure 6.
- I deployed the code on Google Colab and I used GPU and TPU processing and training took around 3 hours to get some generated images, see Figure 7.
- Working code is attached with this report and you can find it as well on My Github on https://github.com/m-aljaili/GANs-model-implementation.git.

4 Related Work

4.1 Implemented Method

The method implemented in this paper to solve the problem of enhancing low-light images involves the design of a generative adversarial network (GAN) with specific components:

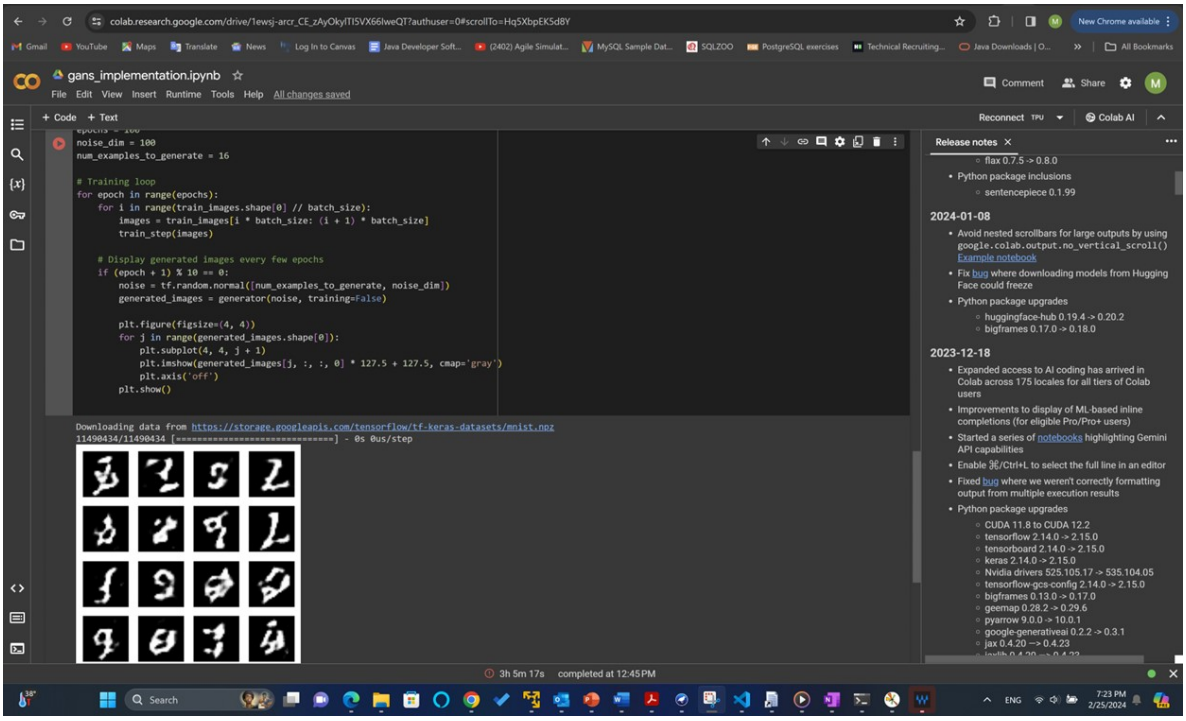

Figure 7: GANs Implementation on Google Colab.

- Residual Module: Used for low-frequency information extraction.
- Hybrid Attention Module: Incorporates attention mechanism for feature enhancement.
- Parallel Dilated Convolution Module: Designed to capture features at different scales.
- Loss Functions: Include adversarial loss, perceptual loss, and pixel loss for image enhancement.
- Training Procedure: Utilizes Adam optimizer and deep-learning framework PyTorch for model training.
- Evaluation Metrics: NIQE, SSIM, PSNR, and BRISQUE used to assess the quality of enhanced images.
- Comparison: Performance compared with other existing methods to demonstrate superiority in enhancing low-light images.

4.2 Data Preprocessing

The data preprocessing done in this paper involves the following steps:

1. Original Image Conversion:
   - Original low-light RGB images are converted into grayscale maps using the luminosity method.
2. Normalization:
   - The grayscale maps are normalized to the range to obtain luminance weight values for each pixel.
3. Attention Map Creation:
   - An attention map is generated by using the image obtained by subtracting the normalized gray scale map from 1.

- This attention map assigns weight values to pixels based on their brightness levels.

4. Image Enhancement Techniques:
   - Various image enhancement methods are applied to improve the quality of low-light images.
   - Techniques include Alpha-rooting, LIME, CycleGAN, Retinex-Net, EnlightenGAN, Zero-DCE, Zero-DCE++, and the proposed method.
5. Quantitative Evaluation:
   - The performance of these methods is evaluated on a full-reference dataset using metrics like PSNR, SSIM, NIQE, and BRISQUE.
   - The proposed method shows superior results in terms of these evaluation metrics compared to other methods.

4.3 Data type

The type of data used in this paper is primarily low-light images. These images serve as the input data for the image enhancement methods and algorithms discussed in the research. The data includes original low-light images, enhanced images generated by different methods, and normal light images for comparison. Additionally, the paper mentions the use of full-reference and no-reference datasets for evaluating the performance of the proposed method and comparing it with existing techniques. The data is used to assess image quality using metrics such as PSNR, SSIM, NIQE, and BRISQUE.

4.4 Model used for analyzing the data

The method implemented in this paper to solve the problem of enhancing low-light images involves the design of a generative adversarial network (GAN) with specific components:

- Residual Module: Used for low-frequency information extraction.
- Hybrid Attention Module: Incorporates attention mechanism for feature enhancement.
- Parallel Dilated Convolution Module: Designed to capture features at different scales.
- Loss Functions: Include adversarial loss, perceptual loss, and pixel loss for image enhancement.
- Training Procedure: Utilizes Adam optimizer and deep-learning framework PyTorch for model training.
- Evaluation Metrics: NIQE, SSIM, PSNR, and BRISQUE used to assess the quality of enhanced images.
- Comparison: Performance compared with other existing methods to demonstrate superiority in enhancing low-light images.

