

# INFO8010: Interior Design Style Transfer

Marie Goffin<sup>1</sup> and Thomas Fredrich<sup>2</sup>

<sup>1</sup>*mgoffin@student.uliege.be (s191956)*

<sup>2</sup>*thomas.fredrich@student.uliege.be (s191717)*

## I. INTRODUCTION

Interior design visualization remains challenging for non-professionals. Most people struggle to imagine how their spaces would look in different design styles. Traditional visualization methods require either professional design skills or extensive photo editing expertise.

In this paper, we explore a machine learning approach for transforming room interior images between design styles. Our method investigates an enhanced CycleGAN architecture that aims to preserve structural elements while changing aesthetic features.

Our work examines:

- Adapting CycleGAN for room style transfer
- Implementing multi-scale discriminators and improved residual blocks
- Testing a structural preservation loss to maintain architectural elements
- Evaluating results using LPIPS and FID metrics

We focus specifically on transferring between industrial and scandinavian bathroom styles. Our approach works with unpaired datasets, addressing the practical limitation that paired before/after images of the same room in different styles are rarely available.

## II. RELATED WORK

### A. Image-to-Image Translation

Image-to-image translation has seen significant progress in recent years. Pix2Pix [1] introduced a general framework for supervised image translation using conditional GANs, requiring paired training data. CycleGAN [2] extended this approach to the unpaired setting by introducing cycle consistency loss, enabling training without directly corresponding before/after images. This unpaired approach is particularly valuable for interior design applications, where obtaining before/after photos of the same room in different styles is difficult.

### B. Style Transfer

Neural style transfer, explored by Gatys et al. [3], showed the ability to transfer artistic styles to photographs using feature representations from pre-trained networks. Subsequent approaches improved efficiency [4] and versatility [5]. In particular, adaptive instance normalization (AdaIN) [5] and whitening and coloring transforms (WCT) [6] demonstrated better separation of content and style. While these methods work well for artistic style transfer, they often struggle with the structural preservation needs of interior design applications.

### C. Diffusion Models

More recently, diffusion-based generative models have shown exceptional capabilities in image generation and manipulation tasks. Denoising diffusion probabilistic models (DDPM) [7] introduced a new paradigm for high-quality image synthesis. These techniques have evolved into powerful text-to-image models such as DALL-E 2 [8] and latent diffusion models [9].

While these models can produce impressive style transformations with text guidance, they typically require large-scale training data and computational resources. Some recent work has explored using diffusion models specifically for image-to-image translation tasks [10], though often with different training requirements compared to GAN-based approaches.

### D. Interior Design Applications

Image translation and style transfer techniques have been applied to interior design in both academic research and commercial applications. Several existing tools offer interior visualization capabilities. Our work builds upon Fu [11], who demonstrated promising results by enhancing CycleGAN for art style transfer.

We adapt Fu’s architectural modifications—including improved residual blocks, multi-scale discriminators, and specialized loss functions—specifically for room style transfer. While newer diffusion-based approaches show promise, we chose to work with CycleGAN for its established ability to work with unpaired image datasets of modest size and its more accessible computational requirements.

### III. METHODOLOGY

#### A. Dataset

Our project uses a subset of bathroom interior images sourced from the "Interior Design Images and Metadata Dataset from Pinterest" (available on Kaggle as "galinakg/interior-design-images-and-metadata"). The original dataset contained multiple room types (bathroom, bedroom, kitchen, living room) with various design styles, accompanied by CSV metadata files linking images to style categories and attributes.

For our implementation, we made several modifications to the dataset:

- We focused exclusively on bathroom images to enable our model to specialize in a single room type
- We cleaned the dataset to remove:
  - Images containing text/writing
  - Images showing only color palettes
  - Images with single objects rather than complete rooms
- We removed the CSV metadata files and instead organized images directly into style-specific directories

While the original dataset contained five bathroom design styles (boho, industrial, minimalist, modern, and scandinavian), we chose to work with **industrial** and **scandinavian** styles. This decision was based on the clear visual contrast between these styles—industrial bathrooms typically feature raw, utilitarian elements with exposed materials and metal fixtures, while scandinavian bathrooms are characterized by light, airy spaces with natural materials and neutral colors. We reasoned that this strong stylistic difference would make the style transfer effects more pronounced and easier to evaluate.

For training our model, we split each style category into train (70%), validation (15%), and test (15%) sets. This resulted in approximately 105 industrial and 79 scandinavian bathroom images for training, with proportional numbers for validation and testing.

#### B. Model Architecture

Our architecture is based on CycleGAN [2] with several enhancements inspired by Fu [11]. The overall system consists of two generator networks and two discriminator networks, enabling bidirectional translation between industrial and scandinavian bathroom styles. Figure 1 illustrates the high-level architecture of our system.

