Adjustable Visual Appearance for Generalizable Novel View Synthesis

Josef Bengtson
Computer Vision Group,
Chalmers University of Technology
bjosef@chalmers.se

David Nilsson
Computer Vision Group,
Chalmers University of Technology
david.nilsson@chalmers.se

Che-Tsung Lin
Computer Vision Group,
Chalmers University of Technology
chetsung@chalmers.se

Marcel Büsching
KTH Royal Institute of Technology
busching@kth.se

Fredrik Kahl
Computer Vision Group,
Chalmers University of Technology
fredrik.kahl@chalmers.se

Abstract

We present a generalizable novel view synthesis method where it is possible to modify the visual appearance of rendered views to match a target weather or lighting condition without any scene specific training. Our method is based on a generalizable transformer architecture and is trained on synthetically generated scenes under different appearance conditions. This allows for rendering novel views in a consistent manner for 3D scenes that were not included in the training set, along with the ability to (i) modify their appearance to match the target condition and (ii) smoothly interpolate between different conditions. Experiments on real and synthetic scenes show that our method is able to generate 3D consistent renderings while making realistic appearance changes, including qualitative and quantitative comparisons with applying 2D style transfer methods on rendered views. Please refer to our project page for video results: https://ava-nvs.github.io

1. Introduction

The field of novel view synthesis has seen rapid progress in the last few years after the success of Neural Radiance Fields (NeRFs) [25] and follow-up works [1, 38, 54]. A desired quality for these types of 3D scene representations is to be able to disentangle different scene properties from each other, for instance, being able to change the visual appearance without changing the content of the scene. There

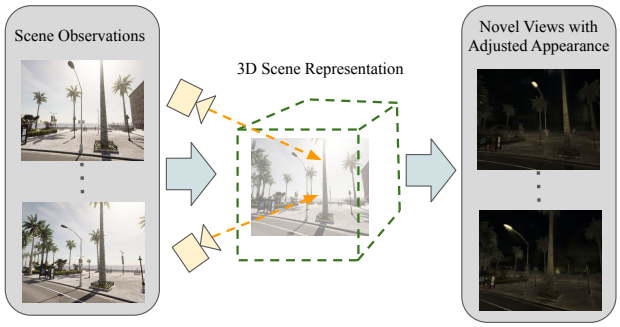

Figure 1. Given multiple views of a scene in one (and only one) weather and lighting condition, we want to generate novel views of the given scene with adjusted visual appearance corresponding to a target condition without the need for scene specific optimization.

exists some works in this direction [24,42], but they are limited to interpolating between *observed* visual appearances of the 3D scene, thus requiring images of the scene with the desired visual appearance. In contrast, we develop a method that is able to generalize to 3D scenes not used in training, and that thus can adjust the appearance of a scene without having access to any images of that scene at the target visual appearance, see Fig. 1.

For traditional NeRF-based methods, the properties of the 3D scene are encoded in the weights of a multilayer perceptron (MLP), so each trained model is exclusive to that particular scene. A main challenge is thus that a separate optimization process has to be performed for each individual scene. One approach to handle this is to find ways to

improve the efficiency of the training process [21,26,48]. A different approach is to avoid per-scene training and instead train cross-scene generalizable methods [6, 23, 41, 49, 52], which are able to synthesize novel views of a scene given just images and corresponding camera poses, and do not require expensive scene-specific optimization.

We present a generalizable novel view synthesis method that allows for changing the visual appearance of a scene while ensuring multi-view consistency. For this we build upon Generalizable NeRF Transformer (GNT) [41], a transformer [44] based novel view synthesis method. Specifically, we introduce a latent appearance variable to enable the control of the visual appearance of rendered views. By using a generalizable NeRF model and the introduced latent appearance variable, we are able to render novel views and change the appearance of scenes that are not seen when training our model without the need for observations of the scene at the target appearance.

Image-translation has made progress in recent years. From GANs [7, 19, 50] to Diffusion models [4, 22, 34], the quality of individual images has significantly improved with better structure-preservation, fewer artifacts, and lessened constraints. However, without ensured multi-view consistency, temporal fluctuations are still noticeable, and appearance changes are inconsistent, e.g. turning on street lamps for only some views.

In summary, our main contributions are:

- We introduce a method that allows for changing the appearance of a novel scene, while ensuring multi-view consistency, by using a latent appearance variable conditioned on a target visual appearance.
- We propose a novel loss function which is designed to align the views rendered with a target appearance to the scene observed in that target condition, which enables jointly learning novel view synthesis and appearance change.
- We create a synthetic dataset containing urban scenes, with each scene available at four different diverse weather and lighting conditions. The dataset is used for training our model for visual appearance change and enables quantitative evaluation. The dataset will be made publicly available.

Limitations. Our work is restricted to realistic weather and lighting conditions, and we do not consider arbitrary appearance changes. Furthermore, our novel view synthesis is limited to the performance of the neural rendering method we base our method on. Progress on neural rendering can be translated to improvements to our method.

2. Related work

Here we will review progress on NeRFs, focusing on works related to generalizable novel view synthesis. We will then review 2D style transfer methods and stylized NeRFs methods.

Neural Radiance Fields. NeRFs [25] synthesize consistent and photo-realistic novel views of a scene, by representing each scene as a continuous 5D radiance field parameterized by an MLP mapping 3D positions and 2D viewing directions to volume densities and view-dependent emitted radiances. Views are synthesized by querying points along camera rays and map the output colors and densities into RGB values. There have been several works improving NeRFs further, e.g. to improve efficiency of the training process [21, 26, 48] and enable handling a low number of input views [20, 27, 52].

Generalizable Novel View Synthesis. The original NeRF methodology is constrained to training a neural network for representing a single scene, which requires a time-consuming optimization process performed from scratch for each new scene, without leveraging any prior knowledge. Methods for generalized neural rendering address this limitation by training across multiple scenes, enabling the learning of a general understanding of how to utilize source observations to synthesize novel views of the scene. Earlier methods such as [6, 52] use a multilayer perceptron (MLP) conditioned on feature vectors extracted from the source images to predict color and radiance values which are aggregated with volumetric rendering. To enhance generalization capabilities and rendering quality, recent approaches have incorporated transformer-based architectures [10, 44] for feature aggregation from the source images [23, 47], computing densities along the camera ray [49], and even for the entire rendering pipeline [12,31,35,39–41]. While these methods have demonstrated impressive rendering quality, they are currently incapable of modifying the appearance of the rendered views.

2D Style Transfer. Advances in 2D style transfer have largely been driven by the success of GANs [14]. Pix2Pix [19] and Pix2Pix-HD [50] are two methods that provide visually plausible images if paired training data is available, i.e., one image in the source condition and the corresponding image in the target condition. Under the same constraint, BicycleGAN [57] can provide diverse images matching a target domain. CycleGAN [56] introduced the cycle-consistency constraint to learn from unpaired data. CyEDA [2] further proposes cycle-object edge consistency to achieve better image-object preservation. NICE-GAN [7]

