StarGAN Model to Translate Interior Style Trained with Virtual Images

Junhyeck Choi[1], Soowoong Hwang[2], Seungho Jeon[2] and Jae Seung Lee[2,3]

[1]Environmental Planning Institute, Seoul National University, conceal4682@snu.ac.kr
[2]Department of Landscape Architecture, Urban Design Concentration,
Graduate School of Environmental Studies, Seoul National University,
soowoong92@gmail.com, jshki3423@snu.ac.kr, jaeseung74@gmail.com
[3]Integrated Major in Smart City Global Convergence, Seoul National University

Corresponding author: Jae Seung Lee, jaeseung74@gmail.com

Abstract. Interest in personalized interior design remains robust, yet it involves time-consuming processes such as 3D modeling and rendering. The image-to-image translation model, an offshoot of deep learning technology, offers the potential to simplify design tasks by transforming the styles of indoor scene images. Even with the challenges of acquiring an interior image dataset before the model development, combining virtual images can help circumvent inconsistency and copyright restrictions. In our research, we synthesized virtual images of indoor scenes in various interior styles using a procedural generation algorithm in Unreal Engine, which enabled fast, real-time rendering. Using the virtual images, we trained an image-to-image translation model called StarGAN to enhance the generator network with skip connections. The model trained with a virtual dataset yielded more practical outcomes than training with only a limited number of real images. Our study underscores the utility of virtual images for training deep-learning models in interior design. The model can enhance existing design tools by providing quick and convenient image generation.

Keywords: Interior Design, Image-to-image Translation, Procedural Generation, StarGAN, Virtual Dataset
DOI: https://doi.org/10.14733/cadaps.2025.12-25

1 INTRODUCTION

The desire for personalized interior design persists as individuals strive to create aesthetically pleasing and comfortable living environments. Historically, Western culture has seen increased interest and market volume for interior design, viewed as an expression of a personal lifestyle aided by a professional designer [12]. As this trend expands, non-Western countries like China and Korea are experiencing rapid growth in the furniture market with diversified designs based on innovations in the e-commerce market [11] [21]. Furthermore, lifestyle changes spurred by the COVID-19 pandemic, such as encouraging contactless communication and increased time spent at home, have sparked a heightened interest in home furnishing [14].

The current interior design process, facilitated by 3D modeling and rendering software, is time-consuming due to complex operations like furniture arrangement, material application, and light adjustment. Leveraging deep learning technology can automate and simplify design tasks through personalized recommendations based on data-driven insights. Moreover, simplified design services can enhance customer accessibility, reduce the risk of trial and error, and lower product return rates [1].

In interior design, the indoor scene contains an overall visual composition of indoor spaces with various elements. Indoor scenes are one of the most popular design outputs for non-professionals in determining the overall context of space. The deep learning-based image-to-image translation model can quickly render indoor scene images by adding light effects [22], coloring greyscale images [15], and transforming interior styles [2]. These translation models are vital for helping customers select personal interior designs in their own space.

To develop the models, acquiring a large number of indoor scene images for training is challenging due to regional and era-specific design inconsistencies and copyright concerns. Virtual images can address these issues by rendering many indoor scenes in a shorter time and lower cost [19]. In many cases, models trained with virtual images have performed comparably to those using only real images [9] [19]. However, the full potential of using virtual images to develop an interior style transformation model still needs to be explored.

We aim to develop a high-performing image-to-image translation model trained with virtual images to transform interior styles. Initially, we synthesized virtual indoor scene images with a procedural generation algorithm to obtain a variety of styles. We then trained the StarGAN image-to-image translation model for multiple styles with virtual and real image datasets. Our model transformed input images into four distinct styles: classic, modern, natural, and romantic. This study contributes to the acceleration of the interior design process with changing indoor scene image styles. It also provides insights into the generation and utilization of virtual images for deep learning training.

2 RELATED WORK

2.1 Procedural Generation of Virtual Indoor Scene

Procedural generation is commonly used to produce content such as images and 3D models that adhere to predefined generation rules. These algorithms can produce large-scale virtual image datasets due to their rapid and repeatable process of creating various outputs that satisfy specific criteria.

In indoor scene generation, existing works have focused on furniture arrangement within 3D modeling to give realism to an arrangement with constraints. The position and orientation of furniture objects were adjusted by optimizing numerical conditions [10] [23] or through training with a floor plan database [8] [17] [20]. Kán and Kaufman [10] expanded this by randomizing material selection with furniture arrangement.

Some algorithms repeatedly generate 3D models and captures virtual images to synthesize indoor scene images. Feklisov et al. [4] created a virtual image dataset for semantic segmentation with randomly arranged furniture objects and materials. Game engines such as Unity [18] and Unreal Engine [5] can be utilized for capturing and rendering scene image datasets. These engines support real-time rendering, enabling rapid generation without extensive prerendering associated with 3D modeling software.

A list of furniture objects and materials can be assembled for each interior style to develop a procedural generation algorithm for stylized virtual indoor scene images. Implementation within a game engine facilitates fast rendering and generation of virtual indoor scene images suitable for training image-to-image translation models.

2.2 Generative Adversarial Networks for Interior Image Translation

Generative adversarial networks (GAN) have attracted attention in image generation due to their high performance and ability to produce realistic outputs. A GAN uses two distinct neural networks called generator and discriminator that learn in an adversarial manner. The generator produces a fake image while the discriminator evaluates its authenticity. As a result, the generator learns to create novel outputs that adhere to the distribution of the training set. GANs perform well in image-to-image translation problems where an image from one domain is transformed into another.