4.5 Comparison with other models

Comparisons were made with various existing models and methods to evaluate the performance of the proposed method for enhancing low-light images. The comparisons included the following:

- Evaluation Metrics: Performance comparison based on metrics such as PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity), NIQE (Natural Image Quality Evaluation), and BRISQUE (Blind/Referenceless Image Spatial Quality Evaluator).
- Quantitative Experiments: Tested different methods on full-reference datasets and compared the proposed method with other methods based on PSNR, SSIM, NIQE, and BRISQUE values.
- Module-wise Comparison: Assessed the impact of individual modules on the overall performance and compared the complete method with all modules against variations without specific components.

- Image Enhancement Techniques: Compared the proposed method with existing techniques such as Alpha-rooting, LIME, CycleGAN, Retinex-Net, EnlightenGAN, Zero-DCE, and Zero-DCE++.
- Overall Performance: Demonstrated that the proposed method outperformed other methods in terms of PSNR, SSIM, NIQE, and BRISQUE values, showcasing its effectiveness in enhancing low-light images.

below are the existing models used in comparison for enhancing low-light images:

- Alpha-rooting Method.
- LIME Method.
- CycleGAN Method.
- Retinex-Net Method.
- EnlightenGAN Method.
- Zero-DCE Method.
- Zero-DCE++ Method.

These methods were compared with the proposed method developed in the paper to evaluate and demonstrate the superiority of the proposed approach in enhancing low-light images. The comparison was based on various evaluation metrics such as PSNR, SSIM, NIQE, and BRISQUE to assess the quality and effectiveness of image enhancement techniques.

4.6 Study Related Work

In this study, conducted a comparative analysis of proposed method with seven low-light enhancement methods: the Alpha-rooting method, the LIME method, the CycleGAN method, the Retinex-Net method, the EnlightenGAN method, the Zero-DCE method, and the Zero-DCE++ method. The performance of these methods evaluated on both a no-reference image dataset and a full-reference image dataset. Used NIQE (Natural Image Quality Evaluation), SSIM (Structural Similarity), PSNR (Peak Signal-to-Noise Ratio), and BRISQUE (Blind/Referenceless Image Spatial Quality Evaluator) as evaluation metrics. In this experiments, the batch size was 4, and other parameters were set to zero to initialize the network parameters. For a fair comparison, used the same Adam optimizer and parameters b1 = 0.5 and b2 = 0.999 as used in EnlightenGAN to optimize the loss function of the network. The network was trained with 200 epochs, a learning rate of 0.0001 for the first 100 epochs, and a linear decrease to 0 for the next 100 epochs. The whole training process is described in Algorithm 1. Used the Ubuntu 18.04 system in this experiment. The GPU is an NVIDIA GeForce GTX 2080Ti. The deep-learning framework is PyTorch.

Algorithm 1. Training procedure for our proposed method.

1: **For** K epochs **do**
2: **For** k(k is a hyperparameter, $k = 1$) steps **do**
3: Sample minibatch of m low-light image samples $\{z^{(1)}, \ldots, z^{(m)}\}$ from low-light image domain.
4: Sample minibatch of m normal-light image samples $\{z^{(1)}, \ldots, z^{(m)}\}$ from normal-light image domain.
5: Update the discriminator by Adam Optimizer:

$$\nabla_D \left\{ \mathbb{E}[(D(x^{(i)}) - 1)]^2 + \mathbb{E}[D(G(z^{(i)}))]^2 \right\}$$

6: **End for**
7: Sample minibatch of m low-light image samples $\{z^{(1)}, \ldots, z^{(m)}\}$ fromlow-light image domain.
8: Update the generator by Adam Optimizer:

$$\nabla_G \left\{ \mathbb{E}[D(G(z^{(i)}))]^2 \right\}$$

9: **End for**

Figure 8: Model Training Algorithm

4.7 Model Architecture

4.8 Implemented Model Architecture

4.8.1 Final Implemented Generator Design

4.8.2 Final Implemented Discriminator Design

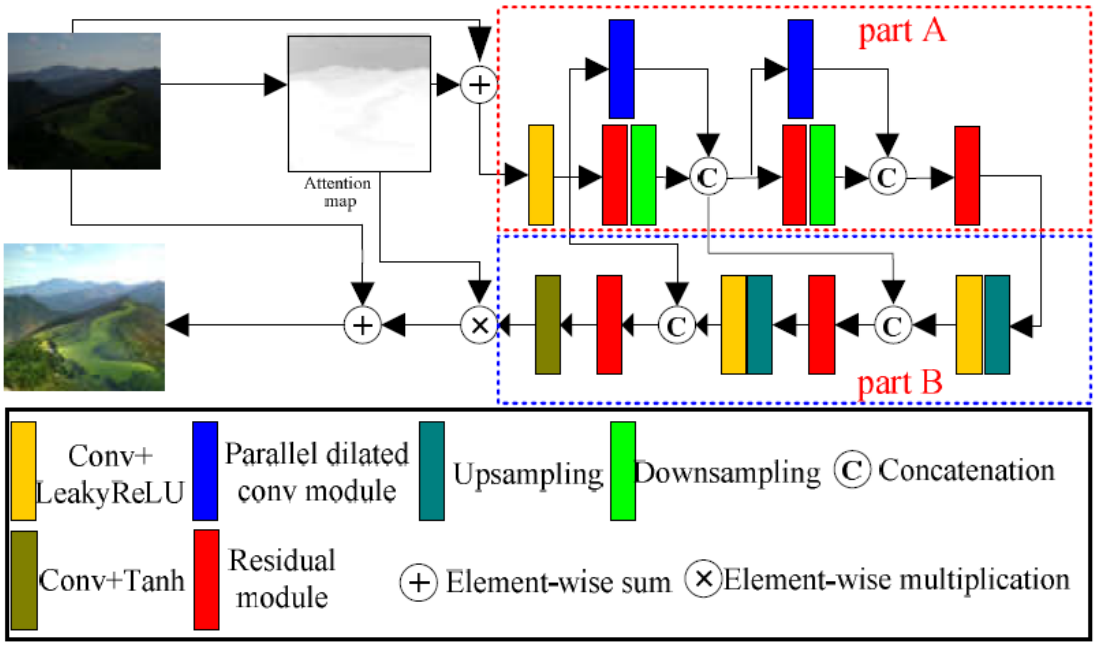

Figure 9: Proposed Generator: Part A) the top-down network, (Part B) the bottom-up network.