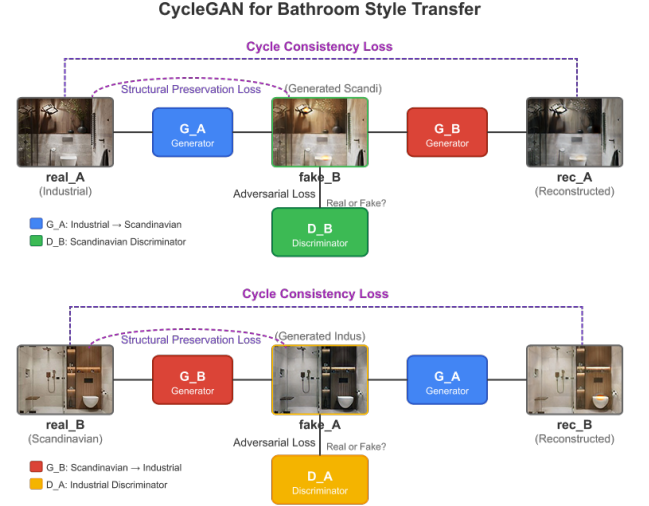


FIG. 1: CycleGAN architecture for bathroom style transfer between Industrial and Scandinavian styles, showing generators ( $G_A$ ,  $G_B$ ), discriminators ( $D_A$ ,  $D_B$ ), and our integrated loss functions.

##### 1. Generator Architecture

Our generator network uses a ResNet-based architecture tailored for interior design style transfer:

- Initial reflection padding and  $7 \times 7$  convolution to preserve edge information, which is critical for maintaining bathroom structural elements
- Two downsampling layers with stride-2 convolutions and instance normalization
- Nine residual blocks for feature transformation while maintaining spatial information
- Bilinear interpolation upsampling followed by  $3 \times 3$  convolutions to reduce checkerboard artifacts (edge distortions caused by convolutional operations at image boundaries) common in transpose convolutions
- Final reflection padding and  $7 \times 7$  convolution with Tanh activation

The dual generator structure allows us to create both Industrial $\rightarrow$ Scandinavian ( $G_A$ ) and Scandinavian $\rightarrow$ Industrial ( $G_B$ ) transformations. This bidirectionality enables cycle consistency training without requiring paired images.

##### 2. Multi-scale Discriminator

To better capture both global structure and local details, we implemented a multi-scale discriminator that operates at two resolutions:

- Full-resolution discriminator for capturing overall image coherence and stylistic patterns
- Half-resolution discriminator for focusing on larger-scale features (like layout and fixtures)
- Each discriminator uses a PatchGAN architecture with spectral [12] normalization for training stability

This multi-scale approach helps the model better distinguish between industrial and scandinavian styles while maintaining structural coherence. For example, the full-resolution discriminator might focus on detailed textures like tile patterns or fixture finishes, while the half-resolution discriminator captures more general properties like color schemes and spatial arrangements.

### 3. Perceptual Network

For our structural preservation loss, we incorporated a perceptual network based on pre-trained VGG19 features. This network extracts features at three different depths, allowing us to capture different levels of structural information:

- Low-level features (edges, textures) from early layers, which help preserve basic elements like walls and fixtures
- Mid-level features (patterns, local structures) from middle layers, capturing bathroom fixtures and their arrangement
- High-level features (spatial arrangements) from deeper layers, representing overall bathroom layout

By weighting these feature levels differently in our loss function (with higher weight on low and mid-level features), we encourage the model to preserve structural elements while still allowing stylistic changes appropriate for the target domain. This is especially important for bathroom style transfer, where we want to maintain the layout and fixtures while changing decorative elements to match either industrial or scandinavian design aesthetics.

### C. Loss Functions

Our training objective combines several loss terms:

$$\mathcal{L} = \mathcal{L}_{GAN} + \lambda_{cycle}\mathcal{L}_{cycle} + \lambda_{id}\mathcal{L}_{id} + \lambda_{struct}\mathcal{L}_{struct} \quad (1)$$

#### 1. Adversarial Loss

We use the GAN loss with mean squared error (MSE) for training stability:

$$\mathcal{L}_{GAN}(G, D) = \mathbb{E}_y[(D(y) - 1)^2] + \mathbb{E}_x[D(G(x))^2] \quad (2)$$

#### 2. Cycle Consistency Loss

This loss ensures that transformed images can be mapped back to their original versions:

$$\mathcal{L}_{cycle} = \mathbb{E}_x[\|G_B(G_A(x)) - x\|_1] + \mathbb{E}_y[\|G_A(G_B(y)) - y\|_1] \quad (3)$$

#### 3. Identity Loss

The identity loss encourages color and texture preservation when appropriate:

$$\mathcal{L}_{id} = \mathbb{E}_x[\|G_B(x) - x\|_1] + \mathbb{E}_y[\|G_A(y) - y\|_1] \quad (4)$$

#### 4. Structural Preservation Loss

This loss helps maintain architectural elements while allowing style changes:

$$\begin{aligned} \mathcal{L}_{struct} = & \sum_{i=1}^3 w_i \cdot \|F_i(G_A(x)) - F_i(x)\|_1 \\ & + \sum_{i=1}^3 w_i \cdot \|F_i(G_B(y)) - F_i(y)\|_1 \end{aligned} \quad (5)$$

where  $F_i$  represents the feature maps from the  $i$ -th layer of the perceptual network, and  $w_i$  are weights decreasing with depth to allow style changes while preserving structure.