Extensions of the basic GAN model for image-to-image translation tasks were developed, such as CycleGAN or StarGAN. CycleGAN [24] is a single-domain translation model that does not require a paired original and target domain dataset. It introduced the concept of cycle consistency, which asserts that a translated image should retain its identity when restored to its original form. Generator networks incorporate a residual block [7] working in extremely deep networks to represent low-frequency information. StarGAN [3] is an advanced multi-domain translation model incorporating input images and target domain information into a single generator network.

In the interior design field, image-to-image translation models are applied to 3D model rendering [22], sketch colorization [15], and style transformation [2]. Yang et al. [22] incorporated a capsule block into the generator to account for relative positions due to the global light effect. Skip connections between the low-level and high-level layers facilitates capturing semantic relationships, thus improving the quality of interior image output [15]. Choi and Lee [2] found that the UNET structure with low-level and high-level channel concatenation can more dramatically transform interior patterns. Most studies conducted on interior design rely on single-domain models, predominantly CycleGAN, using skip-connected generator structures to enhance the quality of indoor scene outputs.

2.3 Contribution

Our research offers several significant contributions.

First, we expand the application of procedural generation in interior design with game engines by synthesizing content in alignment with various styles or themes.

Second, we train an image-to-image translation model using synthesized virtual images, thereby illustrating the potential usefulness of virtual image datasets in image-to-image translation models.

Third, we implemented a multi-domain translation model based on the StarGAN, incorporating skip connections that can be used for convenient interior style transformation.

3 PROCEDURAL GENERATION OF VIRTUAL INTERIOR IMAGES

Initially, we implemented a process to generate virtual indoor scene images of certain interior styles using Unreal Engine. Our algorithm creates the living room scene by looking at the window and depicting four interior styles: classic, modern, natural, and romantic. After defining standards for each style, we created libraries of 3D furniture objects and materials that meet each interior style. Then, virtual living room layouts were generated with randomized flooring and wallpaper combinations. Furniture arrangement, 3D mesh selection, and material application were randomized under the designated style. Rapid, real-time rendering was employed to capture lighting and shadows, and changeable camera angles captured the scenes as saved images. Finally, we produced virtual indoor scene image datasets, each representing the four distinct interior styles.

3.1 Interior Style Classification

Interior style is defined by the overall ambiance created by the interplay of color, form, and texture. The interior elements such as flooring, walls, ceiling, furniture, and lighting must adhere to a specific trend to classify a style. Thus, we established classification criteria for the classic, modern, natural, and romantic styles (Table 1(a), (b)) based on the work of Min and Choi [16]. The classic style incorporates luxurious decorations, traditional patterns, and deep, dark colors. The modern style is characterized by clean, straight designs, minimal ornaments, and achromatic colors. The natural style employs natural materials like wood, stone, and plaster. It frequently includes wooden furniture with checkered fabrics. Finally, the romantic style incorporates its sensitivity to curves by using light, warm colors and floral patterns.

The 3D objects for each furniture type include chairs, couches, drawers, carpets, tables, TVs, TV stands, plants, and picture frames. These were collected from various websites. Materials such as flooring and wall textures and wood and fabric surfaces were also collected. These were organized into libraries according to our interior style classification standards. We found that some furniture meshes could be categorized in more than one style. For example, some meshes and patterns may meet the criteria for both the classic and romantic styles or both the modern and natural styles.

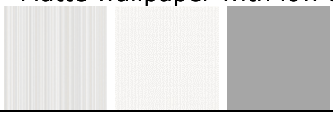
Style	*Components*	*Features*
Classic	Colors	• Deep and dark colors: dark brown, magenta, crimson, beige
	Floor	• Marble or dark timber with low brightness 
	Wall	• Traditional gold and yellow patterns with deep background • Bright orange or yellow marble 
	Furniture	• Dark wood with gold ornaments • Detailed edges with decorative curves • Antique and diverse marble
	Texture	• Leather and velvet are widely used for texture
	Props	• Pottery, painting, and candlesticks are popular
Modern	Colors	• Achromatic colors, blue and green with low saturation
	Floor	• Achromatic marble or tiles • Translucent timber colored with ivory or gray 
	Wall	• Matte wallpaper with low saturation 
	Furniture	• Straight and simple design without ornaments • Cold materials such as metal and stone
	Texture	• Geometrical or simple, bold patterns • Cotton, linen, or leather with monotone
	Props	• Props with simple shape props

Table 1(a): Interior style classification standard (classic, modern style).

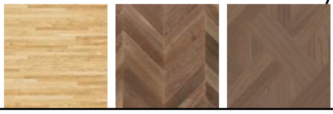
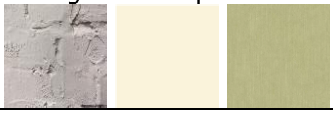
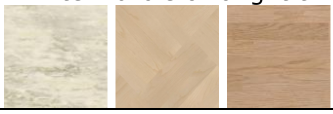
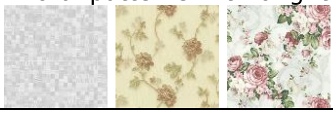
Style	*Components*	*Features*
Natural	Colors	• Natural and comfortable colors: green, brown, ivory, and beige
	Floor	• Vivid wooden floor of varying brightness 
	Wall	• Wooden or beige, light green wallpaper • Rough cast or plaster 
	Furniture	• Simple and crude design • Mostly hardwood materials
	Texture	• Natural fabrics, including cotton, silk, and linen • Flower or leaf, calm checked patterns are often used
	Props	• Plants and livelihood props like quilts and baskets
Romantic	Colors	• Bright and warm colors: red or orange with white background
	Floor	•White marble or bright timber 
	Wall	•Diverse materials in white, pink, and purple •Floral patterns with bright background 
	Furniture	•Mostly bright wood or marble materials sometimes painted in pastel colors. •Curved form and decorations
	Texture	•Soft and bright cotton, linen, and leather
	Props	•Flowers, candlesticks, and quilts

Table 1(b): Interior style classification standard (natural, romantic style).