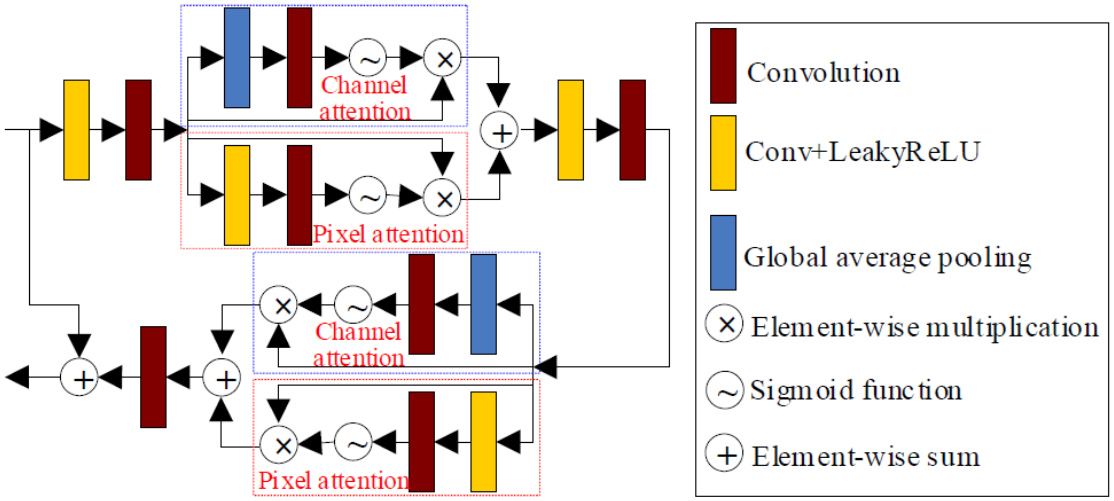

Figure 10: Proposed residual module with a hybrid attention mechanism.

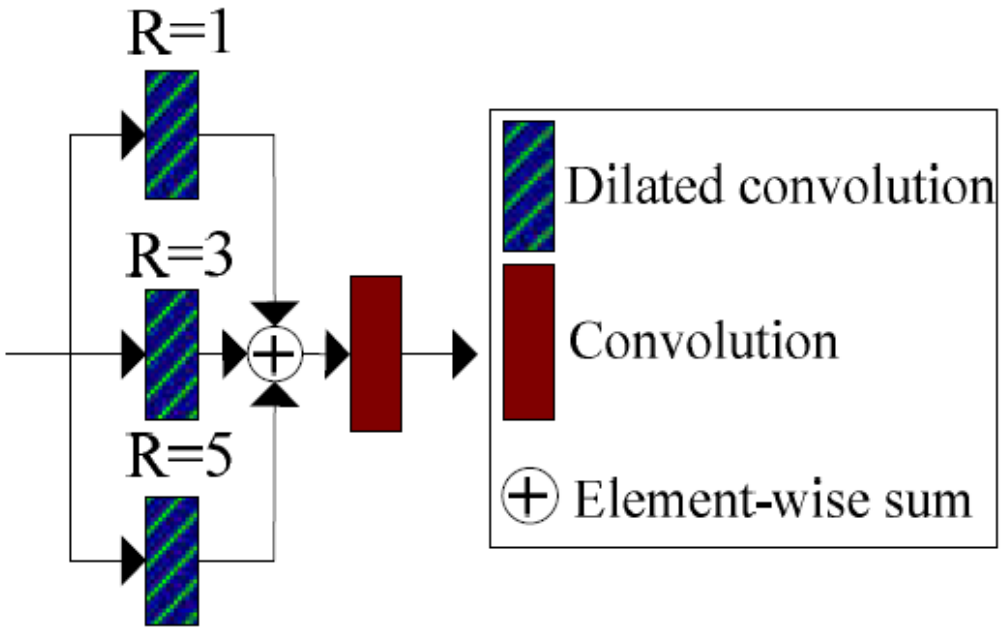

Figure 11: Proposed parallel dilated convolution module.

