Indian Institute Of Technology Guwahati

" IMAGE-TO-IMAGE TRANSLATION FOR ROAD SCENES: A COMPARATIVE STUDY OF PIX2PIX AND CYCLEGAN "

A Intern Lab Project Report Submitted By

K Mahender

220102048

Under the Guidance of

Dr. Anirban Das Gupta

Indian Institute of Technology Guwahati

July 2025

Dept: Electronics and Communication Engineering

TABLE OF CONTENT :

Introduction :

In recent years, generative models—particularly Generative Adversarial Networks (GANs) have significantly advanced the field of image-to-image translation. This project explores the application of two powerful GAN architectures, Pix2Pix and CycleGAN, to the task of night-to-day image translation, focusing specifically on road scenes.

Road images captured at night often suffer from poor visibility, which can negatively impact tasks such as autonomous driving, traffic monitoring, and computer vision-based surveillance. Enhancing these images by converting them into realistic daylight versions can be incredibly useful in both practical and research contexts.

- Pix2Pix: A supervised image-to-image translation model that requires paired night and day images.
- CycleGAN: An unsupervised model capable of learning translations using unpaired datasets, making it more flexible when exact pairs are not available.

The models are trained and evaluated using a dataset of road scenes, with performance measured using both quantitative metrics (PSNR and SSIM) and qualitative visual comparisons. The goal is to determine which approach produces more realistic, high-quality day images from nighttime inputs.

Back Ground :

Generative Adversarial Networks (GANs) are a class of deep learning models introduced by Ian Goodfellow in 2014. They are composed of two neural networks — the Generator and the Discriminator — which are trained together in a game-theoretic setup.

Generator (G): Tries to create realistic fake data .
Discriminator (D): Tries to distinguish real data from fake data.

Pix2Pix is a supervised image-to-image translation model that requires paired training data, for example, matching night and day images of the same scene. It learns a direct mapping between input and target images using adversarial and L1 loss and works best when the input-output pairs are well-aligned.

DATASET

The dataset consists of two separate folders of images representing road scenes captured in two different lighting conditions:

1. /content/drive/MyDrive/Road_images → Contains daytime road images
2. /content/drive/MyDrive/Rainy_road_images → Contains nighttime road images

Each image is typically a color JPEG resized to a consistent shape (128×128 or 256×256 pixels) for training. The images capture highways, streets, and road scenes from various viewpoints, and are likely sourced from either a public dataset or collected manually.

Common Steps (Both Pix2Pix & CycleGAN)

- Import libraries and setup GPU
- Load and preprocess images (resize, normalize, augment)
- Build generator and discriminator models
- Train models with appropriate loss functions
- Save checkpoints and generated output

Pix2Pix (Supervised)

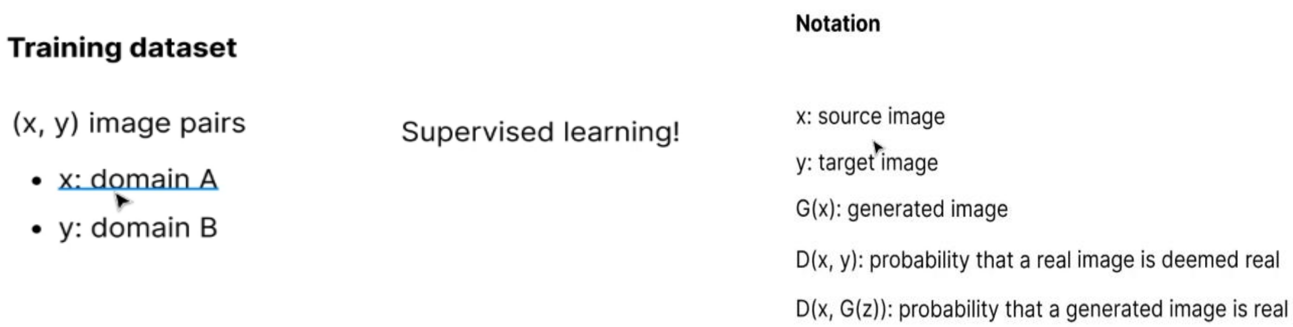
- Requires paired rainy ↔ clear images
- Uses 1 generator (rainy → clear) and 1 discriminator
- Loss: Adversarial + L1 loss
- Simpler and faster to train