3.2 3D Interior Object Generation

3.2.1 Living room layout generation

We generated 3D living room layouts consisted with wall, floor, ceiling, and window objects using Unreal Engine. To adjust object placement and diversify materials, we employed the Unreal Engine blueprint system. This system enables real-time interactive programming by linking nodes to set specific functions and actions between components. Node connections define data flow, operation sequence, and object behaviors such as spawning or animations. We customized a blueprint to control 3D object properties like location, scale, and material (Figure 1). Elements were sequentially generated, starting with the floor shown in figure 1, followed by ceilings, walls, and windows in the same manner. The layout adhered to the standard dimensions of a Korean apartment living room: 4.5 meters in width and 2.5 meters in height. Flooring and wallpaper were combined randomly from the material library according to each style. Using the LDAssistant plugin, we duplicated the layout 100 times at 10-meter intervals along the x-axis.

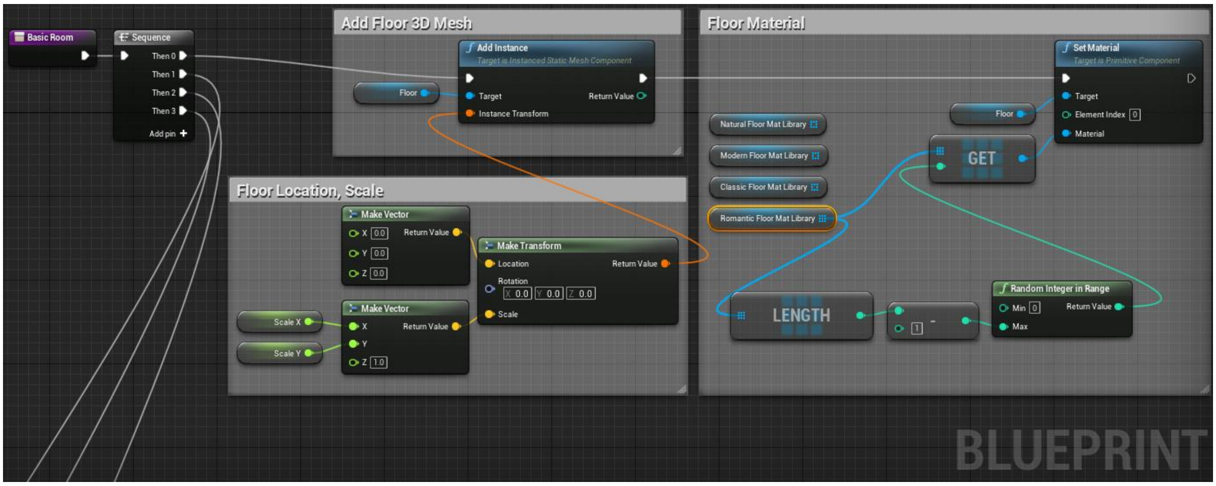


Figure 1: Unreal Blueprint script generating living room layout (floor object).

3.2.2 Furniture arrangement and material selection

We modified a blueprint script to randomly generate 3D furniture objects in the living rooms (Figure 2). The script randomizes the location, rotation, and appearance of each object by selecting its mesh and material. Initially, furniture creation points were duplicated and distributed along the x-axis to appear in each living room. These points were then arbitrarily shifted along the x and y coordinates. A 3D mesh was selected from the mesh library, and the object could be freely rotated if necessary. We adjusted the frequency of appearance of each type of furniture (Table 2) because not all furniture items are represented in the real world. It was adjusted by using an "empty" value in the mesh library for random selection. Primary materials, like couch textiles, were chosen from the material library based on style.

We tailored the blueprint for each furniture type and placed items in the living rooms according to intuitive locations. For example, couches were positioned along the wall facing the TV, while tables and carpets were centrally located (Figure 3). These varied combinations of furniture meshes, materials, and locations can create diverse interior scenes.

3.2.3 Lighting and camera setup

Internal lighting in the living rooms was adjusted by changing light temperature according to the styles: 6000K for modern and romantic, 5500K for natural, and 4500K for classic. Cameras were placed in the center of the room to capture the windows. We also randomized the location, rotation, and wide angles of the camera perspectives in each living room to mimic the diversity of real images.

Type	TV	TV stand	Carpet	Table	Couch
Appearance Frequency	0.66	0.60	0.55	0.66	0.70
Type	Chair1	Chair2	Drawer	Plants	Picture Frame
Appearance Frequency	0.60	0.30	0.55	0.4	0.2

Table 2: Frequency of appearance of each furniture item category.