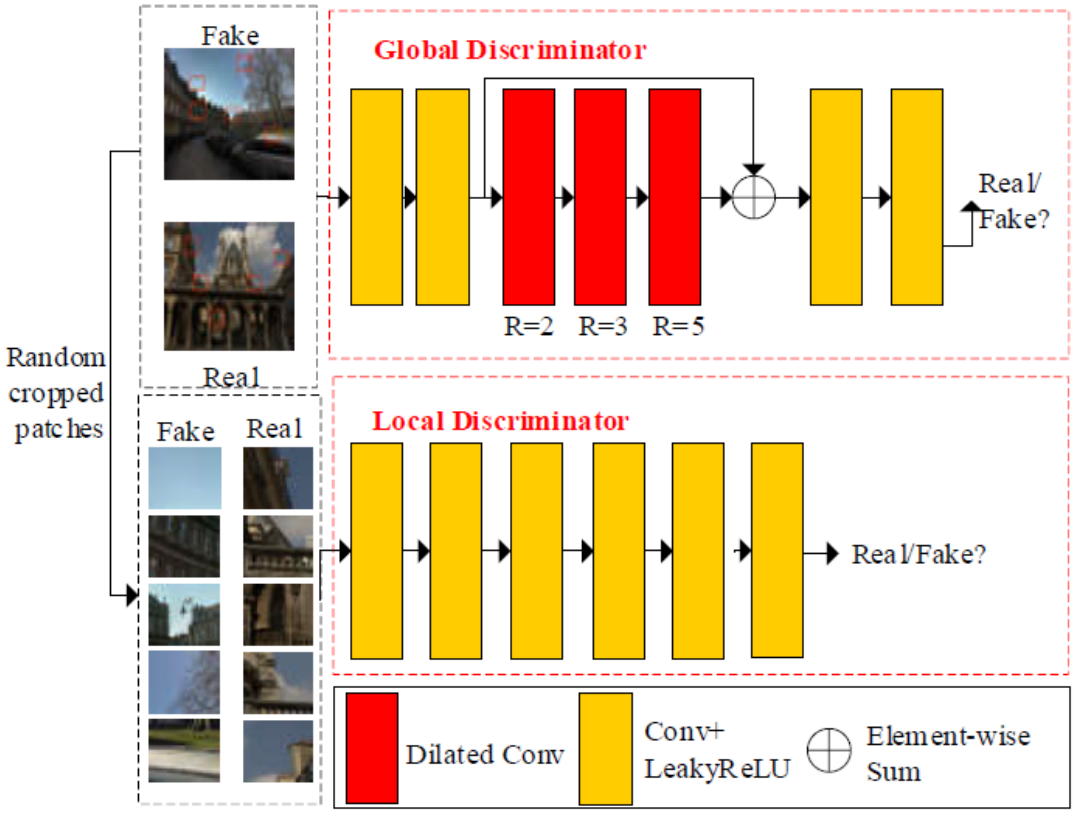

Figure 12: Proposed discriminator.

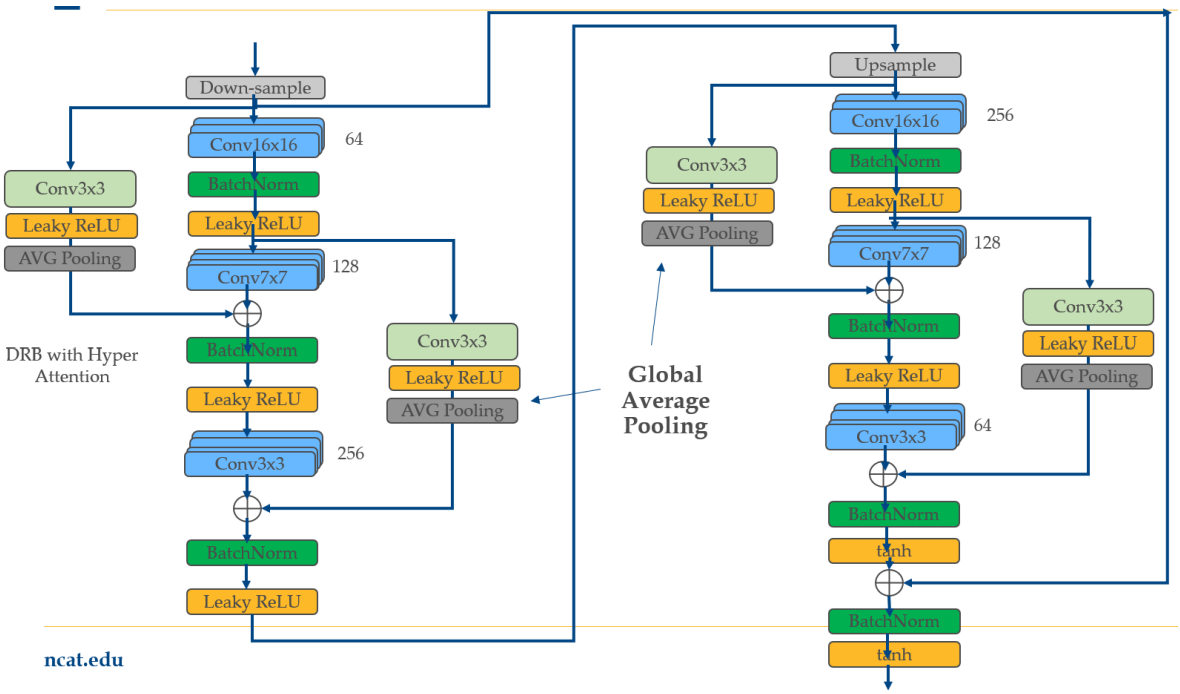

Figure 13: Final Implemented Generator.

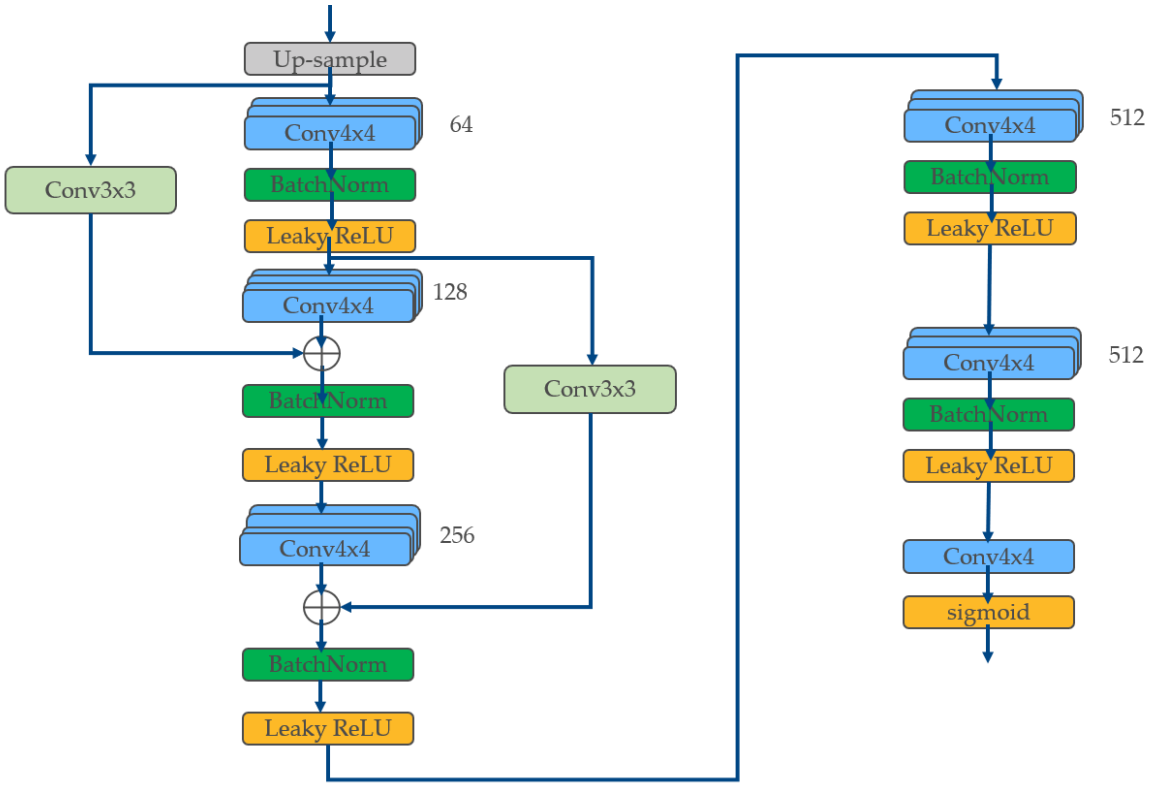

Figure 14: Final Implemented Discriminator.