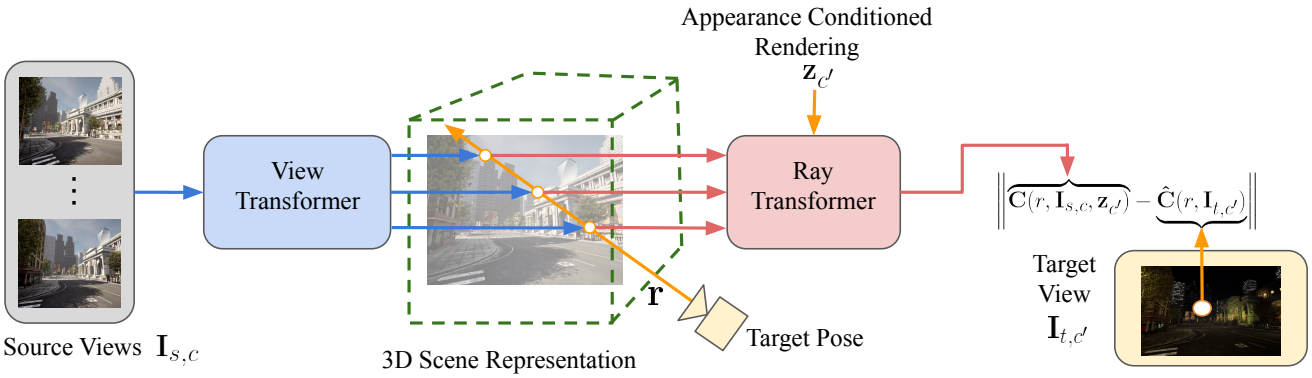

Figure 2. Overview of our method for changing visual appearance of synthesized novel views. A target view direction is chosen and camera rays $\mathbf{r}$ are cast and the corresponding source views $\mathbf{I}_{s,c}$ are used to generate a scene representation. A latent appearance variable $\mathbf{z}_{\mathbf{c}'}$ is included with the goal of adapting the appearance of the rendered image to match the target view. If the target view is at a different weather or daylight conditions ($c \neq c'$) then this means adapting the visual appearance to match that found in the target view $\mathbf{I}_{t,c'}$ instead of the visual appearance of the source views $\mathbf{I}_{s,c}$.

reuses the discriminator for encoding the images of the target domain, thus leading to better results. Besides GANs, style-attentional network (SANet) [29] can synthesize a content image with the style of another image. Recently, diffusion models [9, 16, 37] have achieved better results in image generation. Palette [34] proposed the first diffusion based paired image-translation model. More recently, DiffuseIT [22] presented a novel diffusion-based unsupervised image translation method that can disentangle style and content representations. Instruct-Pix2Pix [4] combined a text-to-image model (Stable Diffusion [32]) and a language model (GPT-3) [5] to enable instruction-based style transfer. Although the images translated by these 2D methods can be realistic individually, temporal consistency is not ensured when the input images come from a sequence of consecutive images. In contrast, our method has built-in 3D consistency, and we experimentally compare our results with 2D style transfer methods applied frame by frame on rendered views.

Visual Appearance Change for NeRF Models. Prior work to enable changing the visual appearance of a NeRF model [24, 42] builds on Generative Latent Optimization (GLO) [3], where each image is assigned to an appearance embedding vector. This embedding vector is included as input to the part of the NeRF network responsible for colors, enabling it to affect the visual appearance but not the geometry, and it is optimized alongside the NeRF model parameters. In [24], they choose a low dimensionality for these embeddings to find a continuous space, allowing for smooth interpolation between lighting conditions. One limitation of this approach is that it only allows for interpolation between two already-seen lighting conditions for a given scene, requiring access to images of that scene at both lighting conditions as input. In contrast, our method is a generalizable method that does not require images at both lighting conditions as input when rendering novel views with changed visual appearance.

Stylized NeRFs. Another line of research is to change the style of a NeRF model based on a given style prompt [15, 17, 18, 53] typically given in the form of a reference image. More recent works [45, 46] use the joint language-image embedding space of the CLIP model [30] to enable specifying the desired style using a text prompt. These methods are all based on per-scene NeRF models and are not generalizable. They focus on performing artistic style changes and have thus not been specifically trained and evaluated on realistic appearance changes such as differences in weather or lighting. In contrast, our method is generalizable and able to handle realistic appearance changes.

3D Scene Generation with controllable appearance. GRAF [36] and GIRAFFE [28] are generative methods for synthesizing radiance fields while introducing an embedding that allows for controlling the shape and appearance of the generated scenes. They enable controllable image synthesis but do not make it possible to generate novel views and change appearance from a given set of scene images. In contrast, our method enables changing the appearance of a given scene to a given appearance condition and also interpolating between these conditions.

3. Method

We present our proposed method for novel view synthesis of a scene while changing the visual appearance. We

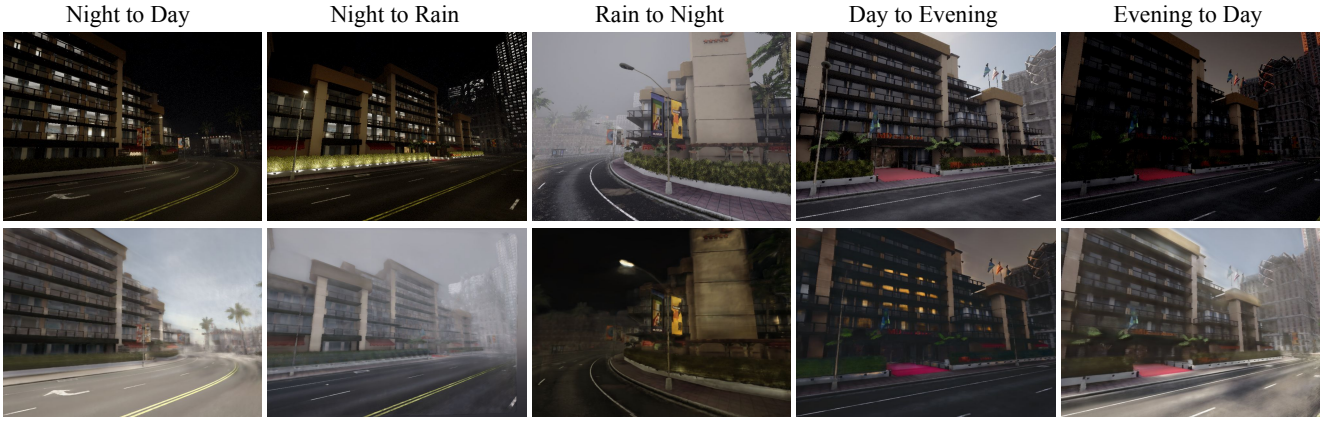

Figure 3. Top: Input images generated in CARLA [11]. Bottom: Images generated by our method, where images from one condition are given as input along with a latent appearance variable corresponding to another condition. We can see that our method is able to change the overall visual appearance of the images to match the desired condition while also making local changes, such as turning on a lamp when changing the appearance to night or adding sunlight reflection on windows when changing the appearance into evening. We also observe that changing the appearance to match the day condition is more challenging, leading to blurriness in the regions of the image that had low visibility in the original condition.

start by giving an overview of Generalizable NeRF Transformer (GNT) [41] and then describe our method for adjusting the visual appearance of synthesized views, along with the visual appearance change loss term we introduce.

3.1. Basics of GNT

GNT utilizes a two-stage transformer-based architecture that allows for efficient novel view synthesis from source views. The first stage is a *view transformer* that aggregates information from neighboring views while using epipolar geometry as an inductive bias. The second stage is a *ray transformer* that performs a learnable ray-based rendering based on the feature vectors generated by the *view transformer* along points on a camera ray.