## IV. IMPLEMENTATION

### A. System Overview

We implemented our system in Python using PyTorch. The system handles the complete pipeline from dataset preparation to inference, with special attention to preserving architectural elements during style transformation.

### B. Project Structure

Our codebase is organized into several modular components:

- **config.py**: Configuration file containing all hyperparameters, including batch size, learning rates, model dimensions, and loss weights. This configuration file contains the standard parameters which can then be modified by specifying the desired values in the command line.

- **data\_preparation.py:** Handles dataset operations including loading bathroom images, splitting into train/validation/test sets (70%/15%/15%), and creating appropriate data loaders with transformations for resizing, cropping, and normalization.
- **train.py:** Implements the training loop with both generator and discriminator optimization steps, periodical checkpoint saving, and sample visualization. Integrated with Weights & Biases to monitor the progress.
- **evaluate.py:** Computes quantitative metrics including LPIPS for measuring perceptual similarity and FID for assessing generation quality. Creates visualizations for qualitative assessment.
- **inference.py:** Applies the trained model to new bathroom images.
- **models/:** Contains the neural network architecture implementation:
  - **cycle\_gan.py:** Implements the core CycleGAN model class, coordinating the interaction between generators, discriminators, and multiple loss functions.
  - **networks.py:** Defines the ResNet generator, multi-scale discriminator, and perceptual network architectures.
  - **losses.py:** Implements the loss functions including GAN, cycle consistency, identity, and structural preservation losses.
- **utils.py:** Provides utility functions for image processing, tensor-to-image conversion, and visualization.

### C. Training Process

We trained our model using the following hyperparameters:

- **Batch size:** 2 (specified via command line for better training results). Smaller batch sizes often lead to better generalization in GANs by providing noisier gradients that help escape sharp local minima.
- **Learning rate:** 0.0002 with Adam optimizer ( $\beta_1 = 0.5$ ,  $\beta_2 = 0.999$ ). This learning rate balances stability and convergence speed, while the reduced  $\beta_1$  value (from standard 0.9 to 0.5) helps address training instability common in adversarial networks.
- **Training epochs:** 100, providing enough iterations for the model to learn style transfer.

- **Image size:**  $256 \times 256$ , offering a good balance between preserving architectural details important for room scenes while keeping reasonable computational requirements.
- **Loss weights:**  $\lambda_{cycle} = 10.0$ ,  $\lambda_{id} = 0.5$ ,  $\lambda_{struct} = 10.0$ . The high weight on cycle consistency enforces content preservation, while the lower identity weight allows for style changes. Following Fu’s [11] approach, we assigned a high structural preservation weight to maintain room layouts during style transformation.

The training process included several strategies for stability:

- Learning rate decay using a step scheduler (reducing by half every 30 epochs)
- Image buffer of size 50 for discriminator training to reduce oscillation by using a history of generated images
- Spectral normalization in the discriminator to enforce Lipschitz continuity and improve training dynamics
- Reflection padding to reduce border artifacts

Weights & Biases allowed us to observe loss curves across different components (adversarial, cycle, identity, and structural), learning rate schedules, and generated samples throughout training.

### D. Training Monitoring

We integrated Weights & Biases (wandb) into our training pipeline to track model performance throughout the 100 epochs. By monitoring various loss components individually, we gained valuable insights into the model’s learning dynamics and convergence patterns.

#### 1. Overall Loss Progression

Figure 7 shows the generator loss (g\_loss) over training. The significant decrease from approximately 24 to 12 indicates improvement in the generator’s ability to produce realistic images that can fool the discriminator while satisfying the various consistency constraints.

The discriminator loss (d\_loss) shown in Figure 8 demonstrates rapid initial improvement followed by stabilization around 0.3, indicating the discriminator quickly learned to distinguish between real and generated images and maintained that capability throughout training.

The validation losses, shown in Figures 9 and 10, reveal potential issues in the adversarial training balance. The validation generator loss (val\_g\_loss) shows persistent fluctuations between 0.3 and 0.7 without clear improvement, while the validation discriminator loss (val\_d\_loss) shows a steady downward trend, stabilizing around 0.15-0.18 by the end of training. This imbalance suggests the discriminator became increasingly effective at identifying generated images, while the generator struggled to produce convincing samples for the validation set. This pattern is a common challenge in GAN training and may partly explain the relatively high FID scores we observed in our final evaluation.

## 2. Learning Rate Schedule

Figure 11 shows our step-based learning rate schedule, with initial rate of 0.0002 halving at epochs 30 and 60. This schedule helped stabilize training in later epochs and allowed for finer model adjustments as training progressed.

## 3. Domain-Specific GAN and Discriminator Losses

The domain-specific GAN losses (Figures 12 and 13) present interesting asymmetry. The GAN loss for domain A and B fluctuate between 0.4 and 0.7, while showing a slight upward trend in later epochs, suggesting increasingly challenging style generation.