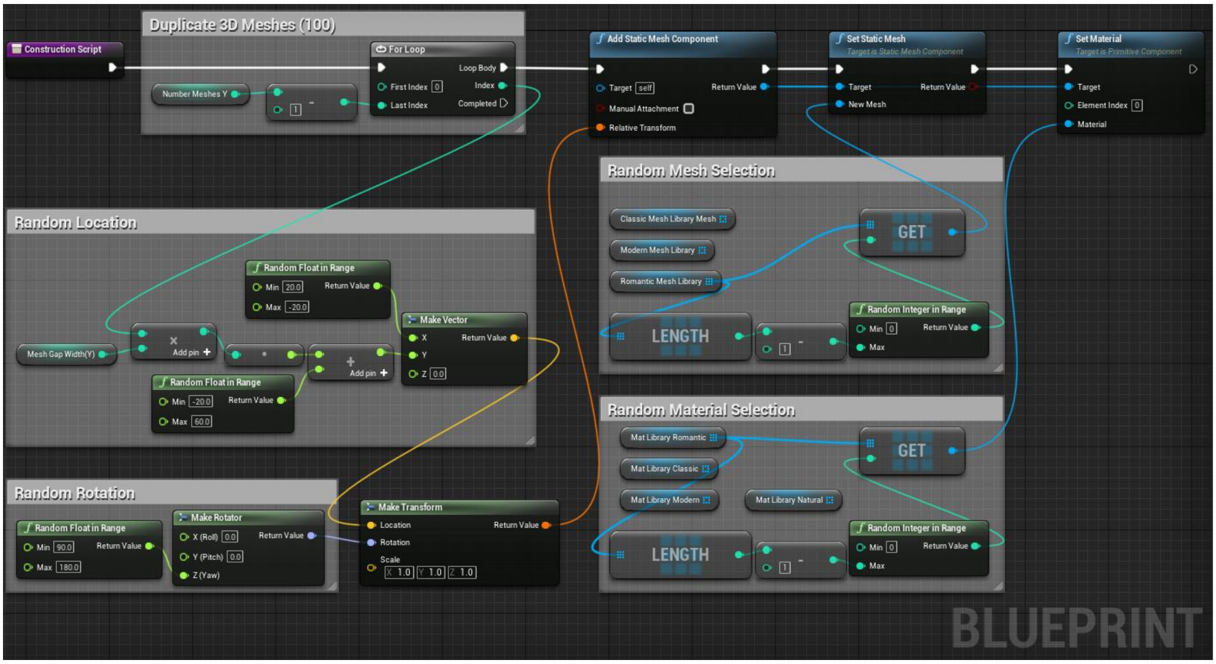


Figure 2: Unreal Blueprint script generating 3D furniture objects.

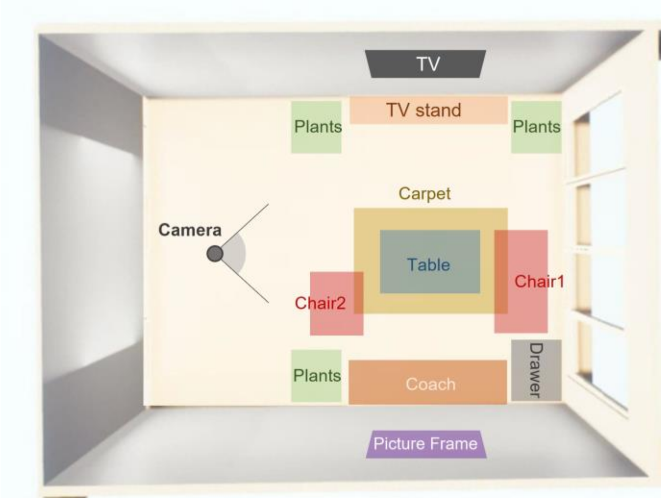


Figure 3: Furniture arrangement in the living room.

3.3 Virtual Image Generation

After creating 3D indoor scenes, cameras automatically capture the scene upon activation. Our generation system creates 100 images per operation, taking only 0.7 seconds per image in our computer with NVIDIA RTX 3700 Ti, Intel Core i9 10900X, and 48GB RAM. By running the capture process multiple times, thousands of virtual images could be captured quickly. We created 8,000 virtual images, 2,000 images for each style, with a unique combination of placement, materials, and scene compositions (Figure 4).

4 INTERIOR STYLE TRANSLATION USING GENERATIVE ADVERSARIAL NETWORKS

4.1 Dataset Construction

4.1.1 Dataset augmentation

Our raw virtual images exhibit consistent brightness, contrast, and horizontal layout, with the TV positioned on the left and the couch on the right. Dataset augmentation with horizontal flipping and color-shifting techniques was employed for the virtual image dataset for a more realistic appearance. Brightness was randomly adjusted between 0.85 and 1.05 times the original image. Contrast was randomly adjusted between 0.9 and 1.1 times.

4.1.2 Real image dataset collection

To construct a dataset with real images, living room scenes facing the window were collected through web crawling with classification in the four interior styles. Ultimately, we collected 160 classic, 320 modern, 256 natural, and 128 romantic training images with a few test images. Finally, we created three training datasets to compare their training performance: a virtual image dataset with 8,000 virtual images, a combined image dataset with 7,136 virtual images and 864 real images, and a real image dataset with 864 real images only.

4.2 Constructing a Generative Adversarial Networks

Our objective was to design a generator and discriminator network capable of simultaneously transporting input interior images into various style domains. The StarGAN model, equipped with an auxiliary classifier on the discriminator, facilitates the transformation of input images into several other styles. The discriminator calculates the authenticity probability and the likelihood of each image belonging to a particular interior style domain, $D : x \rightarrow D_{auth}(x), D_{cls}(x)$. The generator receives the input image and style vector and down samples them using a convolution layer, $G(x, c) \rightarrow y$. After passing through the residual blocks, layers are upsampled with transposed convolution layers to produce the transformed target domain image. Selective skip connections between low-level and high-level layers provide location detail. The model was constructed with the Tensorflow package in Python 3.9.

