DAY AND NIGHT IMAGE TRANSLATION USING CYCLE-GAN

Varad Thikekar - CSC 449 - Department of Computer Science

Tanvi Shroff - CSC 449 - Department of Computer Science

ABSTRACT

The field of image transformation has made significant strides with the introduction of deep learning algorithms, especially in areas like style transfer and picture-to-image translation. Using Cycle Generative Adversarial Networks (CycleGAN) to address the fascinating problem of converting daytime photos into nighttime scenes and vice versa with various datasets is the focus of this paper. The project's primary objective is to generate realistic and convincing images by transforming daytime scenes into nighttime aesthetics. Motivated by practical applications in surveillance, self-driving cars, and photography, the proposed system aims to seamlessly convert images between day and night, enhancing visual quality and functional utility.

1 INTRODUCTION

The ability to turn photos taken in the daytime into realistic night settings and vice versa is extremely promising for a variety of applications, such as autonomous driving, visual content creation, and monitoring. This work explores the day-to-night picture transformation using a CycleGAN-based model and its implementation and evaluation. CycleGAN uses generative adversarial networks to create domain-specific transformations that produce believable, contextually appropriate night sceneries while maintaining the structural integrity of the source photos.
The innate difficulties presented by the dynamic variations in lighting, color, and climatic conditions between day and night serve as the driving force for this research. Conventional techniques for altering images frequently encounter difficulties in capturing the subtle aspects linked to these changes. On the other hand, CycleGAN can generate high-quality images that preserve the semantic content of the input photos while also reflecting the target domain's visual aesthetics thanks to its capacity to learn bidirectional mappings between the day and night domains.
In the subsequent sections, we elaborate on the problem statement, outlining our objectives and the broader motivation for this research. We then detail our approach, including the dataset used, preprocessing techniques, and the architecture of the CycleGAN model.

2 RELATED WORK

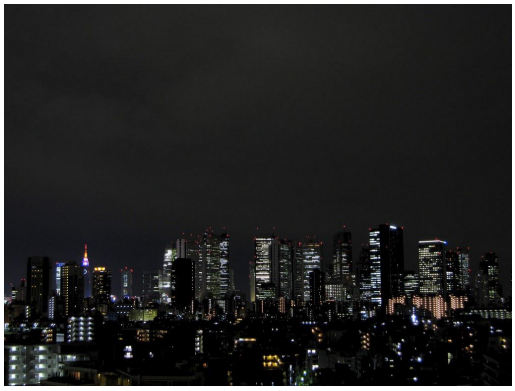
Introduced in 2017, CycleGAN is a relatively new concept that has received a lot of interest in the field of deep learning. Its practical applications have been the subject of numerous studies, with a focus on real-world situations. CycleGAN has found useful uses in a variety of real-life scenarios, even if the majority of its early implementations were turning pictures of horses into zebras, changing the artistic style of photographs, or changing the seasonal characteristics of graphics. A couple of instances of its application are driving simulation scenarios and producing handwritten Chinese characters using CycleGAN methodologies.
With an aim towards improving the realism of nighttime visuals in the field of autonomous driving, the work presented in [1] presents a novel approach to day-to-night image translation. In contrast to previous approaches, the suggested strategy uses generative adversarial networks (GANs) in conjunction with semantic segmentation to provide more accurate and visually appealing results. The proposed method in [2] uses both paired and unpaired images for training. It utilizes unpaired CycleGAN training to generate fake nighttime images, which are then paired with daytime images to

create a day-night paired dataset. The paired CycleGAN is then used to enhance the local contrast and restore fine details in the daytime images. So, both paired and unpaired learning methods are employed in the proposed method.

3 DATASETS

We have used 3 different datasets in our experiments. Firstly, we use the city view day and night dataset. It consists of 522 images taken at daytime and 227 images taken at nighttime. They are of size 1024x1024 and are of high definition. Secondly, we use a road image dataset. This contains 14,607 daylight images and 16,960 nighttime images taken by the front camera of the car. The last set of images are images of people taken during the day and at night. As this dataset has been made by us, it is small. It contains 134 day images and 62 night images. Sample images from the datasets are as shown below.