View Transformer. The internal representation of the method is a coordinate aligned feature field $\mathcal{F} : (\mathbf{x}, \theta) \rightarrow \mathbf{f} \in \mathbb{R}^d$ that maps a 3D position $\mathbf{x}$ and viewing direction θ to a feature vector $\mathbf{f}$, computed by the *view transformer*. Firstly each source view is encoded to a feature map using a U-Net [33] Image Encoder $\mathbf{F}_i = \text{U-Net}(\mathbf{I_i})$. To obtain the feature representation corresponding to a 3D point $\mathbf{x}$, the point is projected to every source image via the camera projections $\Pi_i(\mathbf{x})$, the corresponding feature values are then calculated by interpolating the image-aligned feature map at the projected point. The *view transformer* (VT) is then used to combine all these feature vectors through attention as

$$\mathcal{F}(\mathbf{x}, \theta) = \text{VT}(\mathbf{F_1}(\Pi_1(\mathbf{x}), \theta), \cdots, \mathbf{F_N}(\Pi_N(\mathbf{x}), \theta)). \quad (1)$$

Ray Transformer. The *ray transformer* aggregates information along a given camera ray by performing attention between feature values $\mathbf{f}_i = \mathcal{F}(\mathbf{x}_i, \theta)$, along the ray. The GNT pipeline consists of stacking several view and ray transformer blocks, which iteratively refines the feature field by first aggregating information from source views with the *view transformer* and then accumulating the information along the camera ray with the *ray transformer*. The final *ray transformer* then computes the RGB value $\mathbf{C}(\mathbf{r})$ corresponding to a camera ray $\mathbf{r}$ by feeding the sequence of feature vectors along the ray $\{\mathbf{f}_1, \cdots, \mathbf{f}_M\}$ into the *ray transformer*, performing mean pooling over all predicted tokens, and mapping the pooled feature vector to RGB via an MLP as

$$\mathbf{C}(\mathbf{r}) = \text{MLP} \circ \text{Mean} \circ \text{RT}(\mathbf{f}_1, \ldots, \mathbf{f}_M). \quad (2)$$

This enables training the method using the standard color prediction loss term commonly used by NeRFs. The attention values from the ray transformer correspond to the importance of each feature vector $\mathbf{f}_i$ along the ray to form the image, reflecting point-to-point based occlusion and visibility. It is thus the attention scores that determine the geometry of the rendered scenes, filling a similar role as the opacity in a traditional NeRF method.

3.2. Adjusting Visual Appearance

To change the visual appearance of rendered views to match a target appearance, we propose to introduce a latent appearance variable $\mathbf{z_{c'}}$ as an additional input to the *ray transformer*, with the goal of conditioning the rendering based on the target appearance. The full proposed architecture can be seen in Fig. 2.

The latent variable should correspond to a predefined appearance condition and the value for each condition is

Table 1. Qualitative comparison of rendering quality against 2D-methods (**PSNR↑** | **SSIM↑** | **LPIPS↓**). We observe that our method outperforms all 2D-methods on these metrics, with significant increases in performance on PSNR and SSIM for most scenarios.

Type	Method	Scenarios																	
		Day to Night			Day to Evening			Day to Rain			Night to Day			Evening to Day			Rain to Day		
GANs	Pix2Pix-HD [50]	19.7	0.36	0.565	18.4	0.35	0.603	19.7	0.53	0.582	13.8	0.40	0.629	15.3	0.46	0.619	13.9	0.43	0.629
	BicycleGAN [57]	19.0	0.38	0.556	18.8	0.41	0.587	22.7	0.66	0.578	14.2	0.47	0.630	15.9	0.56	0.627	15.0	0.54	0.630
	NICE-GAN [7]	18.3	0.29	0.553	18.8	0.39	0.589	20.8	0.56	0.583	12.9	0.29	0.626	14.6	0.45	0.618	14.3	0.47	0.624
	CyEDA [2]	17.9	0.32	0.556	18.8	0.40	0.597	20.0	0.67	0.579	11.7	0.47	0.625	14.0	0.59	0.633	12.3	0.53	0.634
	SANet [29]	18.9	0.50	0.571	20.2	0.64	0.606	20.1	0.66	0.581	14.5	0.52	0.629	15.5	0.59	0.618	12.6	0.45	0.616
Diffusion Models	Palette [34]	19.4	0.42	0.577	20.6	0.54	0.618	22.6	0.66	0.601	12.1	0.41	0.689	9.8	0.38	0.688	9.8	0.38	0.695
	DiffuseIT [22]	17.1	0.32	0.594	17.2	0.44	0.627	16.0	0.43	0.613	11.2	0.35	0.626	12.3	0.38	0.618	11.8	0.35	0.625
	Instruct-Pix2Pix [4]	15.9	0.34	0.579	14.3	0.53	0.653	14.1	0.53	0.638	11.5	0.46	0.647	8.7	0.34	0.674	13.4	0.52	0.640
	Ours	**21.0**	**0.56**	**0.549**	**24.1**	**0.75**	**0.585**	**23.4**	**0.71**	**0.577**	**15.3**	**0.56**	**0.624**	**16.7**	**0.64**	**0.615**	**15.7**	**0.59**	**0.602**

jointly optimized with the rest of the network. Since our goal is to change the visual appearance, we include $\mathbf{z}_{\mathbf{c}'}$ so that the geometry is kept unchanged. To ensure this, it is used to update the value-tokens in the *ray transformer* while keeping the attention values unchanged, i.e.,

$$V_{c'} = f_z\left(\begin{bmatrix} V \\ \mathbf{z}_{\mathbf{c}'} \end{bmatrix}\right), \tag{3}$$

where f_z is a single layer MLP that takes in the original value tokens V and generates updated value tokens $V_{c'}$ that have been updated based on the latent appearance variable $\mathbf{z}_{\mathbf{c}'}$. This enables computing a visual appearance change loss term,

$$\mathcal{L}_{appearance} = \left\| \mathbf{C}(r, \mathbf{I}_{s,c}, \mathbf{z}_{c'}) - \hat{\mathbf{C}}(r, \mathbf{I}_{t,c'}) \right\|_2^2. \tag{4}$$

This loss enforces that when inputting source views $\mathbf{I}_{s,c}$ from the condition c together with the latent appearance variable $\mathbf{z}_{c'}$ corresponding to the condition c', then the predicted color $\mathbf{C}(r, \mathbf{I}_{s,c}, \mathbf{z}_{c'})$ should match the ground truth color $\hat{\mathbf{C}}(r, \mathbf{I}_{t,c'})$ for the corresponding target images $\mathbf{I}_{t,c'}$. If the visual appearance of the target image $\mathbf{I}_{t,c}$ corresponds to that of the source images $\mathbf{I}_{s,c}$, then this becomes a traditional reconstruction loss,

$$\mathcal{L}_{rec} = \left\| \mathbf{C}(r, \mathbf{I}_{s,c}, \mathbf{z}_{c}) - \hat{\mathbf{C}}(r, \mathbf{I}_{t,c}) \right\|_2^2. \tag{5}$$

The final loss term is then acquired by combining these two loss terms,

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{appearance}. \tag{6}$$

Rendering images with changed visual appearance is done by computing the rendered color $\mathbf{C}(r, I_{s,c}, \mathbf{z}_{c'})$ for all pixels in an image, giving as input source views from one condition and a latent variable $\mathbf{z}_{c'}$ corresponding to the desired target condition.

4. Experiments

Qualitative and quantitative experiments are performed to test our method's ability to adapt the visual appearance of real and synthetic scenes that have not been seen during training.

Figure 4. Appearance change from the day condition into the three other conditions. We observe that our method is able to take images at one condition and generate new views of that scene at the three other conditions by changing the overall visual appearance of the images to match the desired condition and by making local changes such as turning on street lamps.

Dataset. The used dataset is generated using CARLA [11], an Unreal Engine 4 based open-source simulator for autonomous driving research. CARLA enables the generalization of synthetic images within a simulated city environment along with ground truth camera poses. Additionally, weather and lighting conditions can easily be changed.

For these experiments, four conditions were defined: *night*, *day*, *rain* and *evening*. A scene in the context of this simulated city was defined as a sequence of 10 observations taken along a road, with small pose differences between images. With four different conditions, this led to a total of 40 images per scene. All generated images are 800×600 pixels. The CARLA map was split into two regions, one was used to generate 145 training scenes, and the other region was used to generate 38 evaluation scenes, ensuring separation between training and evaluation scenes. This dataset alongside the code will be made publicly available. We also show qualitative examples using the Spaces dataset [13] to show that our method can generalize to real images.