The discriminator component losses (Figures 14 and 15) show similar convergence patterns, keeping a downward trend with both stabilizing in later epochs.

## 4. Cycle Consistency and Structure Preservation

The cycle consistency losses (Figures 16 and 17) demonstrate effective learning of the bidirectional mapping. Both domains show similar trends, starting around 3.0-3.5 and decreasing to approximately 1.3-1.4 by training completion.

The structural preservation losses (Figures 18 and 19) show rapid initial decrease, indicating the model quickly learned to preserve bathroom structural elements. Both converge to values around 4, confirming the effectiveness of our structural preservation approach.

## 5. Identity Mapping Losses

Finally, the identity mapping losses (Figures 20 and 21) verify the model’s ability to preserve content when no style change is needed. Both domains show successful

convergence from initial values around 0.16 to approximately 0.07.

## E. Evaluation Metrics

We evaluated our model using two complementary metrics:

- **LPIPS** (Learned Perceptual Image Patch Similarity): This perceptual metric measures similarity between images based on features extracted from pre-trained networks. We used LPIPS to evaluate two aspects:
  - Cycle consistency: How well the original image is reconstructed after a full translation cycle ( $A \rightarrow B \rightarrow A$ )
  - Structural preservation: How well architectural elements are maintained when comparing original and style-transferred images
- **FID** (Fréchet Inception Distance): This metric measures the distance between feature distributions of real and generated images using the Inception-v3 network. Lower FID scores indicate generated images more similar to real ones, providing an assessment of overall generation quality and style transfer effectiveness.

These quantitative metrics, combined with qualitative visual assessment, allowed us to evaluate the performance of our bathroom style transfer system.

## V. EXPERIMENTS AND RESULTS

We conducted an evaluation of our style transfer model, inspecting both quantitative metrics and qualitative visual results. Our evaluation focused on the bidirectional style transfer between industrial and scandinavian bathroom styles.

### A. Quantitative Evaluation

Table I presents the quantitative evaluation metrics of our style transfer model after training for 100 epochs.

The LPIPS scores measure perceptual similarity (lower is better), with scores around 0.25 indicating reasonable preservation of content while allowing for style changes. Our model achieves reasonable cycle consistency, with an average score of 0.2536. This suggests that images can be transformed to the target style and back while maintaining content features. The structural preservation scores are similar, with slightly better preservation when converting from Scandinavian

Metric	Value
LPIPS Cycle Consistency A	0.2529
LPIPS Cycle Consistency B	0.2544
LPIPS Average Cycle Consistency	0.2536
LPIPS Structural A to B	0.2620
LPIPS Structural B to A	0.2163
LPIPS Average Structural	0.2391
FID A to B	186.0744
FID B to A	199.4874
FID Average	192.7809

TABLE I: Evaluation metrics for our bathroom style transfer model.

to Industrial (0.2163) compared to Industrial to Scandinavian (0.2620).

FID scores measure the distribution distance between real and generated images. The FID scores (186.0744 for A to B and 199.4874 for B to A) indicate that while the generated images capture stylistic elements of the target domain, there remains a gap between the distribution of generated images and real images. This is expected given the complexity of bathroom designs and our relatively small training dataset.

## B. Qualitative Analysis

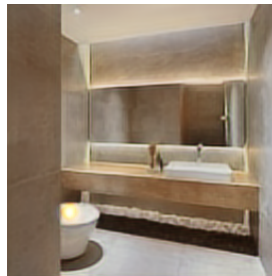
### 1. Industrial to Scandinavian Transfer

Figure 3 shows examples of transforming industrial bathroom styles to scandinavian style.

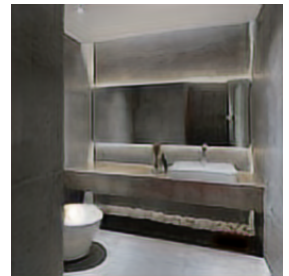


FIG. 2: Real Industrial bathroom (Real A)

Our model brightens industrial bathrooms when converting to Scandinavian style. We observe that the model seems to try to transform materials, with stone surfaces taking on wooden textures, and dark metal fixtures becoming lighter. The overall architectural structure—walls, fixtures, and general layout—is well preserved during the transformation, which is a key objective of our structural preservation loss.



(a) Generated Scandinavian (Fake B)



(b) Reconstructed Industrial (Rec A)

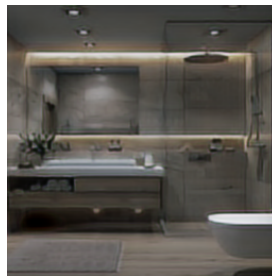
FIG. 3: Style transfer results from Industrial to Scandinavian and back

### 2. Scandinavian to Industrial Transfer

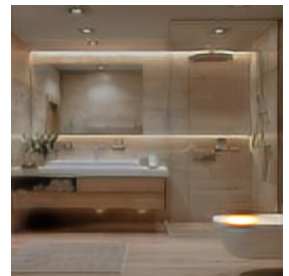
Figure 5 shows the transformation from scandinavian to industrial bathroom style.