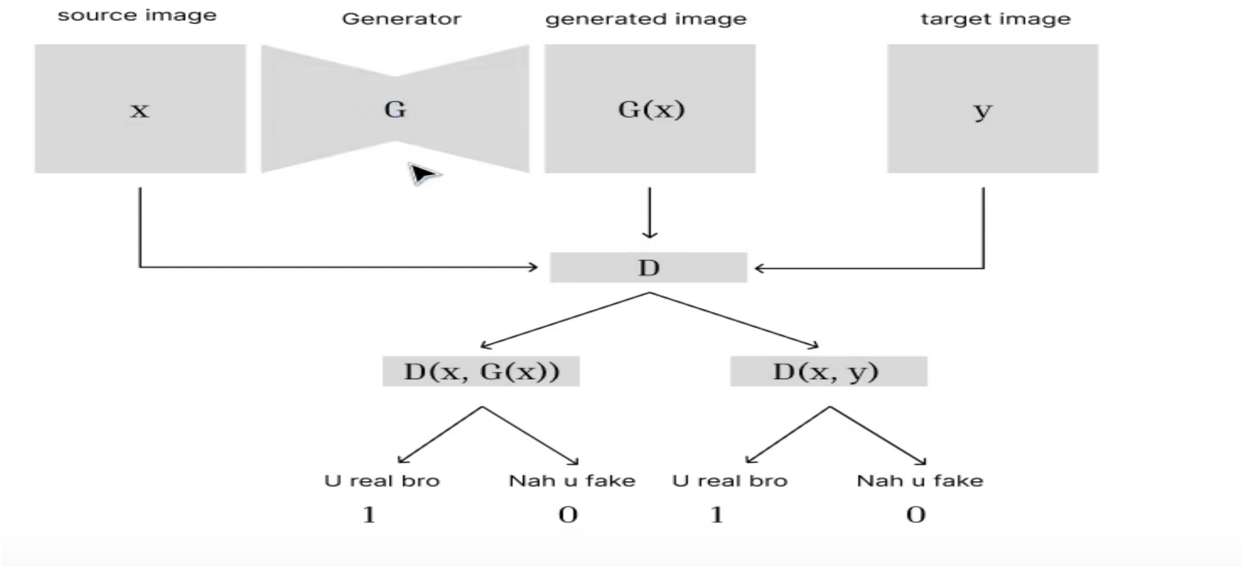
MODEL ARCHITECTURE

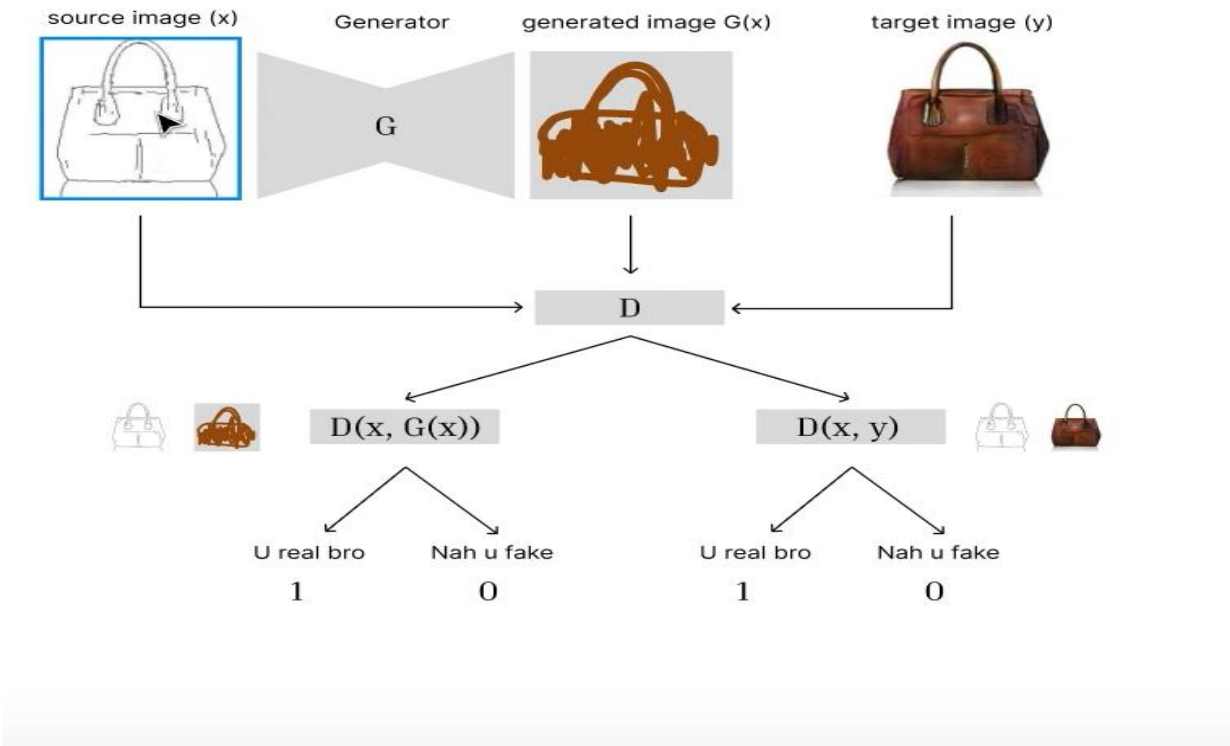
Pix2Pix Model

Generator:

- U-Net architecture
- Converts rainy images → clear image

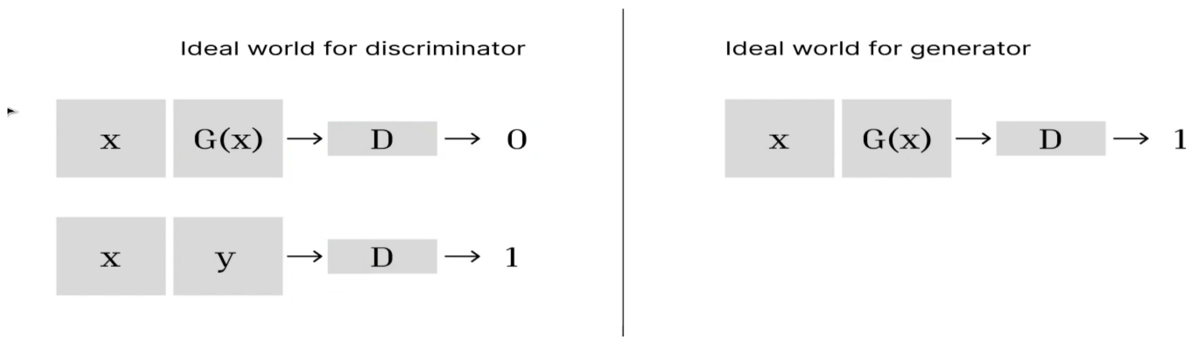






The input image is passed into a Generator (U-Net) that uses downsampling, upsampling, and skip connections to create a target-style image.

The Discriminator (PatchGAN) checks whether the generated image is real or fake by comparing it with the actual target image.



The pix2pix has double losses

1)

Adversarial loss

Generator and discriminator compete. Generator aims to fool discriminator, which aims to expertly distinguish between real and fake pairs.

$$L_{cGAN}(G, D) = log(D(x, y)) + log(1 - D(x, G(x)))$$

$log(D(x, y))$: log probability that (x, y) pair is real

- Discriminator wants to maximize this

$log(1 - D(x, G(x))$: log probability that (x, G(x)) pair is fake

- Discriminator wants to maximize this
- Generator wants to minimize this

2)

Structural loss

Ensures that generated images resemble the input image.