Figure 5. Gradually changing visual appearance by interpolating between latent appearance variables corresponding to day and night. The first row corresponds to latent variables generated with a given structure, enforcing that the evening condition lies between day and night in the latent space, and the second row corresponds to a learned latent variable with no enforced structure. We observe that the model is able to smoothly interpolate between two conditions, generating plausible intermediate visual appearances. Additional results for interpolation can be seen in the video on the project page.

Table 2. Comparison of similarity of rendered views for our method with ground truth images for all combinations of weather and lighting conditions (**PSNR↑** | **SSIM↑** | **LPIPS↓**). The values along the diagonal correspond to novel view synthesis without appearance change. The off-diagonal values correspond to evaluating novel views with changed visual appearance to match the target condition.

	From Day			From Night			From Evening			From Rain		
Into Day	23.9	0.77	0.60	15.3	0.56	0.62	16.7	0.64	0.61	15.7	0.59	0.60
Into Night	21.0	0.56	0.55	27.4	0.68	0.57	20.7	0.54	0.55	21.2	0.57	0.55
Into Evening	24.1	0.75	0.58	20.0	0.62	0.57	25.4	0.76	0.58	21.4	0.69	0.57
Into Rain	23.4	0.71	0.58	21.7	0.66	0.57	21.3	0.69	0.56	26.8	0.78	0.58

Implementation. The model was initialized with weights from a GNT network pretrained on a combination of synthetic and real data, with parameters as specified in [41]. The model was trained to perform visual appearance change using the 145 training scenes from the introduced CARLA dataset, including the proposed appearance change loss term (4). The training was performed on a single A100 GPU, taking approximately 8 hours, and the method was then able to generalize to scenes not seen during training. Note that the model was trained for all training scenes at the same time, and there is no scene-specific training for the test scenes. When we test the model, we only use images of the test scene in the source condition $\mathbf{I}_{s,c}$, and the method does not have access to any images of the test scene in the target condition $c' \neq c$.

Main Results. Our trained model is evaluated on the 38 evaluation scenes not seen during training. The method is capable of synthesizing novel views of a given scene using only a set of images with corresponding camera poses. Furthermore, it is able to adapt the visual appearance of the scene to specified weather and lighting conditions, without having access to observations of the scene under those target conditions. We show several qualitative examples of this. Figures 3 and 4 show that our method is able to change the visual appearance of images to match a target weather and lighting condition, and Fig. 8 shows a comparison with applying different 2D style transfer methods on rendered views. Only Pix2Pix-HD [50] and Palette [34] learn to turn on the street lamps. CyEDA [2] and SANet [29] achieved better structure preservation, although there is some unknown lighting around some objects or structures. As to diffusion models, both DiffuseIT [22] and Instruct-Pix2Pix [4] can provide visually plausible individual results, but some hallucinations that do not exist in the original images can be observed. Palette can generate realistic images that correspond to the structure of the original images, however, multi-view consistency cannot be fulfilled.

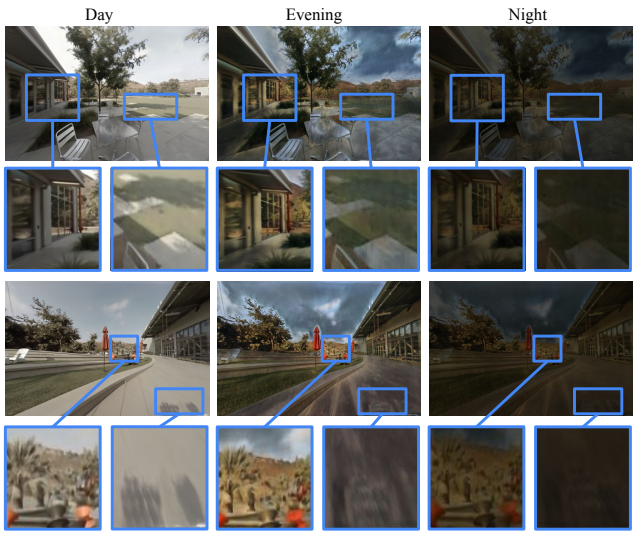

Figure 6. Visual appearance change applied on two daytime scenes from the Spaces dataset [13]. We observe that our method is able to make realistic appearance changes, such as adding sunlight on the background and light reflections in the windows for the evening condition and removing shadows for the night condition, without being trained on any scenes from this dataset.

It also becomes possible to interpolate between two latent variables corresponding to different conditions by defining $\mathbf{z}_\alpha = \alpha \mathbf{z}_\mathbf{c} + (1-\alpha)\mathbf{z}_{\mathbf{c}'}$ for $\alpha \in [0, 1]$. In Fig. 5, we observe that this enables getting realistic intermediate visual appearances that are not included in the original images. The model trained on appearance change of synthetic scenes can also be applied to change appearance of real scenes [13], as seen in Fig. 6 where we can see realistic appearance changes even though the model is not trained on that data.

We now show quantitative rendering quality results based on the three performance metrics Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure

Table 3. Quantitative comparison of the consistency of novel view rendering against 2D-methods (**tOF↓** | **tLP↓** [8]). We can observe that our method significantly outperforms most of the 2D methods. Please see the video on the project page for a visual comparison of the rendering consistency for the different methods.

Type	Method	Scenarios							
		Day to Night		Day to Evening		Day to Rain		Night to Day	
GANs	Pix2Pix-HD [50]	2.59	0.147	1.58	0.169	1.60	0.030	2.49	0.078
	BicycleGAN [57]	5.13	0.053	4.79	0.083	5.10	0.024	5.20	0.047
	NICE-GAN [7]	1.93	0.040	1.24	0.081	1.25	0.014	2.09	0.055
	CyEDA [2]	1.62	0.027	1.21	0.115	**0.96**	0.022	**1.25**	**0.032**
	SANet [29]	2.37	0.069	2.05	0.097	1.73	0.088	2.01	0.092
Diffusion Models	Palette [34]	10.12	0.115	7.21	0.109	8.41	0.057	18.77	0.050
	DiffuseIT [22]	24.62	0.236	29.23	0.242	27.43	0.166	29.75	0.188
	Instruct-Pix2Pix [4]	1.62	0.123	1.37	0.121	1.45	0.088	1.48	0.092
	Ours	**1.44**	**0.026**	**1.10**	**0.035**	0.97	**0.013**	**1.25**	**0.032**

Table 4. Ablation Study comparing two approaches for generating latent appearance variables, comparing the similarity of rendered views with ground truth images (**PSNR↑** | **SSIM↑** | **LPIPS↓**). We observe that both approaches give similar performance for changing appearance from one condition to another.

Latent Variables	Scenarios											
	Day to Night			Day to Evening			Day to Rain			Night to Day		
Enforced structure	21.0	0.56	0.55	24.1	0.75	0.58	23.4	0.71	0.58	15.3	0.56	0.62
No enforced structure	21.6	0.57	0.55	23.2	0.71	0.57	22.8	0.70	0.55	15.3	0.56	0.61

(SSIM) [51] and Learned Perceptual Image Patch Similarity (LPIPS) [55]. The images with changed appearance are evaluated against the corresponding ground truth images for the target weather and lighting conditions. In Table 2, we show how our method performs on all possible combinations of source and target conditions. Using the same source and target conditions corresponds to novel view synthesis without appearance change, which, as expected, gives better metrics, but the gap is small for some combinations, e.g. comparing Day into Evening with Evening into Evening. In Table 1, we compare our method with several 2D style transfer methods. We see that our method outperforms the 2D methods on the performance metrics for all combinations. We observe that performance varies for the different conditions and that adapting images from another condition into day is the most challenging, while transforming from day gives significantly higher performance for all methods.