(a) City view Daylight Image



(b) City view Nighttime Image

(c) Road Day Image

(d) Road Night Image

(e) Portrait Day Image

(f) Portrait Night Image

Figure 1: Datasets

4 METHOD

4.1 DATA PREPROCESSING

We are conducting the following preprocessing on the three datasets:

- Random cropping: Because it introduces variability in the training data by showing the model different perspectives of the same image. This helps the model generalize better to different scales, positions, and orientations of objects within the image.
- Normalization: Because normalizing images helps stabilize and expedite the training process. By scaling pixel values to [-1, 1], the optimization algorithms are less likely to encounter numerical instability issues during backpropagation.
- Random flipping: Because it simulates real-world scenarios where objects may be viewed from different perspectives. This helps the model learn to recognize objects in a more realistic setting.
- Random rotating: Because it encourages the model to learn features invariant to rotations. This can result in more informative and generalized feature representations.
- Image Resizing: Because it ensures that both domains (source and target) have a consistent input size. This consistency is crucial for the generators and discriminators in the CycleGAN, as they expect images of the same dimensions.

(a) Orginal image (b) Preprocessed image

Figure 2: Data Preprocessing

4.2 MODEL ARCHITECTURE

The CycleGAN consists of two generators (day2night_gen and night2day_gen) and two discriminators (day2night_disc and night2day_disc). The generators aim to transform images from one domain to another, and the discriminators try to distinguish between real and generated images.
The following points describe the key components and architecture of the model:

1. Generator:
   - Reflection Padding: By using reflection padding, the model can capture spatial information near the image borders without introducing artifacts that may occur with zero-padding. It helps in maintaining the structure and features at the edges of the image during convolutional operations.
   - Downsampling Block: This is used in the generator to progressively reduce the spatial dimensions of the input image, extracting essential features and capturing the overall structure.
   - Residual Block: This consists of two convolutional layers with instance normalization and skip connections.

  - Upsampling Block: This is similar to downsampling blocks. These blocks use transposed convolutional layers for upsampling. The number of filters is halved in each block.
  - Activation Function: ReLU activation is used throughout the generator, except for the last layer, which uses the tanh activation function.
2. Discriminator:
  - Downsampling Block: This downsamples the image size using convolutional layers with increasing filter sizes.
  - Convolutional Block: This block uses convolutional layers with Leaky ReLU activation for feature extraction and downsamples the input image.
  - Activation Function: Leaky ReLU activation is used throughout the discriminator.
3. Loss Functions:
  - Discriminator Loss: Binary Crossentropy Loss is used for both the real and generated images.
  - Generator Loss: Binary Crossentropy Loss is also used for the generator.
  - Cycle Consistency Loss: Mean Absolute Error Loss is used for cycle consistency between the original and reconstructed images.
  - Identity Loss: Mean Absolute Error Loss is used for identity mapping, where the generator should map an image to itself.

4.3 Evaluation Metrics

The following quality metrics are used by us:

- Visual Inspection: Visual inspection is important in assessing the performance of CycleGAN or any generative model. While quantitative metrics provide objective measures, visual inspection allows us to subjectively evaluate the generated images' quality.
- Structural Similarity Index (SSIM): SSIM assesses the structural similarity between the generated images and the ground truth images. It takes into account luminance, contrast, and structure, providing a score between -1 and 1, where 1 indicates perfect similarity.
- Mean Squared Error (MSE): It measures the average squared difference between the pixel values of the generated images and the corresponding ground truth images.

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (I_{\text{gt}}(i) - I_{\text{gen}}(i))^2$$

- Peak Signal-to-Noise Ratio (PSNR): PSNR is a measure of the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation.

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

5 Experiments

To assess the efficacy of our CycleGAN model in performing night-to-day and day-to-night image transformations, we conducted a series of comprehensive experiments. The two primary objectives of our project are:

- **Autonomous Driving Enhancement**: We want to develop a CycleGAN model capable of effective day-night conversion for road images captured by car cameras. The goal is to

improve the performance of autonomous driving systems by allowing them to perceive and navigate road scenes under varying lighting conditions. Through this, we aim to enhance the safety and reliability of autonomous vehicles, particularly in scenarios where visibility is challenging, such as during nighttime or adverse weather conditions.

- **Tourism-Driven Photorealism**: Another key objective is to extend the capabilities of the CycleGAN model to perform day-night conversion on images of human subjects in different tourist destinations. By achieving realistic transformations, the project seeks to address the contemporary trend of tourists selecting the time of their visit based on the desired photographic aesthetics. This application aims to provide visually appealing day and night images of landmarks and tourist spots, catering to the preferences of tourists and influencing their decisions on when to visit iconic destinations. The project thereby aims to contribute to the field of tourism photography and social media content creation.