	No_Attention	No_Parallel Dilated Conv	No_Cascaded Dilated Conv	No_Pixel Loss	Ours
PSNR	20.5918	23.3964	25.9701	26.1541	26.6451
SSIM	0.7686	0.7761	0.8667	0.8771	0.8817
NIQE	6.7748	5.2070	4.8803	4.9612	4.4719
BRISQUE	25.3088	22.2576	21.5149	40.0626	21.2218

Figure 15: Assessment results of each module.

	Alpha	LIME	CycleGAN	Retinex-Net	EnlightenGAN	Zero-DCE	Zero-DCE++	Ours
1st image	6.9561	6.6348	7.5064	12.4557	5.2253	8.8149	7.8973	5.1815
2nd image	7.1162	4.7842	5.8236	6.7309	3.6692	5.1637	4.5945	3.4426
3rd image	8.2354	8.3674	6.5934	10.6873	5.2263	6.3784	6.1283	5.1637
4th image	8.3671	6.3724	7.6354	11.3648	4.6992	4.4762	4.1164	3.9651
Average	7.6687	6.5397	6.8897	10.3097	4.7050	6.2083	5.6841	4.4382

Figure 16: NIQE values of four enhanced images by different methods.

5 Simulation and Discussion

5.1 Results

5.2 Final Training Results

5.3 Final Results on Test Data

5.4 Final Results Comparison

	Alpha	LIME	CycleGAN	Retinex-Net	EnlightenGAN	Zero-DCE	Zero-DCE++	Ours
1st image	42.6327	43.6249	38.6523	44.2361	30.4125	36.2578	33.6245	30.5261
2ndimage	41.2961	40.4375	40.1263	48.2763	29.1547	30.1542	28.7163	20.5238
3rd image	36.1284	37.9658	28.1267	38.1476	23.6321	29.3697	24.3698	22.9086
4th image	39.6183	50.5563	36.2548	53.2174	35.2147	40.1236	29.7211	25.4236
Average	39.9189	43.1461	35.7900	45.9694	29.6035	33.9763	29.1079	24.8455

Figure 17: BRISQUE values of four enhanced images by different methods.

	Alpha	LIME	CycleGAN	Retinex-Net	EnlightenGAN	Zero-DCE	Zero-DCE++	Ours
NIQE	8.8655	8.9884	8.8816	7.8564	5.2901	6.1564	5.7911	4.8656
BRISQUE	41.0368	40.3652	39.6235	41.3498	30.2563	28.6392	24.3687	23.6987

Figure 18: Average NIQE and BRISQUE values of enhanced images by different methods on a noreference dataset.

		Alpha	LIME	CycleGAN	Retinex-Net	EnlightenGAN	Zero-DCE	Zero-DCE++	Ours
1st image	PSNR	20.1381	19.8378	23.0964	15.9278	21.8685	15.9880	15.4431	24.1329
	SSIM	0.8032	0.7732	0.7944	0.7839	0.9213	0.8529	0.6098	0.9267
	NIQE	5.9932	6.9635	7.2511	7.5961	4.9968	5.0166	4.9888	4.7602
	BRISQUE	36.5827	34.5062	36.7823	28.1026	23.6384	26.8221	24.7335	23.0285
2nd image	PSNR	19.6382	17.9721	19.8103	13.2096	23.2018	16.5817	16.3813	27.3032
	SSIM	0.7886	0.7770	0.8515	0.7312	0.9208	0.8294	0.7320	0.9352
	NIQE	7.6631	8.3652	8.2236	6.2358	3.9624	4.9968	4.6379	3.8891
	BRISQUE	37.8260	36.5896	31.2014	24.1036	28.0457	28.0211	27.4125	24.1027
3rd image	PSNR	18.6357	17.3886	19.0584	15.8160	17.3886	17.2072	18.8322	20.5577
	SSIM	0.7081	0.6572	0.6837	0.6637	0.8802	0.7673	0.6311	0.8695
	NIQE	9.6635	7.6354	9.6102	5.1632	5.1022	5.8906	5.1063	4.9924
	BRISQUE	31.2569	30.1265	29.6321	29.3678	30.9519	37.2673	36.1859	28.6571
Average	PSNR	19.3764	18.3995	22.6550	14.9845	20.8196	16.5923	16.8855	23.9979
	SSIM	0.7815	0.7358	0.7765	0.7263	0.9074	0.8165	0.6576	0.9101
	NIQE	8.9657	7.6547	8.3616	6.3317	4.6871	5.3013	4.9110	4.5472
	BRISQUE	34.5632	33.7408	32.5386	29.1913	27.5453	30.7035	29.4440	25.2628

Figure 19: Performance index of three images enhanced by different methods.

	Alpha	LIME	CycleGAN	Retinex-Net	EnlightenGAN	Zero-DCE	Zero-DCE++	Ours
PSNR	17.3354	17.8337	21.1463	17.7947	23.9674	19.7008	18.8698	26.6451
SSIM	0.7022	0.6321	0.8322	0.6257	0.8640	0.7416	0.6463	0.8817
NIQE	8.9621	8.9673	8.1960	6.9928	4.8963	6.0023	5.3725	4.4719
BRISQUE	39.6477	40.3188	30.3485	34.5698	23.2056	29.4853	24.8423	21.2218

Figure 20: Performance index of all images enhanced by different methods.