Additionally, we show two consistency metrics [8] in Table 3. If $(x_1, \ldots, x_n)$ and $(y_1, \ldots, y_n)$ are two image sequences rendered from the same pose sequences, we define $\text{tOF} = \|\text{OF}(y_{t+1}, y_t) - \text{OF}(x_{t+1}, x_t)\|_1$, where OF is the optical flow computed via RAFT [43] and $\text{tLP} = \|\text{LPIPS}(y_{t+1}, y_t) - \text{LPIPS}(x_{t+1}, x_t)\|_1$. The metrics are low if the reference images and the rendered images yield similar optical flow and similar changes in LPIPS, which is assumed to correspond to a consistent rendering. We see that our method significantly outperform most of the 2D style transfer methods regarding consistency. CyEDA gives comparable consistency metrics for some scenarios, but this method gives less realistic rendered views as seen in Fig. 8 and Table 1.

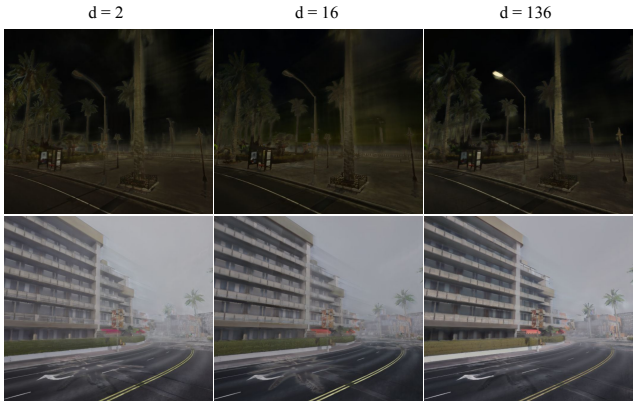

Figure 7. Qualitative comparison of rendered views with changed appearance for different sizes d of the latent appearance variable $\mathbf{z_c}$. We observe that a higher value of d leads to better local appearance changes in rendered views, such as turning on street lamps and removing shadows.

Ablation Study. We compare two different ways of learning latent appearance variables $\mathbf{z_c} \in \mathbb{R}^d$. One approach is to initialize a random d-dimensional vector for each condition and then include it as a learnable parameter that is optimized jointly with the rest of the model. In this case, the latent variables are fully learned with no enforced structure.

Another approach is to enforce structure by representing each condition as a fixed 2D-coordinate, placing them such that the evening condition is in between day and night, based on the assumption that one should pass through evening when going from day to night. These fixed 2D coordinates are then fed through a small learned fully-connected network to generate $\mathbf{z_c}$, a learned latent representation of dimension d for each condition. Comparing

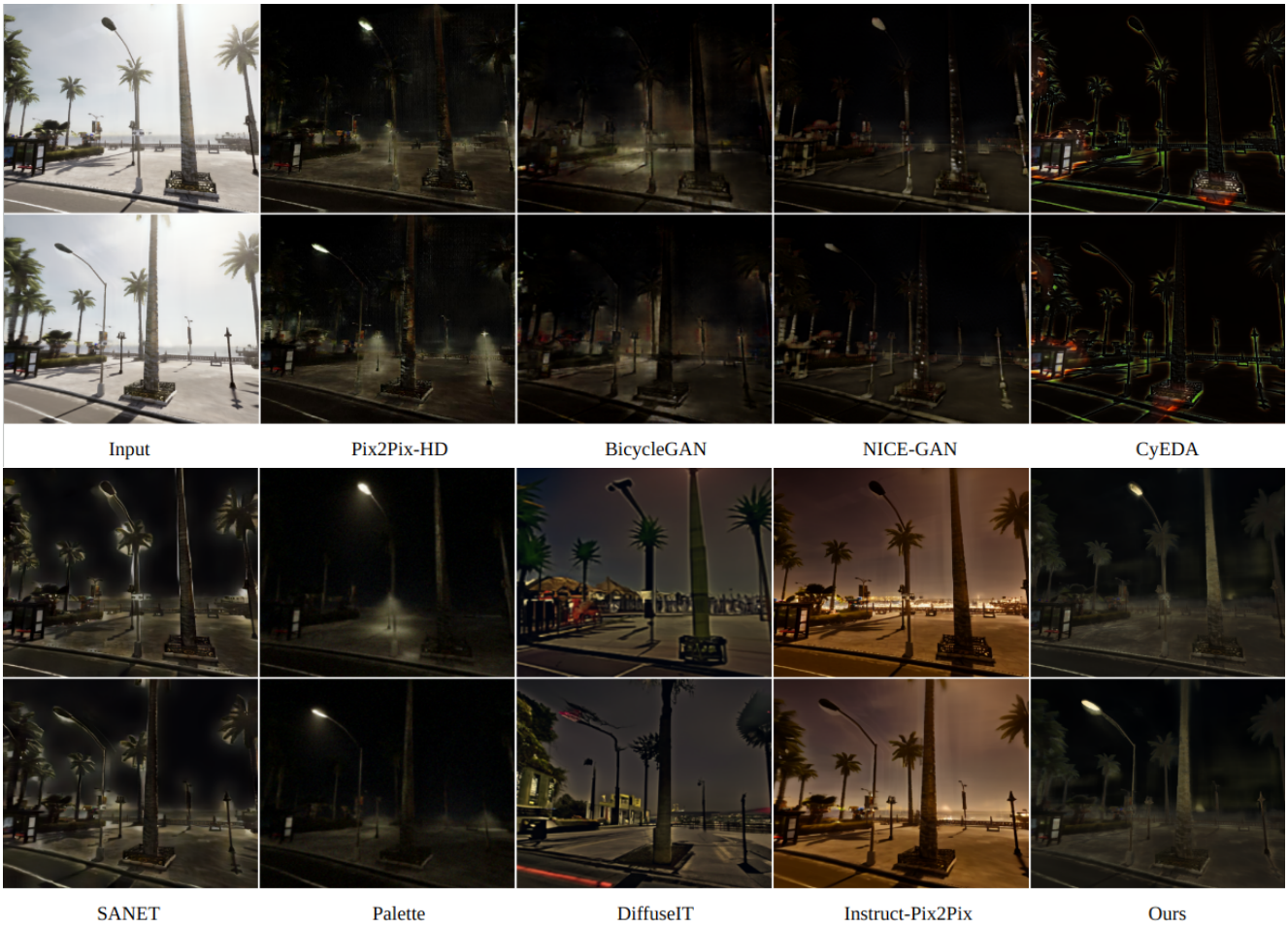

Figure 8. Comparing our method with applying 2D style transfer on rendered images. Our method is able to change the visual appearance while preserving scene content and gives multi-view consistent renderings, as can be seen in Tab. 3 and the video on the project page. Only Pix2Pix-HD [50] and Palette [34] learn to turn on the street lamps. SANet [29] and CyEDA [2] achieve better structure preservation with some noticeable artifacts. The diffusion models DiffuseIT [22] and Instruct-Pix2Pix [4] can provide visually plausible results for individual images, but there are hallucinations that do not exist in the original images. Palette provides more realistic images, but it is however lacking in temporal consistency. Comparisons for additional conditions can be found in appendix A.

the performance metrics for these two approaches, as can be seen in Table 4, shows that both approaches for generating the latent variable $\mathbf{z_c}$ give similar performance when changing appearance from one condition to another. Enforcing structure in the latent space results in more realistic lighting effects during interpolation, as can be seen in Fig. 5, giving the appearance of a sunset. Based on this, it was decided to use the latent appearance variable with the enforced structure for our experiments.

Choosing $d = 136$ for the latent appearance variable resulted from testing various values and finding that higher dimensions better handle local appearance changes, e.g., street lamp activation and shadow removal, shown in Fig. 7. More details regarding the generation of latent appearance variables can be found in appendix B.