Initially, we start with a model with the following generator and discriminator architecture:

The ”get generator” function defines a generator model for CycleGAN with a specified number of residual connections. It takes an input image with shape (width, height, 3) and consists of an encoder, residual connections, and a decoder. Encoder: The input image undergoes a series of downsampling operations (enc1, enc2, enc3, enc4) to extract hierarchical features. Residual Connections: Residual blocks are applied to the output of the encoder. The residual blocks help the model capture and propagate important features through the network. Decoder: The decoder part of the network involves upsampling operations (dec1, dec2, dec3) with skip connections to concatenate features from the corresponding encoder layers. This process allows the model to recover spatial details lost during the downsampling. Output Layer: The final output is generated by applying a convolutional layer with tanh activation to ensure the pixel values are in the range [-1, 1].

The ”PATCH discriminator” function defines a discriminator model with an encoder structure for patch-wise discrimination. The input to the discriminator is an image with shape (width, height, 3). Encoder: The input image undergoes a series of downsampling operations (dwn1 to dwn5). Each downsampling step reduces the spatial dimensions of the input while increasing the number of filters. Output Layer: The final output is obtained using a convolutional layer with a single filter and a kernel size of 4. This produces a single-channel prediction map with the spatial dimensions (29, 29, 1). The output represents the discriminator's decision about the authenticity of the input image.

With the above model architecture, we get the following outputs:

(a) Day to night transformation (b) Night to day transformation

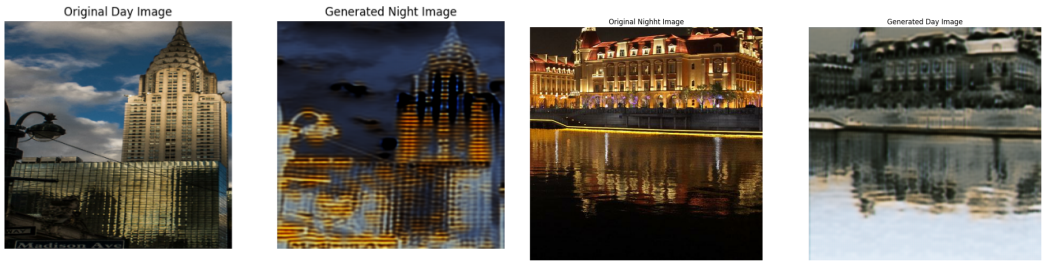
Figure 3: Initial model output

We can visually inspect and infer that the model has not done a good job. Now to fine tune the model, we increase the number of residual blocks in the generator to learn more complex and abstract features from the input data. As the number of residual blocks increases, the network gains the capacity to capture hierarchical representations, potentially improving its ability to understand intricate patterns. We also halve the upsample and downsample blocks for faster training and inference times due to less number of parameters.

(a) Day to night transformation (b) Night to day transformation

Figure 4: Improved model output

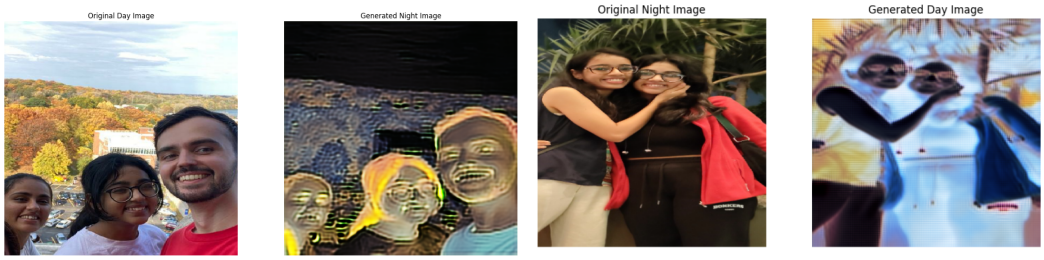
Now, since we are satisfied with the model outputs we test if the model output could be improved by using better resolution images for training. Since, we could not find better road image dataset, we train the model on city view images. The output are as follows:



(a) Day to night transformation (b) Night to day transformation

Figure 5: City view image transformation

Now since the model is giving decent outputs for the transformation, we train the same model on the human images. The outputs are as follows:



(a) Day to night transformation (b) Night to day transformation

Figure 6: City view image transformation

6 RESULTS

Generated samples	SSIM	MSE	PSNR
Sample 1	-0.005	0.472	3.259
Sample 2	0.428	1.351	-1.309
Sample 3	0.030	0.473	3.245
Sample 4	-0.140	1.019	-0.0843
Sample 5	0.244	1.109	-0.451
Sample 6	-0.244	0.71	1.48
Sample 7	0.138	0.99	0.0005
Sample 8	-0.096	0.946	0.241
Sample 9	0.117	0.729	1.372
Sample 10	-0.123	1.234	0.678