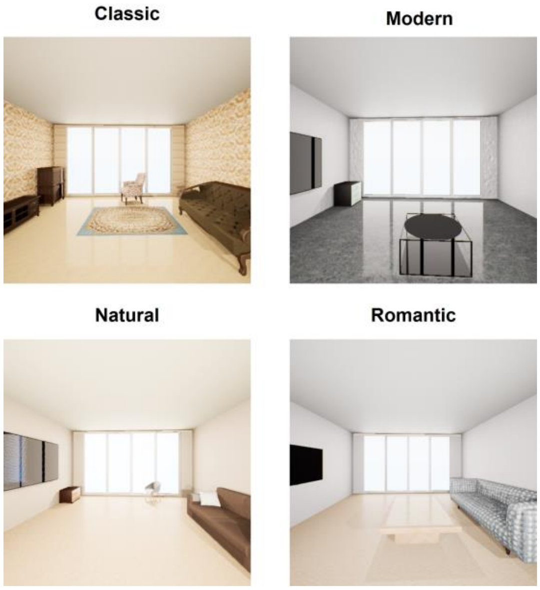


Figure 4: Synthesized virtual images for each style.

The generator should translate images that are realistic and congruent with the target style and based on the original image's shape. The cycle-consistency concept was implemented to ensure realism and preserve input image identity.

4.2.1 Network structure

We implemented improvements so the generator captures interior features and patterns with greater accuracy. In the original StarGAN generator with single-scale residual blocks, information had to pass through a bottleneck, limiting the amount of learning place information [6]. We inserted residual blocks at multiple layers of varying scales to improve multi-scale transition and overall image quality. Additionally, we added skip connections to preserve the semantic information of low-level features. The structure of our generator is illustrated in Figure 5.

4.2.2 Object loss function

The StarGAN algorithm suggested three objectives during the training of the generator and discriminator [3]. The first objective was to generate images the discriminator cannot distinguish from real images. Adversarial loss facilitates a competitive dynamic between the two networks. Generator adversarial loss was computed from the authenticity probability of fake images, $G(x,c)$, provided by the discriminator (Equation 1). The discriminator also tried to accurately distinguish real and fake images by maximizing discriminator adversarial loss (Equation 2).

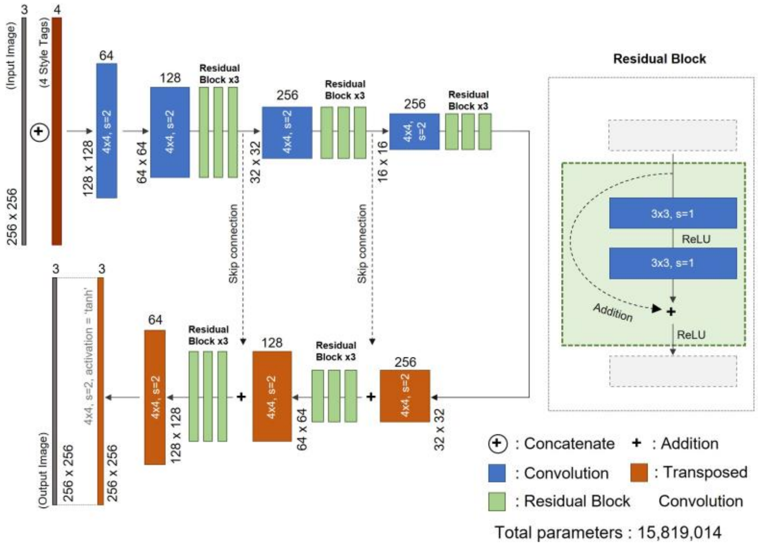


Figure 5: Generator network structure.

$$L_{adv}^{G} = E_{x,c}[\log(1 - D_{auth}(G(x,c)))] \quad (1)$$

$$L_{adv}^{D} = E_{x}[\log D_{auth}(\mathrm{x})] + E_{x,c}[\log(1 - D_{auth}(G(x,c)))] \quad (2)$$

The second objective was to create images suitable for the target domain c, which the discriminator can classify accurately. Domain classification loss was optimized to enhance the accuracy for transformation to the target domain. The classification results of fake images were numerically expressed as fake image classification loss (Equation 3), which was optimized by the generator. The discriminator learned to distinguish the interior style of input images by optimizing real image classification loss (Equation 4).

$$L_{cls}^{f} = E_{x,c}[-logD_{cls}(c|G(x,c))] \tag{3}$$
$$L_{cls}^{r} = E_{x,c'}[-logD_{cls}(c'|x)] \tag{4}$$

Minimizing adversarial and domain classification loss for the generator does not guarantee the identity of the original image after transformation. Introducing the concept of cycle consistency can achieve the last objective of preserving the identity of the original image although translation is performed. For the objective, reconstruction loss was calculated as the L1 norm between the original image and the twice-transformed image to the target domain c, followed by the original domain c' (Equation 5).

$$L_{rec} = E_{x,c,c'}[||x - G(G(x,c),c')||_1] \tag{5}$$

Additionally, we incorporated a gradient penalty into the objective function of the discriminator to avert the gradient vanishing issue (Equation 6). The interpolated images are represented as $\hat{x}$ were created randomly between pairs of real and fake images. The total objective function of the generator and discriminator are shown in Equation (7) and Equation (8), respectively. They are the sum of the weighted losses with weight parameters as $\lambda_{cls} = 1$, $\lambda_{rec} = 7$, and $\lambda_{gp} = 10$.

$$L_{gp} = E_{\hat{x}}[(||\nabla_{\hat{x}}D_{auth}(\hat{x})||_2 - 1)^2] \tag{6}$$
$$L_G = L_{adv}^{G} + \lambda_{cls}L_{cls}^{f} + \lambda_{rec}L_{rec} \tag{7}$$
$$L_D = -L_{adv}^{D} + \lambda_{cls}L_{cls}^{r} + \lambda_{gp}L_{gp} \tag{8}$$

Finally, the generator and discriminator were trained to minimize their loss functions, which were calculated by authentication and classification errors alongside reconstruction loss.