5. Conclusions

We present a transformer based generalizable novel view synthesis method that allows for change of visual appearance without any scene-specific training. This is achieved by introducing a latent appearance variable that is used to change the visual appearance to match a given weather and lighting condition while keeping the structure of the scene unchanged. We present experiments that show that this method is able to change the visual appearance of both synthetic and real scenes that have not been seen during training, to match a specified weather and lighting condition. The generated latent variables also make it possible to smoothly interpolate between different weather and lighting conditions. Compared to 2D style transfer, our method

is view consistent by design. We experimentally show that our method outperforms multiple 2D style transfer methods, both in terms of rendering quality as well as by showing that the rendering consistency of nearby views are more consistent. We also introduce a synthetic dataset based on CARLA for training and evaluating the methods.

Societal Impact. A potential positive impact of our work is that moving towards using generalizable methods that do not require scene-specific training leads to less requirement of large computational resources. The goal of our research is to generate photorealistic views of a scene with changed visual appearance. A potential negative impact of this could be to spread misinformation, generating fake images or videos that could be presented as being real.

Acknowledgments. This work was fully supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

[1] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. *ICCV*, 2021. 1

[2] Jing Chong Beh, Kam Woh Ng, Jie Long Kew, Che-Tsung Lin, Chee Seng Chan, Shang-Hong Lai, and Christopher Zach. Cyeda: Cycle-object edge consistency domain adaptation. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 2986–2990. IEEE, 2022. 2, 5, 6, 7, 8

[3] Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. *ICML*, 2018. 3

[4] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 2, 3, 5, 6, 7, 8, 15

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[6] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14124–14133, October 2021. 2

[7] Runfa Chen, Wenbing Huang, Binghui Huang, Fuchun Sun, and Bin Fang. Reusing discriminators for encoding: Towards unsupervised image-to-image translation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8168–8177, 2020. 2, 5, 7

[8] Mengyu Chu, You Xie, Jonas Mayer, Laura Leal-Taixé, and Nils Thuerey. Learning temporal coherence via self-supervision for gan-based video generation. *ACM Transactions on Graphics (TOG)*, 39(4):75–1, 2020. 7

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 3

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*, 2022. 2

[11] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017. 4, 5

[12] Yilun Du, Cameron Smith, Ayush Tewari, and Vincent Sitzmann. Learning To Render Novel Views From Wide-Baseline Stereo Pairs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4970–4980, 2023. 2

[13] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5, 6

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[15] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. StyleneRF: A style-based 3d aware generator for high-resolution image synthesis. In *International Conference on Learning Representations*, 2022. 3

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 3

[17] 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 (ICCV)*, Oct 2017. 3

[18] Yi-Hua Huang, Yue He, Yu-Jie Yuan, Yu-Kun Lai, and Lin Gao. Stylizednerf: Consistent 3d scene stylization as stylized nerf via 2d-3d mutual learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[19] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1125–1134, 2017. 2

[20] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5885–5894, October 2021. 2

[21] Andreas Kurz, Thomas Neff, Zhaoyang Lv, Michael Zollhöfer, and Markus Steinberger. Adanerf: Adaptive sampling for real-time rendering of neural radiance fields. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[22] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022. 2, 3, 5, 6, 7, 8, 13, 14

[23] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Theobalt Christian, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *CVPR*, 2022. 2

[24] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7210–7219, June 2021. 1, 3

[25] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1, 2

[26] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. 2

[27] Michael Niemeyer, Jonathan T. Barron, Ben Mildenhall, Mehdi S. M. Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5480–5490, June 2022. 2

[28] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[29] Dae Young Park and Kwang Hee Lee. Arbitrary style transfer with style-attentional networks. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5880–5888, 2019. 3, 5, 6, 7, 8

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021. 3

[31] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10901–10911, October 2021. 2

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 3

[33] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 4

[34] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 2, 3, 5, 6, 7, 8

[35] Mehdi S.M. Sajjadi, Henning Meyer, Etienne Pot, Urs Bergmann, Klaus Greff, Noha Radwan, Suhani Vora, Mario Lucic, Daniel Duckworth, Alexey Dosovitskiy, Jakob Uszkoreit, Thomas Funkhouser, and Andrea Tagliasacchi. Scene Representation Transformer: Geometry-Free Novel View Synthesis Through Set-Latent Scene Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6219–6228, 2022. 2

[36] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 3

[37] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 3

[38] Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *CVPR*, 2021. 1

[39] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Generalizable Patch-Based Neural Rendering. In *European Conference on Computer Vision (ECCV)*, pages 156–174, 2022. 2

[40] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Light Field Neural Rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8259–8269, 2022. 2

[41] Mukund Varma T, Peihao Wang, Xuxi Chen, Tianlong Chen, Subhashini Venugopalan, and Zhangyang Wang. Is attention all that neRF needs? In *The Eleventh International Conference on Learning Representations*, 2023. 2, 4, 6

[42] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P. Srinivasan, Jonathan T. Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8248–8258, June 2022. 1, 3

[43] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 7

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Neural Information Processing Systems (NIPS*, pages 6000–6010, 2017. 2

[45] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3835–3844, June 2022. 3

[46] Can Wang, Ruixiang Jiang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Nerf-art: Text-driven neural radiance fields stylization. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–15, 2023. 3

[47] Dan Wang, Xinrui Cui, Septimiu Salcudean, and Z. Jane Wang. Generalizable neural radiance fields for novel view synthesis with transformer, 2022. 2

[48] Huan Wang, Jian Ren, Zeng Huang, Kyle Olszewski, Menglei Chai, Yun Fu, and Sergey Tulyakov. R2l: Distilling neural radiance field to neural light field for efficient novel view synthesis. In *ECCV*, 2022. 2

[49] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P. Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699, June 2021. 2

[50] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8798–8807, 2018. 2, 5, 6, 7, 8, 13

[51] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 7

[52] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4578–4587, June 2021. 2

[53] Kai Zhang, Nick Kolkin, Sai Bi, Fujun Luan, Zexiang Xu, Eli Shechtman, and Noah Snavely. Arf: Artistic radiance fields, 2022. 3

[54] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 1

[55] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 7

[56] 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/CVF International Conference on Computer Vision (ICCV)*, pages 2223–2232, 2017. 2

[57] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. *Advances in neural information processing systems*, 30, 2017. 2, 5, 7

Table 5. Comparison of similarity of rendered views when using learnable latent variables with no enforced structure. Comparing with ground truth images for all combinations of weather and lighting conditions (**PSNR**↑ | **SSIM**↑ | **LPIPS**↓). The values along the diagonal correspond to novel view synthesis without appearance change.

	From Day			From Night			From Evening			From Rain		
Into Day	23.3	0.76	0.61	15.3	0.56	0.61	16.4	0.61	0.60	15.9	0.61	0.61
Into Night	21.6	0.57	0.55	28.1	0.73	0.55	21.1	0.56	0.54	21.4	0.58	0.55
Into Evening	23.2	0.71	0.57	19.9	0.47	0.58	23.1	0.55	0.57	20.3	0.50	0.57
Into Rain	22.8	0.70	0.55	20.9	0.66	0.56	21.3	0.69	0.57	23.1	0.55	0.58

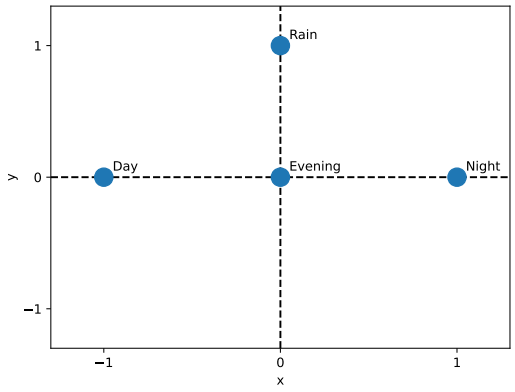

Figure 9. Chosen fixed 2D coordinates for each condition, ensuring that one passes through the evening condition when interpolating between the day and night conditions. Rain is placed on a separate axis since it corresponds to appearance change not directly connected to variations in daylight.