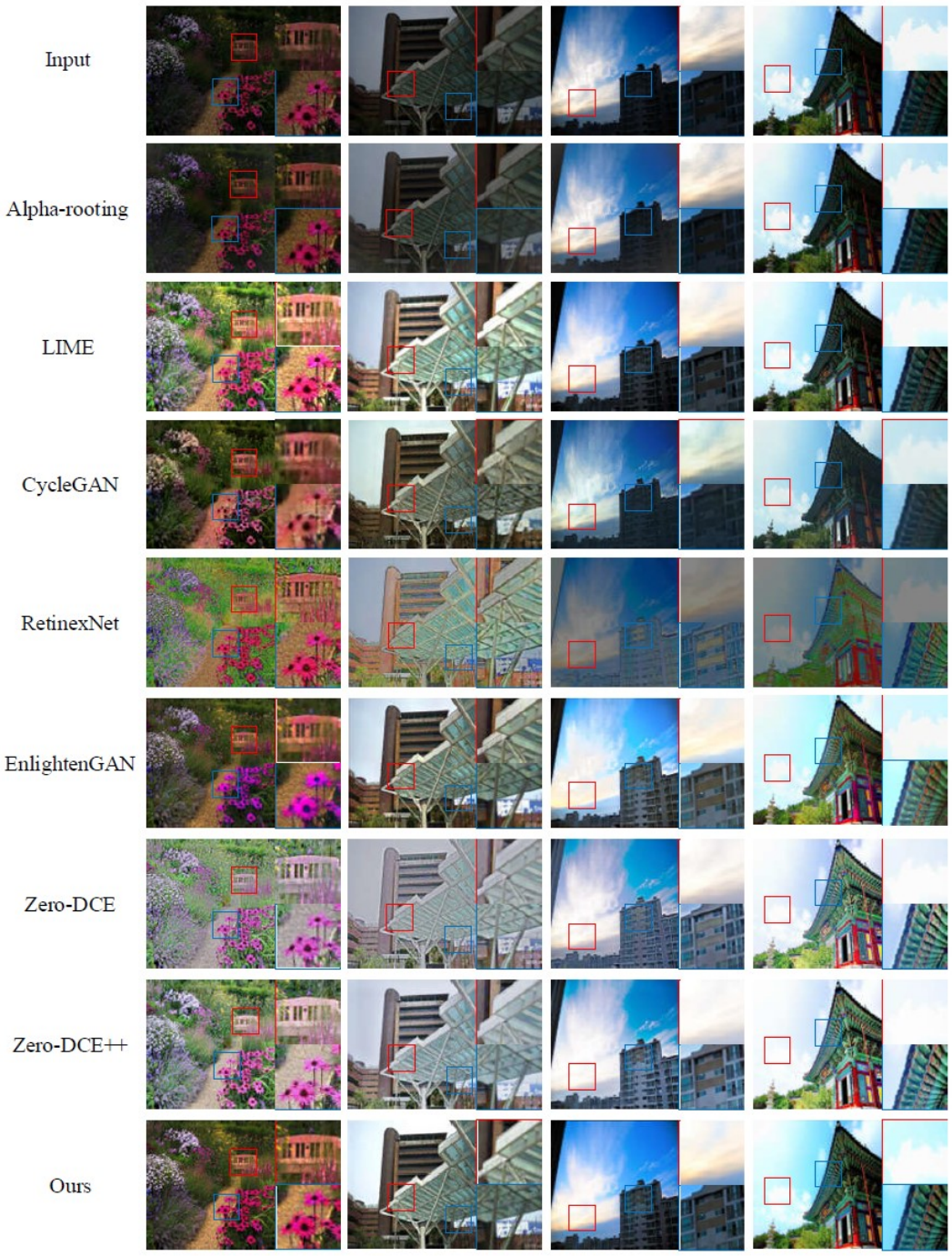

Figure 21: Low-light image enhancement results on the no-reference dataset.