FIG. 4: Real Scandinavian bathroom (Real B)



(a) Generated Industrial (Fake A)



(b) Reconstructed Scandinavian (Rec B)

FIG. 5: Style transfer results from Scandinavian to Industrial and back

When converting from Scandinavian to Industrial style, our model creates darker tones, sometimes approaching a black and white aesthetic. The model seems to be adding bronze-like materials or anthracite grey/black accents to sinks and mirror borders. This subtlety may be attributed to the model's focus on preserving structural elements while changing stylistic features.

### C. Cycle Consistency Analysis

Figure 6 illustrates the full cycle of transformations for both directions.

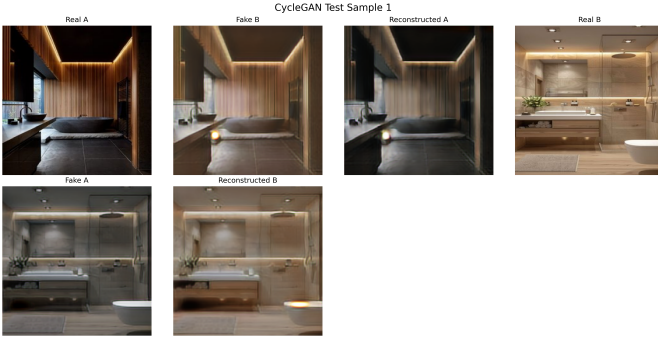


FIG. 6: Complete style transfer cycles. Industrial  $\rightarrow$  Scandinavian  $\rightarrow$  Industrial. Scandinavian  $\rightarrow$  Industrial  $\rightarrow$  Scandinavian.

The cycle consistency results show that after a full transformation cycle, images return to a state visually similar to their originals. For the Industrial  $\rightarrow$  Scandinavian  $\rightarrow$  Industrial cycle, the reconstructed image recovers the darker tones, while for the Scandinavian  $\rightarrow$  Industrial  $\rightarrow$  Scandinavian cycle, the model successfully retrieves the brighter aesthetic of Scandinavian design.

### D. Structural Preservation

A key feature of our approach is the structural preservation loss, which aims to maintain architectural elements while allowing stylistic changes. As visible in Figures 3 and 5, our model preserves the bathroom layout, fixture positions, and architectural boundaries while transforming the atmosphere, tones and sometimes materials.

The LPIPS structural scores (0.2620 for A to B and 0.2163 for B to A) support our visual observation that structural elements are largely preserved during style transfer. The better score for Scandinavian to Industrial transformation suggests that preserving structure while darkening and adding industrial textures may be easier than the reverse process. This asymmetry could be due to the nature of Scandinavian design, which often features simpler, cleaner lines that provide less structural information compared to the more complex textures and fixtures often found in industrial bathrooms.

Looking at the transformation examples, we can observe that elements positions and forms remain consistent through the style transfer process, while surface materials, color palettes, and lights are modified to match the target aesthetic. This shows the effectiveness of our

structural preservation loss in guiding the model to distinguish between content that should be preserved and stylistic elements that should be transformed.

## VI. DISCUSSION

While our CycleGAN approach produces promising results, our project has several limitations that require critical examination.

### A. Limitations of Style Transfer for Interior Design

Our experiments demonstrate that the current approach can effectively transform certain stylistic elements like colors, textures, and material appearances, but falls short in making more substantial changes that may be desired in interior design contexts. The model primarily alters the "atmosphere" of the bathrooms—brightening spaces when converting to Scandinavian style or darkening them for Industrial style—but struggles with more meaningful transformations.

The high FID scores (average of 192.78) indicate a significant distribution gap between generated and real images, suggesting that our transformations are not fully capturing the authentic characteristics of each design style. This limitation may stem from both the model architecture and the relatively small dataset size.

### B. Insufficient Feature Engineering

One shortcoming of our implementation is the minimal feature engineering. While we adopted Fu's enhanced CycleGAN architecture, we did not develop domain-specific features tailored to interior design. For example:

- No semantic segmentation to distinguish between different bathroom elements (walls, fixtures, floor, etc.)
- Lack of material-specific transformations that could better capture design style differences
- No specialized attention mechanisms to focus on style-defining elements
- Absence of design-specific perceptual features for more meaningful loss functions

These limitations restrict the model's ability to make targeted, design-relevant transformations, instead producing more general stylistic changes.



### C. Structural Preservation vs. Design Transformation

Perhaps the most significant limitation lies in our adherence to structural preservation. While Fu’s approach made sense for artwork where preserving the composition was essential, interior design transformation often requires changes to furniture and fixtures. Our strong structural preservation loss forces the model to maintain all architectural elements, preventing it from:

- Replacing industrial metal fixtures with wooden Scandinavian alternatives
- Transforming bathtub designs between styles
- Changing sink shapes and materials in a style-appropriate manner
- Modifying lighting fixtures which are key style indicators

This reveals a fundamental tension in our approach: successful interior style transfer often requires structural changes, yet our model explicitly penalizes such modifications. This constraint significantly limits the practical utility of the system for design visualization.