Appendix

In appendix A we show comparisons with 2D style transfer methods for additional scenarios. In appendix B we give more detail regarding the generation of latent appearance variables.

A. Comparisons for additional scenarios

We further compare our method with applying 2D style transfer on rendered images for additional scenarios, including Day to Evening (Fig. 10), Day to Rain (Fig. 11), and the most challenging scenario Night to Day (Fig. 12). Our method is able to clearly preserve content and 3D consistency while making appropriate adjustments to the appearance of the scene. For instance, we effectively eliminate shadows from images for the Day to Evening scenario. In the Day to Rain scenario, our model maintains scene content while altering visual appearance, ensuring multi-view consistency. Lastly, our model can even learn to deactivate interior lighting in buildings in the most challenging Night to Day scenario.

B. Generating Latent Variables

We propose two approaches for generating latent appearance variables $\mathbf{z_c} \in \mathbb{R}^d$. One approach is to initialize a random d-dimensional vector for each condition and then include it as a learnable parameter that is optimized jointly with the rest of the model. In this case, the latent variables are fully learned with no enforced structure.

Another approach is to enforce structure on the latent appearance variables by defining fixed 2D-coordinates c corresponding to each condition that is then passed through a small fully-connected network to generate $\mathbf{z_c} = f_z(c)$, where the parameters of this additional fully-connected network are learned jointly with the rest of the model. For our case with four weather and lighting conditions, we define the fixed 2D coordinates as shown in Fig. 9.

The reason behind this placement is to get the desired behavior when interpolating between two conditions, ensuring that the evening condition is passed through when interpolating between day and night conditions, and places rain on a separate axis since it corresponds to appearance change not directly connected to variations in daylight. The fully connected network $f_z(c)$ takes in a 2D coordinate corresponding to a condition and outputs a latent appearance variable $\mathbf{z_c}$ of dimension d. For the performed experiments, we used $d = 136$, and two hidden layers of size 16 and 68, respectively.

Comparing performance metrics for learnable latent variables with no enforced structure in Table 5 with the ones in Table 2 where latent appearance variables with enforced structure are used shows that both approaches for generating the latent variable $\mathbf{z_c}$ give similar performance when changing appearance from one condition to another. However, enforcing a structure on the latent space leads to more realistic lighting effects when interpolating between two conditions, as can be seen in Fig. 5, giving the appearance of a sunset. Based on this, it was decided to use the latent appearance variable with the enforced structure for our experiments.

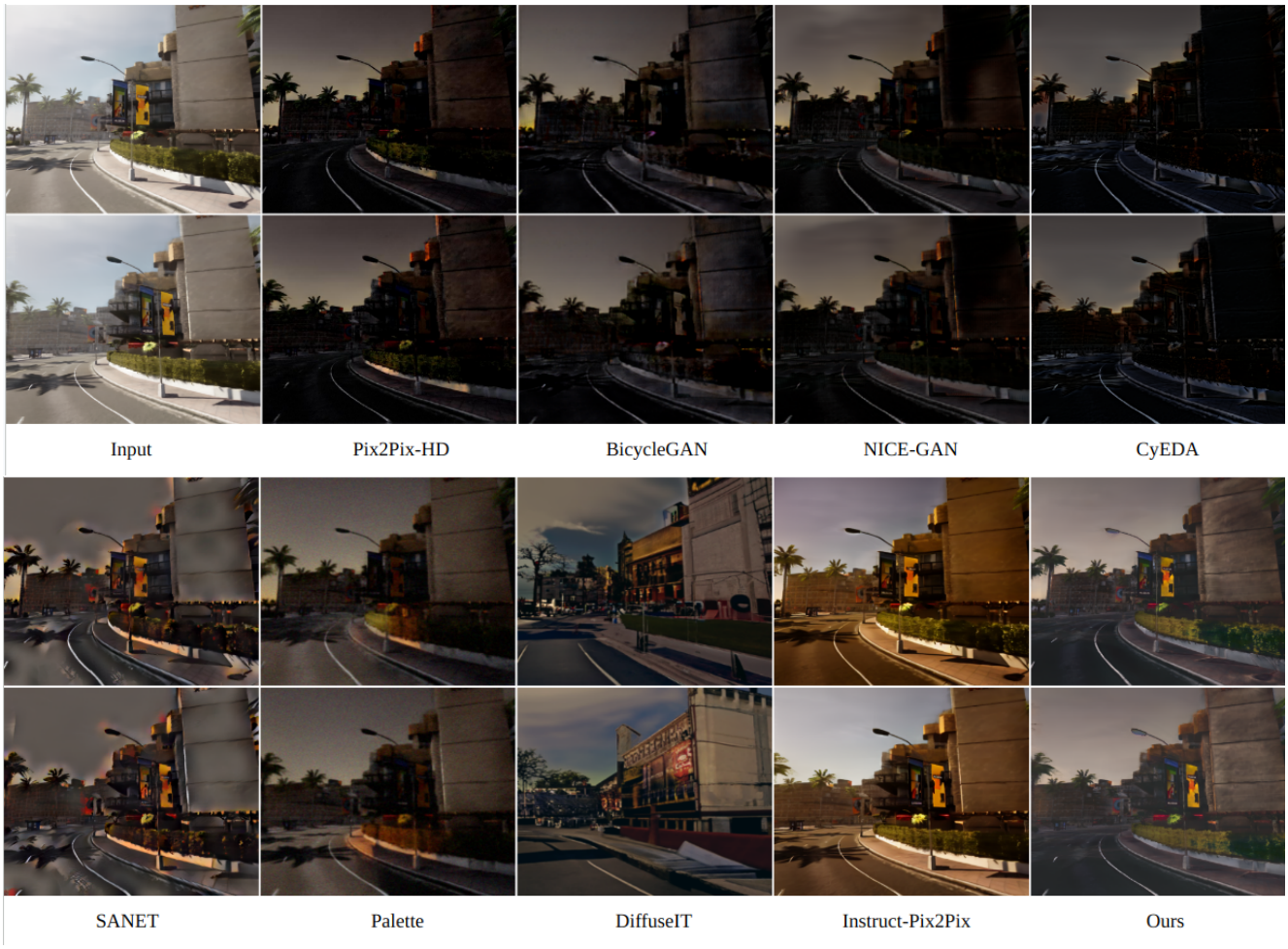

Figure 10. Comparing our method with applying 2D style transfer on rendered images for the Day to Evening scenario. Our method is able to change the visual appearance appropriately, in part by removing shadows. Other than our method, the shadows of trees and street lamps in the images generated by most of the 2D methods are still clearly noticeable, but they should be less visible in the evening when there is no direct sunlight. Pix2Pix-HD [50] seems to be close to our result, but our model can achieve the best quantitative results as seen in Table 1. DiffuseIT [22] can provide visually plausible results for individual images, but there are hallucinations that do not exist in the original images.