Table 1: Evaluation for human portraits view generated iamges (SSIM, MSE, and PSNR)

Generated samples	SSIM	MSE
Sample 1	-0.024	0.87
Sample 2	-0.346	1.25
Sample 3	-0.23	0.93
Sample 4	-0.39	0.34
Sample 5	0.049	1.094
Sample 6	-0.073	0.59
Sample 7	-0.03	0.558
Sample 8	-0.04	0.86
Sample 9	-0.16	0.17
Sample 10	-0.38	1.321

Table 2: Evaluation for city view generated iamges (SSIM, MSE, and PSNR)

For Human Portraits View Generated Images:

Average SSIM: (0.005+0.428+0.0300.140+0.2440.244+0.1380.096+0.1170.123)/10=0.027
Average MSE: (0.472+1.351+0.473+1.019+1.109+0.71+0.99+0.946+0.729+1.234)/10=0.954

For City View Generated Images:

Average SSIM: (0.0240.3460.230.39+0.0490.0730.030.040.160.38)/10=0.175
Average MSE: 0.87+1.25+0.93+0.34+1.094+0.59+0.558+0.86+0.17+1.321)/10=0.776

7 CONCLUSION

The better performance of the CycleGAN model on city view images compared to human portraits can be attributed to the simpler and more consistent structures present in cityscapes. City view images often exhibit well-defined semantic boundaries, aiding the model in learning and generalizing day-to-night transformations effectively. In contrast, the complexity and variability inherent in human portraits, including diverse expressions and lighting conditions, present a greater challenge for accurate transformation. The dataset characteristics, particularly the limited size and variability of the human portraits dataset, and the chosen model architecture may contribute to the observed differences.
The domain gap between day and night images for human subjects may be more challenging to bridge compared to road scenes. Addressing domain gaps often requires additional techniques like domain adaptation or more sophisticated architectures.
This project focused on implementing a CycleGAN model for the transformation of images between day and night settings. The primary objective was to enhance the adaptability of the model across diverse scenarios, specifically targeting road scenes and human portraits. Through extensive

experimentation, we fine-tuned the model architecture, adjusting parameters such as the number of residual blocks and upsampling/downsampling layers. Our approach involved training the model on distinct datasets, evaluating its performance using metrics like SSIM, MSE, and PSNR. The outcomes provided valuable insights into the model's strengths and areas for improvement, paving the way for future advancements in image translation and content adaptation.

REFERENCES

[1] Lee, Jinho, Daiki Shiotsuka, Geonkyu Bang, Yuki Endo, Toshiaki Nishimori, Kenta Nakao, and Shunsuke Kamijo. "Day-to-night image translation via transfer learning to keep semantic information for driving simulator." IATSS Research (2023).

[2] Son, Dong-Min, Hyuk-Ju Kwon, and Sung-Hak Lee. "Enhanced Night-to-Day Image Conversion Using CycleGAN-Based Base-Detail Paired Training." Mathematics 11, no. 14 (2023): 3102.

[3] https://www.kaggle.com/datasets/heonh0/daynight-cityview

[4] https://www.kaggle.com/datasets/raman77768/day-time-and-night-time-road-images

[5] https://www.tensorflow.org/tutorials/generative/cyclegan

[6] Adhikari, Binod, K. C. Hari, and Sharan Thapa. "NIGHT TO DAY AND DAY TO NIGHT IMAGE TRANSFER USING GENERATIVE ADVERSARIAL NETWORK."

[7] X. Shao, C. Wei, Y. Shen and Z. Wang, "Feature Enhancement Based on CycleGAN for Nighttime Vehicle Detection," in IEEE Access, vol. 9, pp. 849-859, 2021, doi: 10.1109/ACCESS.2020.3046498.

[8] Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros. "Unpaired image-to-image translation using cycle-consistent adversarial networks." In Proceedings of the IEEE international conference on computer vision, pp. 2223-2232. 2017.

[9] https://www.kaggle.com/code/virajkadam/day-night-image-translations-using-cyclegan

[10] Lin, Che-Tsung, Sheng-Wei Huang, Yen-Yi Wu, and Shang-Hong Lai. "GAN-based day-to-night image style transfer for nighttime vehicle detection." IEEE Transactions on Intelligent Transportation Systems 22, no. 2 (2020): 951-963.