$$L_{L1}(G) = ||y - G(x)||_1$$

Final Objective

$$\arg\min_G \max_D L_{cGAN}(G, D) + \lambda L_{L1}(G)$$

IMPLEMENTATION

Tools and Libraries Used

Python – Main programming language

TensorFlow / Keras – Deep learning framework

NumPy – Numerical operations

Matplotlib – For visualizing images and training progress

Google Colab – For GPU-based model training

TQDM – Progress bars for training

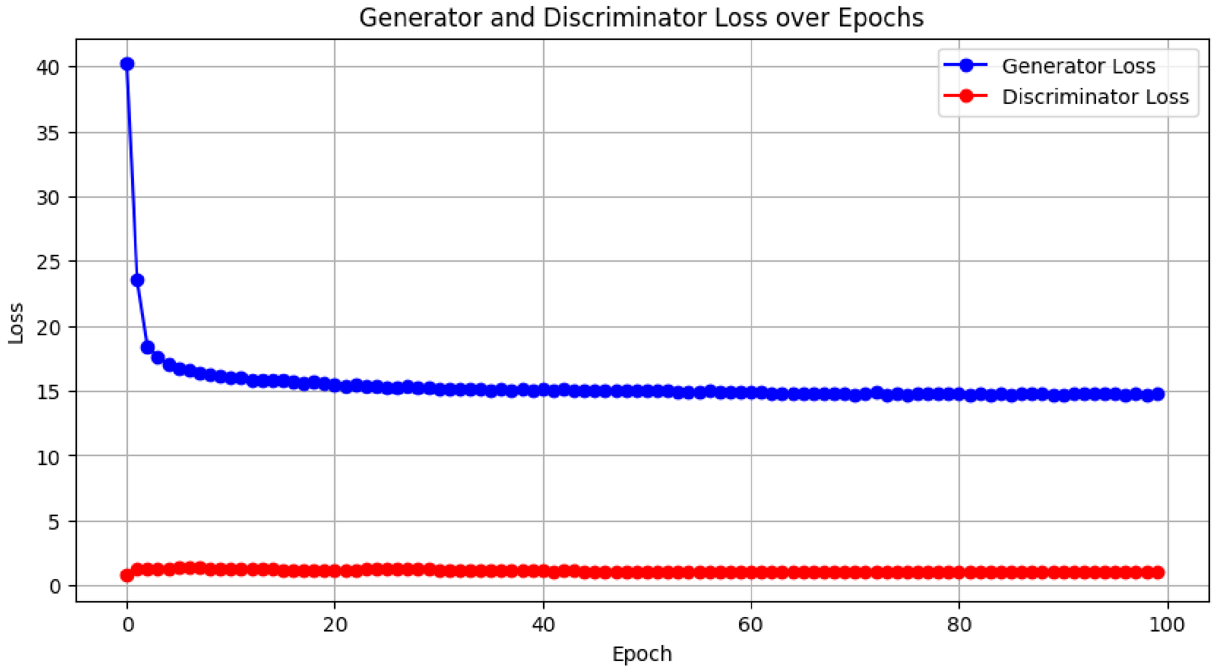
skimage – To calculate PSNR and SSIM

os / glob / PIL – File handling and image loading

RESULTS :