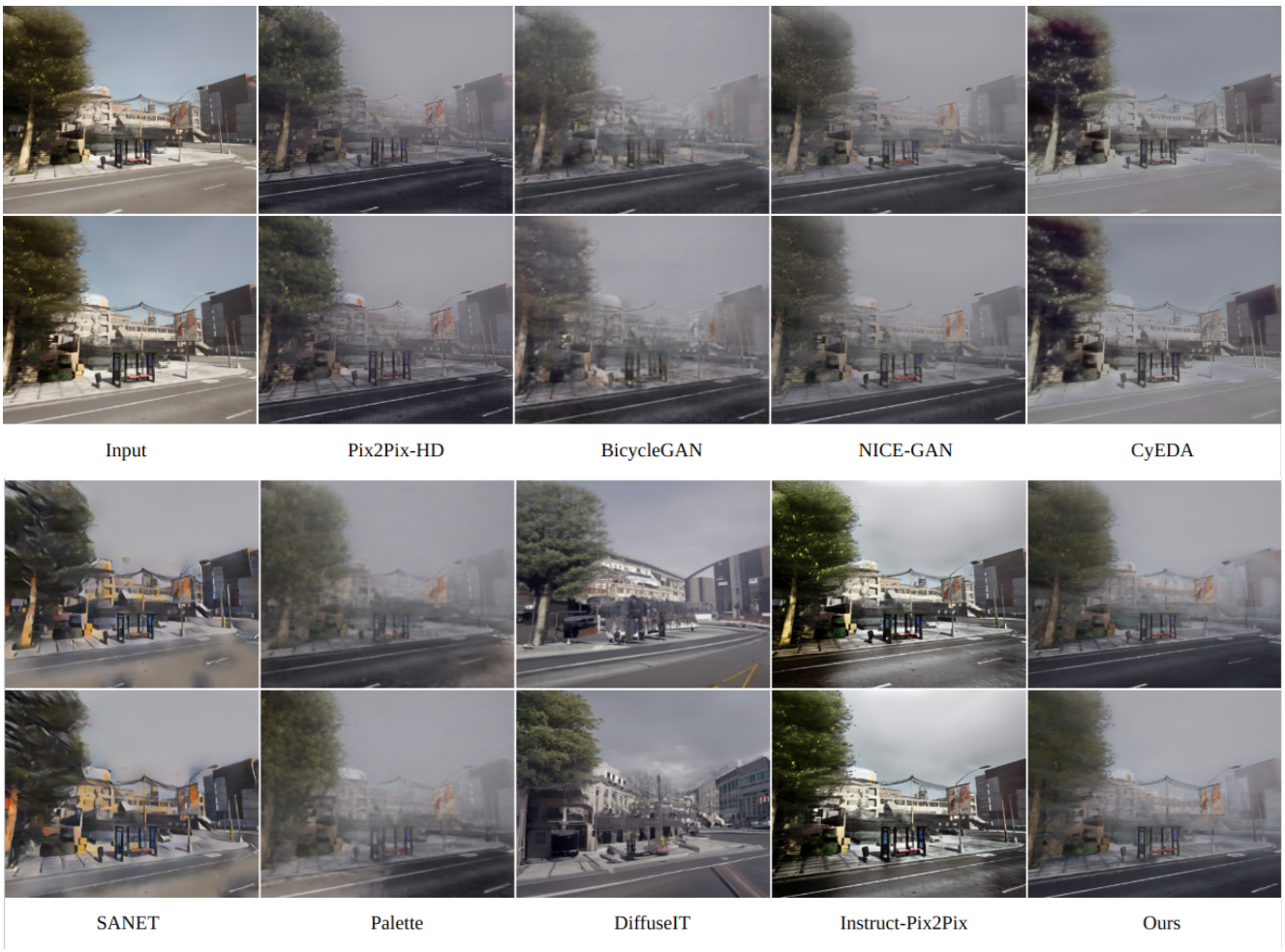

Figure 11. Comparing our method with applying 2D style transfer on rendered images for the Day to Rain scenario. Our method is able to change the visual appearance appropriately while clearly preserving scene content. In this scenario, all the 2D methods give more realistic renderings, except DiffuseIT [22], which hallucinates content that does not exist in the original images.

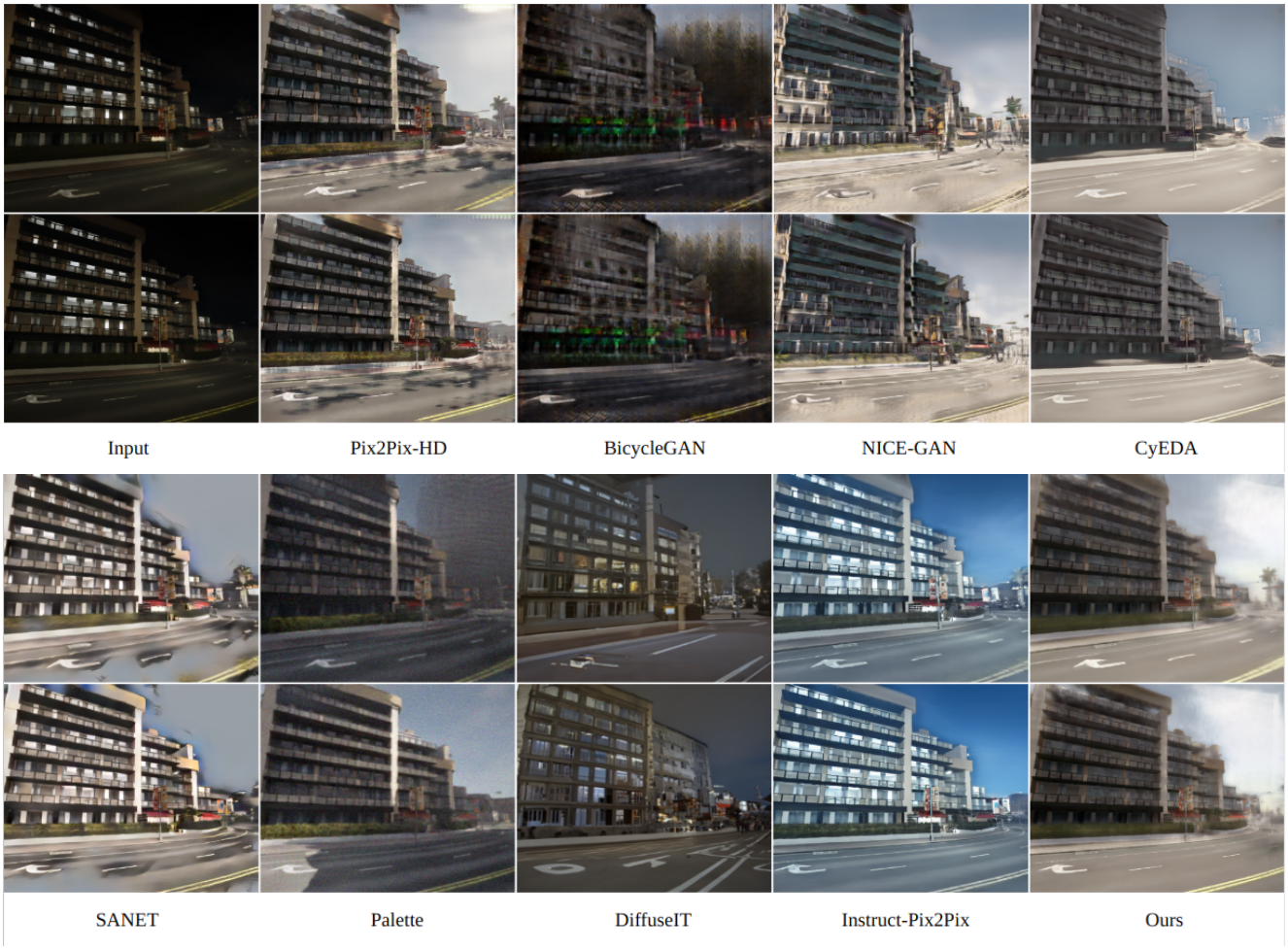

Figure 12. Comparing our method with applying 2D style transfer on rendered images for the difficult Night to Day scenario. Our method is able to change the visual appearance appropriately, in part by deactivating interior lighting inside the buildings. Most of the 2D methods struggle to provide structure-consistent results, and for some, the sky is not even bright. Instruct-Pix2Pix [4] achieves plausible results at first sight, but the interior lighting is still shining clearly, which should not be the case at day time.