4.2.3 Training process

The training set images were resized to 256×256 pixels, which preserved the details of small objects and patterns. Each image in the training set was assigned a style classification label using one-hot encoding. The datasets were divided into batches of 16.

We trained the models for 700 epochs for the virtual and combined image datasets and 6,000 epochs for the real image dataset. The training process took approximately a day to complete using an NVIDIA RTX 3700 Ti GPU. The learning rate was set to 0.0001, and the beta for the ADAM optimizer was 0.5.

5 RESULTS AND DISCUSSION

The synthesized virtual images exhibited clear characteristics of each interior style in the living room with diverse combinations of furniture arrangement, material, and camera composition. From these images, the model learned the consistent colors, patterns, and shapes of furniture, flooring, and wallpaper to create the overall ambiance of the four interior styles by defined standards. Also, unique virtual image combinations helped the model learn the transition of diverse cases to reduce failure.

The loss functions performed as expected upon training the GAN model using virtual images. The adversarial loss fluctuated within a normal range (Figure 6(a)), while other losses steadily approached zero as the training epoch increased (Figure 6(b)). These findings indicate that the model was successfully trained to achieve appropriate style transformations of the input images.

After we trained the virtual image dataset, the model converted input images as practical outputs of targeted styles fast with changing texture and color (Figure 7). The model successfully recognized the boundary of the floor, wall, and furniture for transformation. Overall light temperature and hue were reflected to translate into other interior styles.

In the other cases where real images were mixed, characteristics of interior styles generated by the model were less obvious than in the virtual image dataset case (Figure 8). The combined image dataset model converted light temperature and color ambiguously and showed some failure to recognize boundaries, especially in modern style input. In the model trained with only real images, image quality degradation with broken shapes and unclear targeted style characteristics occurred due to insufficient training images (Figure 9).

Mixing real images in the dataset can weaken the interior style clarity of output because they seldom meet the exact interior style standards, despite the use of strict collection and sorting practices. On the other hand, virtual images reflect more accurate style characteristics based on defined standards.

As a result, our procedural generation algorithm built on the Unreal Engine rapidly produced virtual images of various interior styles. The multi-domain StarGAN model enhanced with scaled residual blocks and skip connections transformed the style of an indoor scene image within a second. Also, the model trained on the virtual images was the most accurate method for our purposes. Our research underscores the feasibility of using synthesized virtual images for training deep learning models in interior design. This method supplements limitations of real images such as insufficiency and design inconsistency.

Our model offers practical applications for 3D modeling and Building Information Modeling (BIM) by enabling the rendering of diverse interior styles without altering materials. Interior service platforms can use this model to offer customers with personalized style options for indoor scenes. The model quickly generates images without text prompts unlike general-purpose generative AIs like Midjourney and DALL·E2, so it is fast and convenient.

Our procedural generation algorithm is currently limited to living room space, specific window views, and four interior styles. Future work should extend this algorithm to other living spaces including bedrooms, kitchens, and offices. This extension could build upon previous studies like that of Feklisov et al. [4], which generated whole-house layouts. Small props such as books, flowers, and clothes can be added for realism. Increasing style options and creating a robust database of meshes and materials would also enhance usability.

Our StarGAN model operates at a 256×256 resolution due to computational constraints. While suitable for mobile and web applications, higher resolutions are required for professional rendering. This could be achieved by resizing the input and output layers. Furthermore, future versions of the model should focus on more dramatic transformations of furniture textures, forms, and ornaments. Employing an independent classifier instead of an auxiliary classifier in the discriminator could yield more detailed feature changes [13].

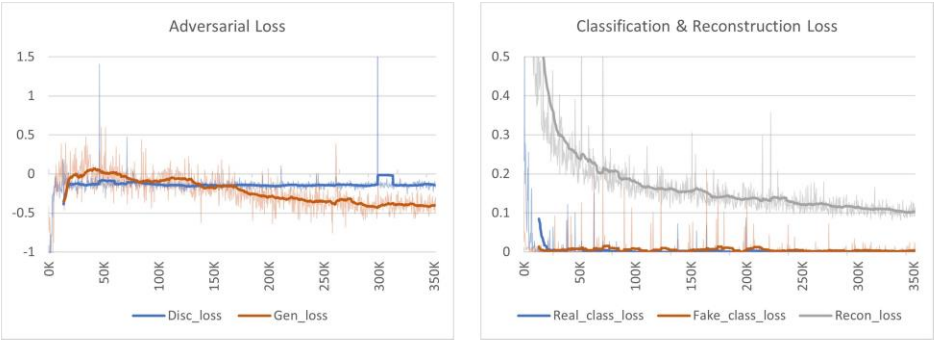


Figure 6: Change of losses during training: (a) adversarial loss, (b) domain classification loss, and reconstruction loss.