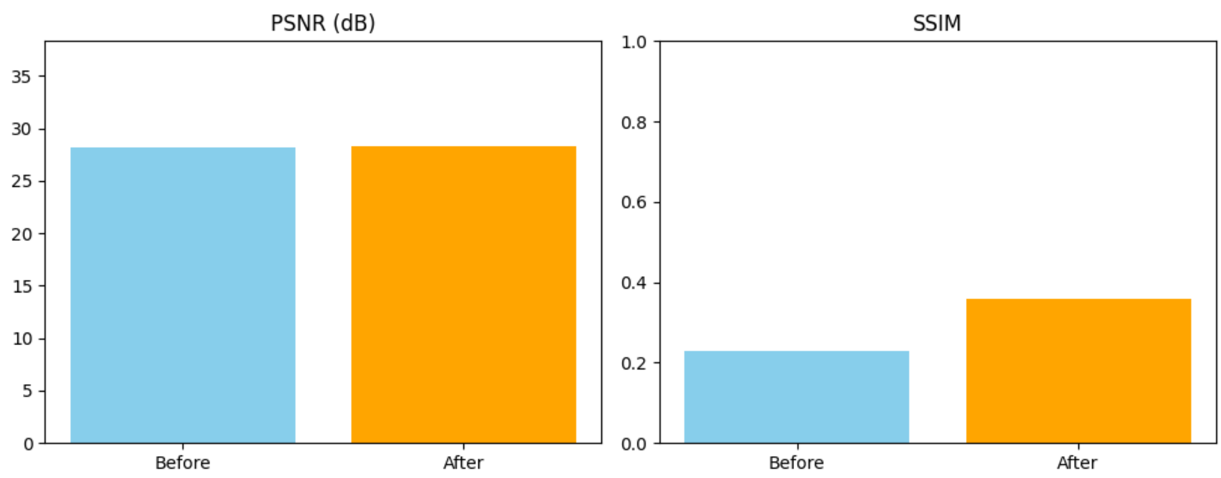
PIX2PIX Model :

Loss Functions



Average PSNR (Rainy→Clear) is 28.21 dB

Average SSIM (Rainy→Clear) is 0.3578



Conclusion

This project successfully demonstrated the application of Pix2Pix and CycleGAN models for transforming nighttime road images into clear daytime scenes using advanced image-to-image translation techniques.

Both models showed promising results:

- Pix2Pix produced sharp, accurate outputs due to its supervised training with paired images.
- CycleGAN performed remarkably well in handling unpaired datasets, learning stylistic transformations without direct supervision.

Through this implementation:

- We achieved good perceptual quality and structure in the generated outputs.
- Integrated features like checkpointing, resume training, and loss visualization improved usability and training efficiency.
- Evaluation using PSNR and SSIM confirmed the effectiveness of the Pix2Pix model, with room for further optimization in CycleGAN.

" Overall, this work demonstrates that GAN-based architectures are powerful tools for real-world computer vision problems such as low-light image enhancement and domain adaptation. With further refinements like identity loss and data augmentation, both models have strong potential for production-level performance. "

1. Generator Loss in Pix2Pix

The generator's loss is:

$$L_G = L_{\text{GAN}} + \lambda L_{L1}$$

Where:

- L_{GAN} → Binary cross-entropy (how well it fools the discriminator).
- L_{L1} → Pixel-wise difference between generated and real images.
- λ → Large weight (typically 100) to prioritize image reconstruction.

Future Work

While the current Pix2Pix and CycleGAN implementations yielded promising results, several enhancements can further improve performance and robustness:

1. Improved Image Quality
    - Incorporate perceptual loss or VGG-based content loss to better preserve details and texture.
    - Use attention mechanisms (like self-attention GANs) to focus on critical areas such as road boundaries and vehicle lights.
2. Data Augmentation
    - Apply advanced augmentation techniques (e.g., random shadows, Gaussian blur, or color jitter) to make the model more generalizable to real-world night scenes.
3. Higher Resolution Outputs
    - Extend the model to handle higher resolution images using techniques like progressive growing or multi-scale GANs.
4. Real-Time Inference
    - Optimize the trained models using TensorRT, ONNX, or TF Lite for deployment in real-time applications such as autonomous driving systems.
5. Quantitative Evaluation
    - Expand evaluation metrics beyond SSIM and PSNR, using LPIPS, FID, or user studies for more perceptual relevance.
6. CycleGAN Enhancements
    - Add identity loss, semantic consistency, and improved discriminators to make the unpaired translation more stable and realistic.
7. Multi-Domain Training
    - Combine multiple weather or lighting domains (e.g., fog, rain, dusk) using StarGAN or Unified GANs for better adaptability.

References

- *Pix2Pix (Isola et al., 2017)*
  Image-to-Image Translation with Conditional Adversarial Networks
  CVPR 2017
  https://arxiv.org/abs/1611.07004

- *tutorial of pix2pix and cycle gan from the youtube*

  https://www.youtube.com/watch?v=UcHe0xiuvpg