### D. CycleGAN Limitations for Design Applications

Our results suggest that CycleGAN, even with enhancements, may not be the optimal architecture for sophisticated interior design transformations. The inherent limitations include:

- Inability to generate new elements not present in the original image
- Limited capacity to understand the semantic meaning of different room components
- Tendency to produce artifacts in complex regions with many geometric details

These architectural limitations are evident in our results, where style transfer is largely limited to color shifts and texture changes rather than meaningful design transformations.

### E. Alternative Approaches

The limitations identified suggest that newer generative models, particularly diffusion models, might be better suited for interior design style transfer. Recent text-to-image and image-to-image diffusion models such as Stable Diffusion have shown superior capabilities in:

- Generating new objects and elements while maintaining context
- Producing high-fidelity transformations guided by text prompts
- Preserving important elements while allowing for targeted modifications

A diffusion-based approach would potentially allow for both preservation of room structure and transformation of specific design elements through controlled generation. This could enable more meaningful style transfers that actually replace industrial fixtures with Scandinavian ones, rather than merely adjusting their appearance.

Furthermore, we could also address the fundamental limitation of our current system, which applies the same transformation logic to all bathroom elements regardless of their design significance.

## VII. CONCLUSION AND FUTURE WORK

In this paper, we presented a deep learning approach for bathroom interior design style transfer using an enhanced CycleGAN architecture. Our system transforms bathroom images between industrial and scandinavian styles, changing the atmosphere and material appearances while preserving the room structure. The quantitative results, with LPIPS cycle consistency scores around 0.25, show that our model achieves a reasonable balance between style transformation and content preservation. Our visual results confirm this finding, showing clear stylistic changes from dark, metallic industrial aesthetics to bright, clean scandinavian designs and vice versa. The multi-scale discriminator and structural preservation loss were valuable in maintaining architectural integrity throughout these transformations.

Despite these achievements, our approach has limitations in making substantial furniture and fixture modifications necessary for complete style transformations. Future work should incorporate semantic understanding of room elements to enable more targeted transformations, potentially allowing specific fixtures to be re-designed while maintaining overall room structure. Additionally, exploring more advanced generative models, such as diffusion-based approaches, could overcome the limitations of CycleGAN and enable more sophisticated interior design transformations with improved fidelity. As a conclusion, while our current system shows the feasibility of automated room style transfer, the path toward truly comprehensive design visualization will require these more nuanced approaches that better balance preservation and transformation.

- Understanding semantic content of scenes



## Appendix A: Training Visualizations

This appendix contains the complete set of training visualization charts generated during our model training process. These visualizations were captured using Weights & Biases (wandb) monitoring tool over the course of 100 epochs.

### 1. Overall Loss Charts

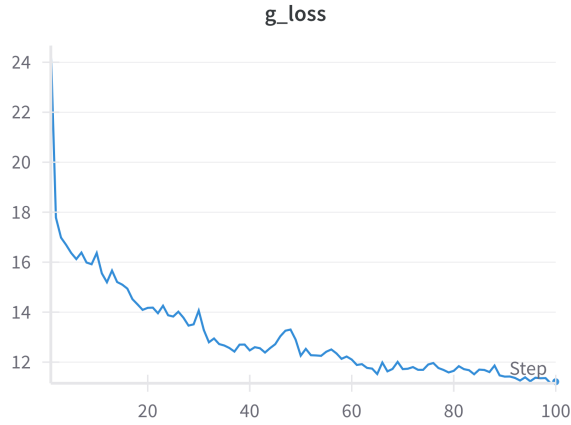


FIG. 7: Generator loss (g\_loss)

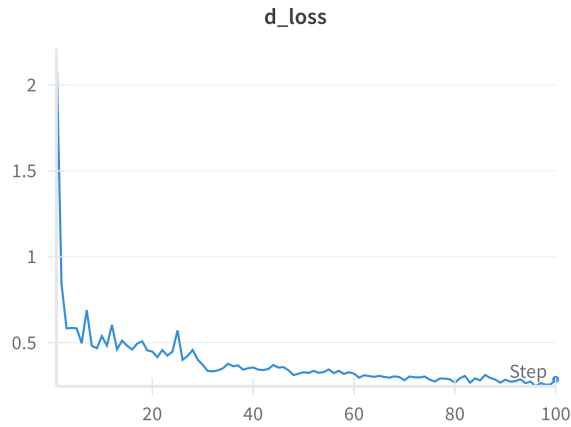


FIG. 8: Discriminator loss (d\_loss)