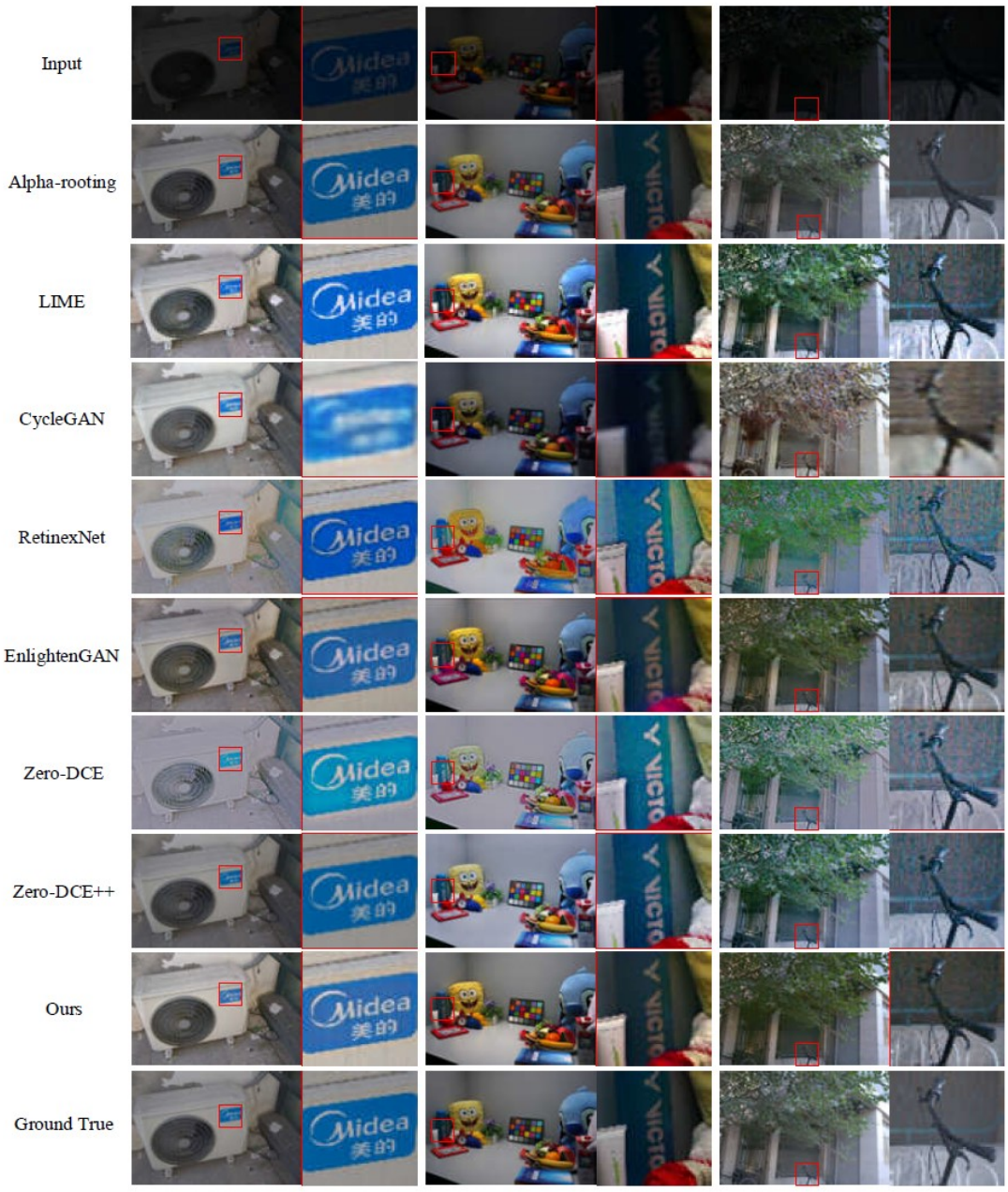

Figure 22: Low-light image enhancement results on the full-reference dataset.

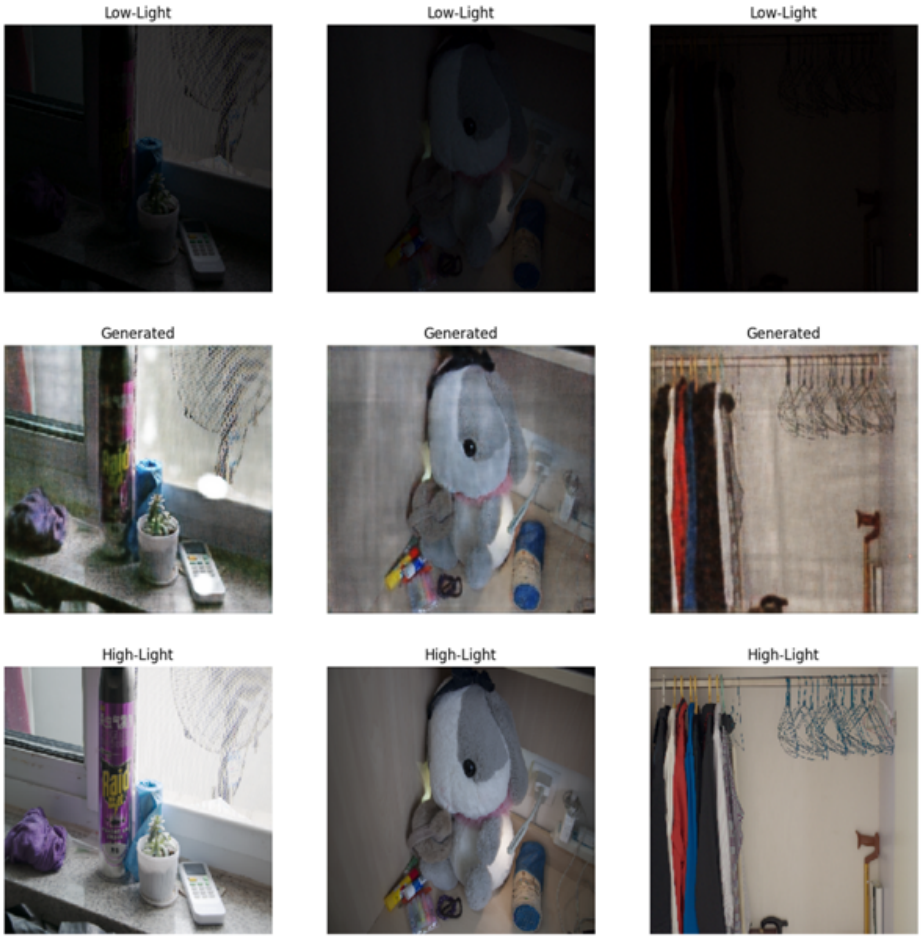

Figure 23: Final Training Result Using Training Data.