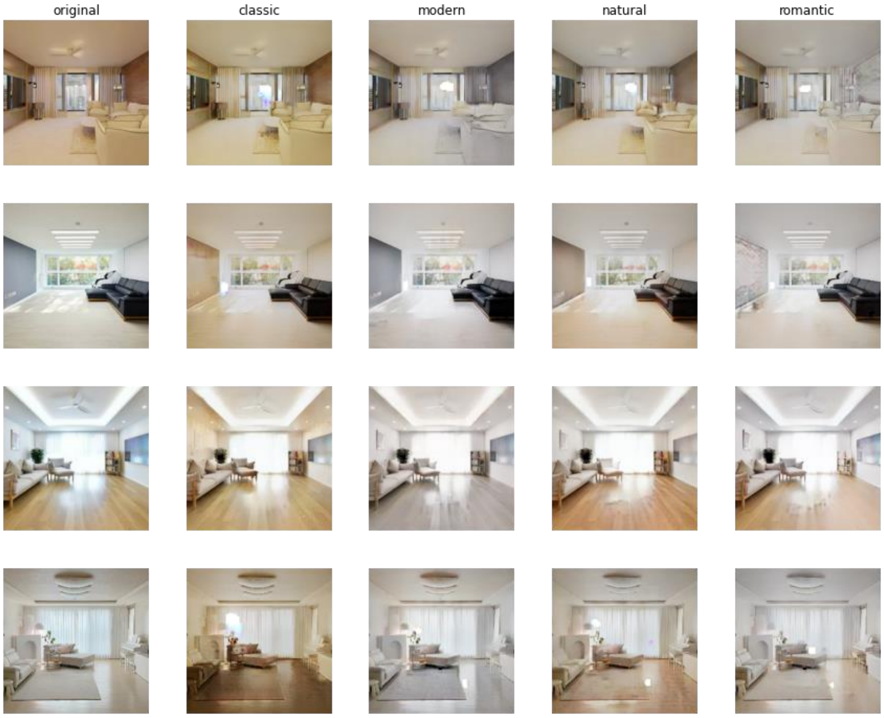


Figure 7: Model output trained with only virtual images.

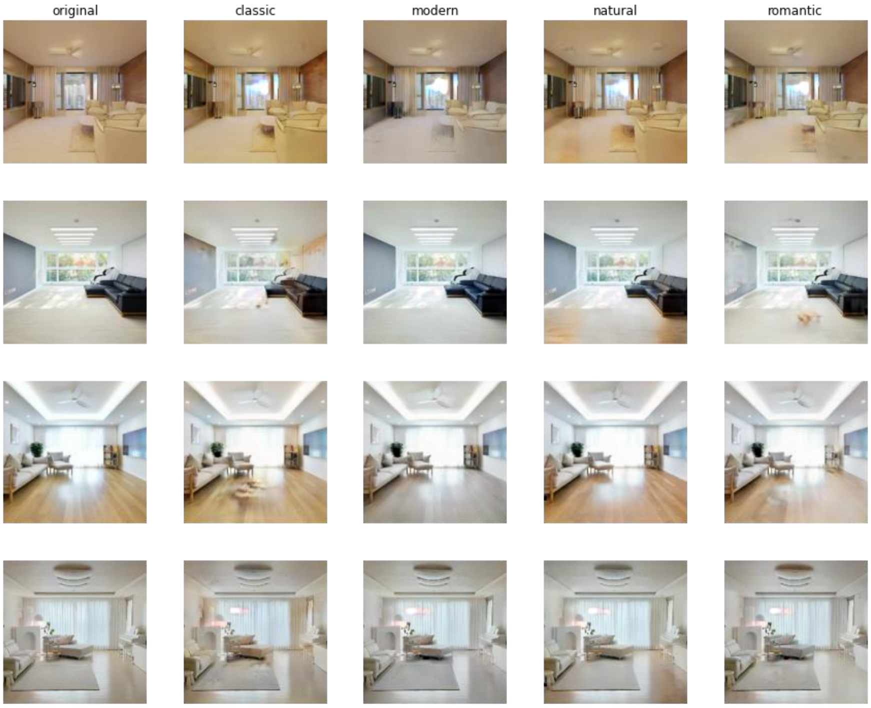


Figure 8: Model output trained with combined dataset of virtual and real images.

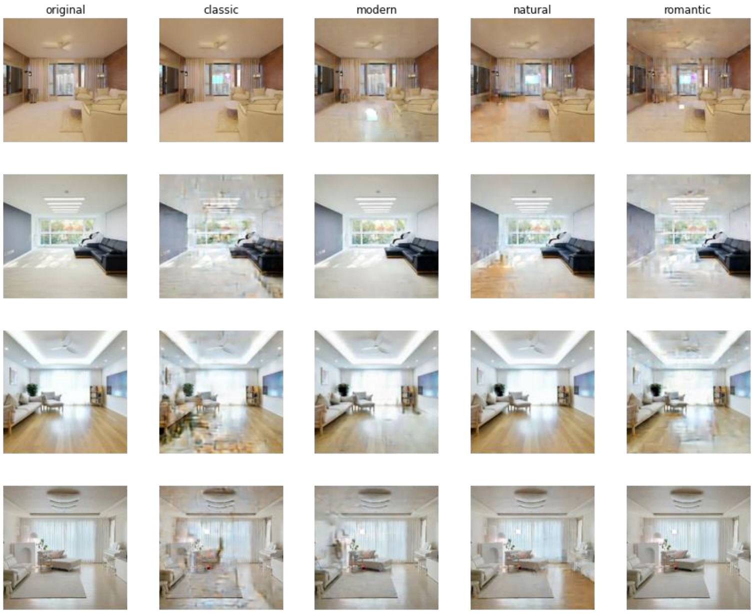


Figure 9: Model output trained with only real images.

6 ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF), which is funded by the Korean government (MSIT) (No. 2021R1F1A1047492) and the SNU Environmental Planning Institute.