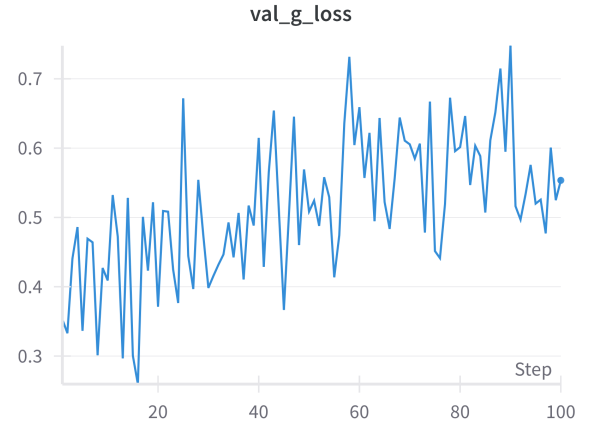


FIG. 9: Validation generator loss (val\_g\_loss)

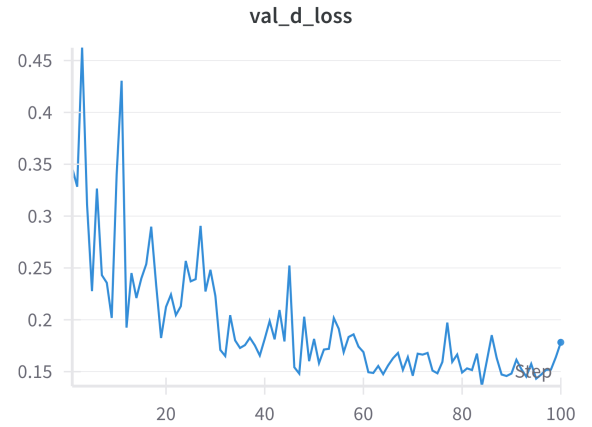


FIG. 10: Validation discriminator loss (val\_d\_loss)

### 2. Learning Rate Schedule

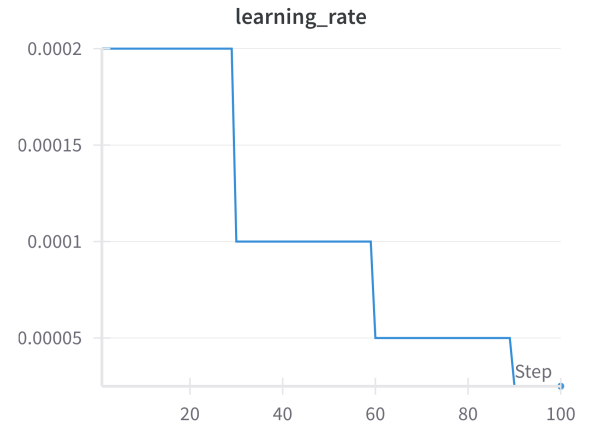


FIG. 11: Learning rate

### 3. Domain-Specific Adversarial Losses

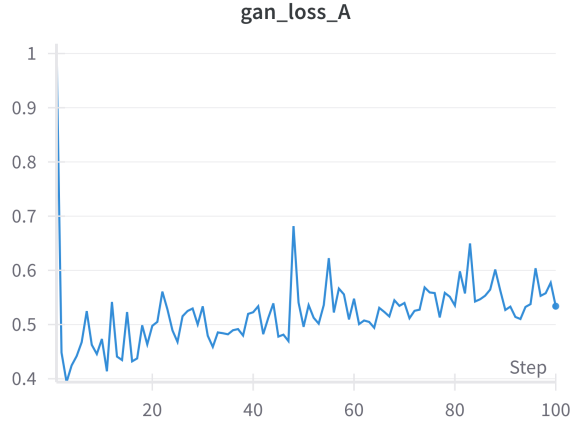


FIG. 12: GAN adversarial loss for domain A (gan\_loss\_A)

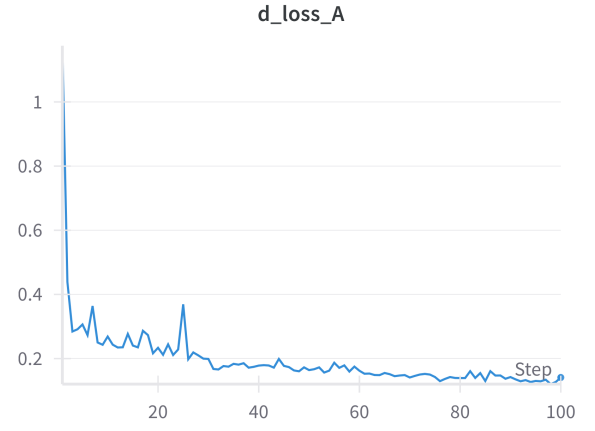


FIG. 14: Discriminator loss for domain A (d\_loss\_A)

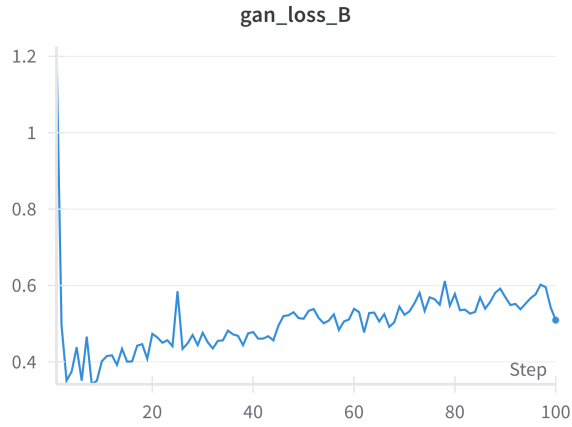


FIG. 13: GAN adversarial loss for domain B (gan\_loss\_B)