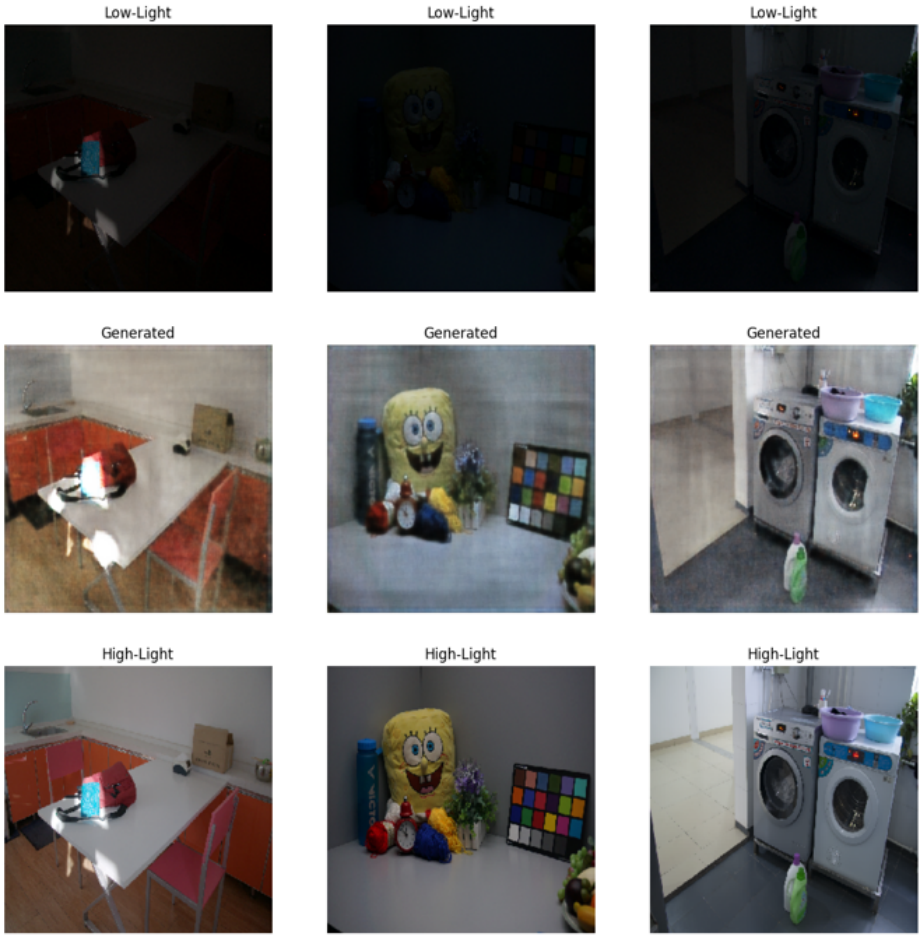

Figure 24: Final Result Using Test Data.

	Alpha	LIME	CycleGAN	Retinex-Net	EnlightenGAN	Zero-DCE	Zero-DCE++	Ours
PSNR	17.3354	17.8337	21.1463	17.7947	23.9674	19.7008	18.8698	26.6451
SSIM	0.7022	0.6321	0.8322	0.6257	0.8640	0.7416	0.6463	0.8817
NIQE	8.9621	8.9673	8.1960	6.9928	4.8963	6.0023	5.3725	4.4719
BRISQUE	39.6477	40.3188	30.3485	34.5698	23.2056	29.4853	24.8423	21.2218

Existing Results

My Results

		My Model
1st Design	PSNR	9.76
	SSIM	0.46
2nd Design With Improvement changes	PSNR	14.19
	SSIM	0.54
3rd Design With Improvement changes	PSNR	16.46
	SSIM	0.61
4th Design With Improvement changes	**PSNR**	**17.97**
	SSIM	**0.76**

- NIQE measures image quality without reference.
- **SSIM quantifies image similarity.**
- **PSNR measures signal-to-noise ratio.**
- BRISQUE evaluates spatial image quality without a reference.

Figure 25: Final Results Comparison.

6 Future Work

- Overall, the method combines advanced network architecture, novel loss functions, and thorough experimentation to develop an effective solution for enhancing low-light images, addressing the challenges of visibility, contrast, and overall image quality for various computer vision applications.
- Method achieved promising results and it could be enhanced more by applying parallel dilated convolution layers to enhance low-frequency information extraction for better image enhancement.
- Modify Generator network by incorporating short connections to connect the down-sampling and up-sampling networks.

7 Conclusion

Generative Adversarial Networks (GANs) represent a revolutionary approach to generative modeling, enabling the creation of realistic data samples across various domains. Despite facing challenges such as mode collapse and training instability, GANs continue to evolve, with new architectures and techniques continuously pushing the boundaries of what is possible. GANs represent a powerful tool for generating realistic and diverse data. While challenges remain, the active research in this area promises even more exciting developments and applications in the future and With their wide-ranging applications and ongoing advancements, GANs remain a focal point of research and innovation in the field of deep learning.

The conclusion drawn from the results is that the proposed method for enhancing low-light images outperforms existing methods in terms of image quality and performance. Key points from the conclusion include:

- Superior Performance: The proposed method achieved the highest values in PSNR and SSIM, and the smallest values in NIQE and BRISQUE compared to other existing methods.
- Module Importance: Each module proposed in the method plays a crucial role in enhancing low-light images, as demonstrated by the quantitative assessment results.
- Quantitative Experiments: Comprehensive experiments on full-reference datasets confirmed the superior performance of the proposed method over seven existing methods.
- Overall Effectiveness: The proposed method, with its integrated pixel loss function and GAN architecture, successfully recovered detailed information in enhanced images, showcasing its effectiveness in low-light image enhancement.
- Comparative Analysis: Comparisons with various existing models highlighted the strengths of the proposed method, emphasizing its ability to produce high-quality enhanced images with improved metrics across PSNR, SSIM, NIQE, and BRISQUE.

In summary, the results validate the effectiveness and superiority of the proposed method for enhancing low-light images, showcasing its potential for practical applications in image processing and enhancement tasks.

References

1. Dai, Y.; Liu, W. GL-YOLO-Lite: A Novel Lightweight Fallen Person Detection Model. Entropy 2023, 25, 587. [CrossRef] [PubMed].
2. Wang, H.; Chen, Y.; Cai, Y.; Chen, L.; Li, Y.; Sotelo, M.A.; Li, Z. SFNet-N: An Improved SFNet Algorithm for Semantic Segmentation of Low-Light Autonomous Driving Road Scenes. IEEE Trans. Intell. Transp. Syst. 2022, 23, 21405–21417. [CrossRef].
3. Lim, S.; Kim,W. DSLR: Deep Stacked Laplacian Restorer for Low-Light Image Enhancement. IEEE Trans. Multimed. 2021, 23, 4272–4284. [CrossRef].
4. Veluchamy, M.; Bhandari, A.K.; Subramani, B. Optimized Bezier Curve Based Intensity Mapping Scheme for Low Light Image Enhancement. IEEE Trans. Emerg. Topics Comput. 2021, 6, 602–612. [CrossRef].
5. Chen, Y.-S.; Wang, Y.-C.; Kao, M.-H.; Chuang, Y.-Y. Deep Photo Enhancer: Unpaired Learning for Image Enhancement from Photographs with GANs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018; pp. 6306–6314.
6. Shi, Y.;Wang, B.;Wu, X.; Zhu, M. Unsupervised Low-Light Image Enhancement by Extracting Structural Similarity and Color Consistency. IEEE Signal Process. Lett. 2022, 29, 997–1001. [CrossRef].