Junhyeck Choi, http://orcid.org/0009-0005-4380-525X
Jaeseung Lee, http://orcid.org/0000-0001-9087-5975

REFERENCES

[1] Berman, B.; Pollack, D.: Strategies for the successful implementation of augmented reality, Business Horizons, 64(5), 621-630, 2021. https://doi.org/10.1016/j.bushor.2021.02.027.
[2] Choi, J.H.; Lee, J.S.: A study of interior style transformation with GAN model, Journal of KIBIM, 12(1), 55–61, 2022. https://doi.org/10.13161/kibim.2022.12.1.055.
[3] Choi, Y.; Choi, M.; Kim, M.; Ha, J.W.; Kim, S.; Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation, Proceedings of the IEEE conference on computer vision and pattern recognition, 8789-8797, 2018. https://doi.org/10.1109/cvpr.2018.00916.
[4] Feklisov, E.; Zingerenko, M.; Frolov, V.; Trofimov, M.: Procedural interior generation for artificial intelligence training and computer graphics, CEUR-WS, 269-276, 2020. https://doi.org/10.30987/conferencearticle_5fce2771c14fa7.77481925.
[5] Gao, Q.; Shen, X.; Niu, W.: Large-scale synthetic urban dataset for aerial scene understanding, IEEE Access, 8, 42131-42140, 2020. https://doi.org/10.1109/ACCESS.2020.2976686.
[6] Gokaslan, A.; Ramanujan, V.; Ritchie, D.; Kim, K.I.; Tompkin, J.: Improving shape deformation in unsupervised image-to-image translation, Proceedings of the European Conference on Computer Vision (ECCV), 649-665, 2018. https://doi.org/10.1007/978-3-030-01258-8_40.

[7] He, K.; Zhang, X.; Ren, S.; Sun, J.: Deep residual learning for image recognition, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 770-778, 2016. https://doi.org/10.1109/cvpr.2016.90.
[8] Henderson, P.; Subr, K.; Ferrari, V.: Automatic generation of constrained furniture layouts, arXiv:1711.10939, 2017.
[9] Jaipuria, N.; Zhang, X.; Bhasin, R.; Arafa, M.; Chakravarty, P.; Shirivastava, S.; Manglani, S.; Murali, V.: Deflating dataset bias using synthetic data augmentation, Proceedings of the IEEE/CVF Conference on Conputer Vision and Pattern Recognition Workshops, 772-773, 2020. https://doi.org/10.1109/cvprw50498.2020.00394.
[10] Kán, P.; Kaufmann, H.: Automated interior design using a genetic algorithm, Proceedings of t he 23rd ACM Symposium on Virtual Reality Software and Technology, 1-10, 2017. https://doi .org/10.1145/3139131.3139135.
[11] Kim, S.K.: A study on the development plan of the domestic home furnishing industry, Journa l of the Korean Furniture Society, 29(4), 228-244, 2018. https://doi.org/10.22873/kofuso.20 18.29.4.228.
[12] Kleiner, F.S.: Gardner's art through the ages: A global history, Cengage Learning, 2015.
[13] Ko, K.; Yeom, T.; Lee, M.: Superstargan: Generative adversarial networks for image-to-imag e translation in large-scale domains, Neural Networks, 162, 330-339, 2023. https://doi.org/1 0.1016/j.neunet.2023.02.042.
[14] Koo, Y.S.; Yoon, C.: Analysis of home furnishing trends caused by the COVID-19 pandemic, I nternational Journal of Costume and Fashion, 21(2), 80-90, 2021. https://doi.org/10.7233/ij cf.2021.21.2.080.
[15] Li, X.; Li, H.; Wang, C.; Hu, X.; Zhang, W.: Visual-attention GAN for interior sketch colorization, IET Image Processing, 15(4), 997-1007, 2021. https://doi.org/10.1049/ipr2.12080.
[16] Min, J.H.; Choi, G.S.: Study on color & texture image of finishing material by interior compon ent, Journal of Korea Society of Color Studies, 28(4), 75-87, 2014. https://doi.org/10.17289 /jkscs.28.4.201411.75.
[17] Ritchie, D.; Wang, K.; Lin, Y.A.: Fast and flexible indoor scene synthesis via deep convolutional generative models, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6182-6190, 2019. https://doi.org/10.1109/cvpr.2019.00634.
[18] Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M.: The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3234-3243, 2016. https://doi.org/10.1109/cvpr.2016.352.
[19] Seib, V.; Lange, B.; Wirtz, S.: Mixing real and synthetic data to enhance neural network working. arXiv:2007.08781, 2020.
[20] Wang, K.; Savva, M.; Chang, A.X.; Ritchie, D.: Deep convolutional priors for indoor scene syn thesis, ACM Transactions on Graphics (TOG), 37(4), 1-14, 2018. https://doi.org/10.1145/31 97517.3201362.
[21] Xiong, X.Q.; Guo, W.J.; Fang, L.; Zhang, M.; Wu, Z.H.; Lu, R.; Miyakoshi, T.: Current state and development trend of the Chinese furniture industry, Journal of Wood Science, 63, 433-444, 2017. https://doi.org/10.1007/s10086-017-1643-2.
[22] Yang, F.; Lu, Z.; Qiu, G.; Zhang, Q.: Capsule-based image translation network. IET Doctoral Forum on Biomedical Engineering, Healthcare, Robotics and Artificial Intelligence 2018 (BRAIN 2018), Ningbo, China, 1-8, 2018. https://doi.org/10.1049/cp.2018.1723.
[23] Yu, L.F.; Yeung, S.K.; Tang, C.K.; Terzopoulos, D.; Chan, T.F.; Osher, S.J.: Make it home: automatic optimization of furniture arrangement, ACM Transactions on Graphics (TOG), 30(4), 1-12, 2011. https://doi.org/10.1145/1964921.1964981.
[24] Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks, Proceedings of the IEEE International Conference on Computer Vision, 2223-2232, 2017. https://doi.org/10.1109/iccv.2017.244.