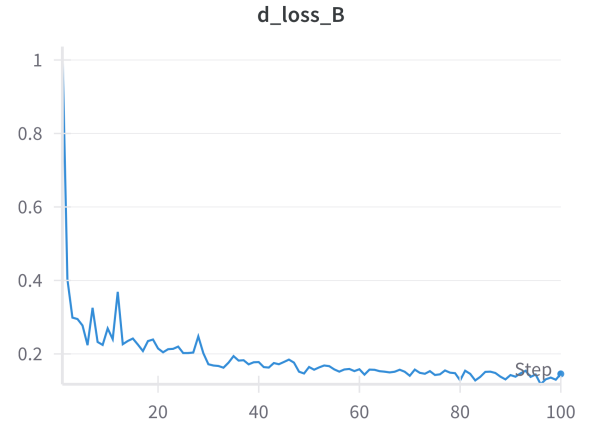


FIG. 15: Discriminator loss for domain B (d\_loss\_B)

#### 4. Cycle Consistency and Structural Preservation Losses



FIG. 16: Cycle consistency loss for domain A (cycle\_loss\_A)



FIG. 18: Structural preservation loss for domain A (struct\_loss\_A)

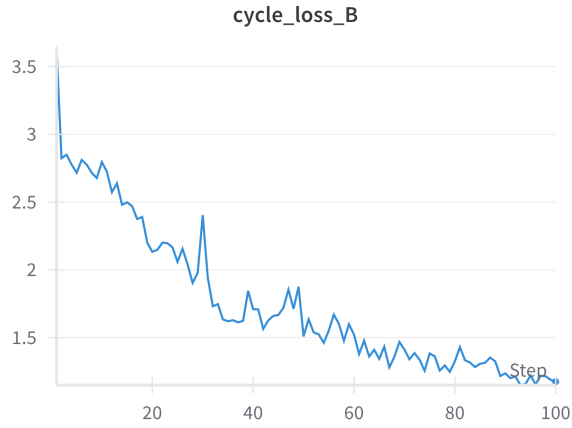


FIG. 17: Cycle consistency loss for domain B (cycle\_loss\_B)



FIG. 19: Structural preservation loss for domain B (struct\_loss\_B)

## 5. Identity Mapping Losses

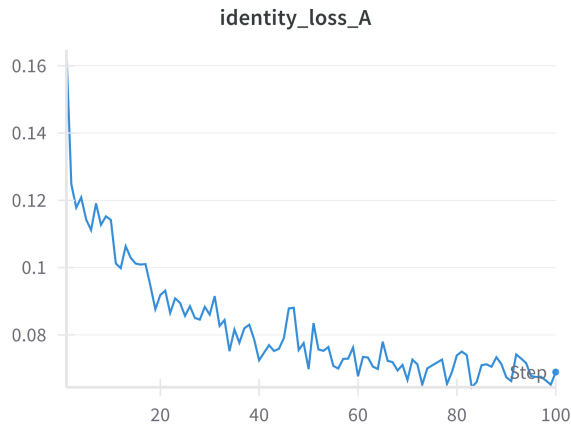


FIG. 20: Identity mapping loss for domain A (identity\_loss\_A)

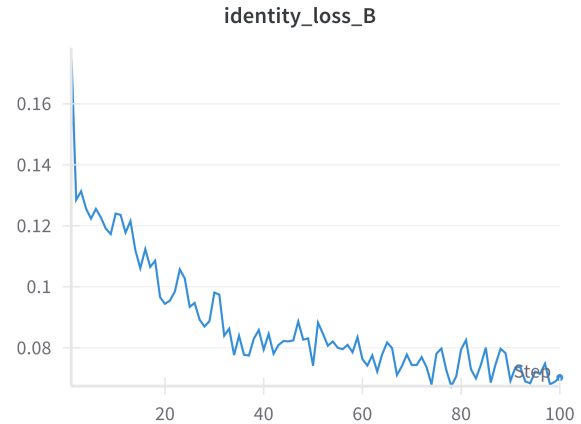


FIG. 21: Identity mapping loss for domain B (identity\_loss\_B)

- 
- [1] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.
  - [2] Jun-Yan Zhu, 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*, pages 2223–2232, 2017.
  - [3] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.
  - [4] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.
  - [5] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.
  - [6] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *Advances in neural information processing systems*, pages 386–396, 2017.
  - [7] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.
  - [8] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.
  - [9] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.
  - [10] Chitwan Saharia, William Chan, Huiwen Chang, Christopher A Lee, Jonathan Ho, Tim Salimans, David J Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022.
  - [11] Xuhui Fu. Digital image art style transfer algorithm based on cyclegan. *Scientific Programming*, 2022:6075398, 2022.
  - [12] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.