TimeNeRF: Building Generalizable Neural Radiance Fields across Time from Few-Shot Input Views

Anonymous Authors

1 Additional Implementation Details

1.1 Architecture

1.1.1 The modifications of DRIT++

In Stage 1, we utilize DRIT++ [6] for feature disentanglement. As described in the main paper, we extract content features at three different levels from different convolutional layers of the content extractor, in contrast to the original DRIT++, which relies only on the final layer output of the content extractor. These multi-level content features capture different semantic information. We then merge each level of content features with the style feature to generate stylized images, as illustrated in Fig. 1. Utilizing features from multiple layers in the generation process encourages the model to leverage more content features and enhances the model's capabilities for stylized image generation.

1.1.2 Details of the implicit scene network

The specific implementation details of $H(\cdot)$, mentioned in Sec. 3.4 of the main paper, are adapted from GeoNeRF [4]. Following GeoNeRF, we employ Multi-Head Self-Attention (MHSA) [9] and full-connected layers to aggregate information of different input views. Below, we explain the procedure:

First, to facilitate the exchange of information between different views, features are aggregated via MHSA layers [9] by the following equations.

$$\tilde{\sigma}^{\mathbf{x}}, \{\tilde{w}_i^{\mathbf{x}}\}_{i=1}^{N} = \mathrm{MHSA}(h(cf^{\mathbf{x}}), \{g_i^{\mathbf{x}}\}_{i=1}^{N}). \tag{1}$$

$$h(z) = \mathrm{FC}(\mathrm{mean}(z)||\mathrm{var}(z)). \tag{2}$$

Here, $cf^{\mathbf{x}}$ represents $\{cf_i^{\mathbf{x}}\}_{i=1}^{N}$, the content features of N input views. $g_i^{\mathbf{x}}$ is the interpolation of the geometry feature at $\mathbf{x}$ from the input view i. $||$ is the concatenation and FC refers to fully-connected layers. MHSA layers combine features from different views, producing enhanced features $\tilde{\sigma}$ and $\{\tilde{w}_i\}_{i=1}^{N}$.

Then, an auto-encoder (AE) network is applied to the features $\{\tilde{\sigma}_p\}_{p=1}^{M}$ of all the sample points $\{\mathbf{x}_p\}_{p=1}^{M}$ along the ray $\mathbf{r}$ for facilitating information exchange along the ray and achieve the updated density features $\{\sigma'_p\}_{p=1}^{M}$ of the M points by

$$\{\sigma'_p\}_{p=1}^{M} = \mathrm{AE}\left(\{\tilde{\sigma}_p\}_{p=1}^{M}\right), \tag{3}$$

where $\tilde{\sigma}_p$ is obtained from eq. (1) for the sample point $\mathbf{x}_p$ in the ray $\mathbf{r}$. After enhancing the geometry information, an MLP is used to predict the final density of each 3D sample point by

$$\sigma = \mathrm{MLP}_{\sigma}(\sigma'). \tag{4}$$

1.1.3 Details of the time-dependent radiance field constructor

As discussed in Sec. 3.5 of the main paper, our time-dependent radiance field constructor, T, uses a two-branch network to convert the content radiance field into the time-dependent radiance field. The architecture design, as shown in Fig. 2 including branches T_1 and T_2 along with an MLP for color decoding, ensures our constructor adequately models the temporal variations. Specifically, the first branch T_1 combines 3D content feature ξ with time t, to serve as the template for the change over time. The second branch T_2 integrates 3D content feature ξ with both time t and time-irrelevant feature a to further tune the color according to time-irrelevant features a. Finally, the output features of T_1 and T_2 are then summed and passed through an MLP to generate the time-dependent color $\mathbf{c}_t$.

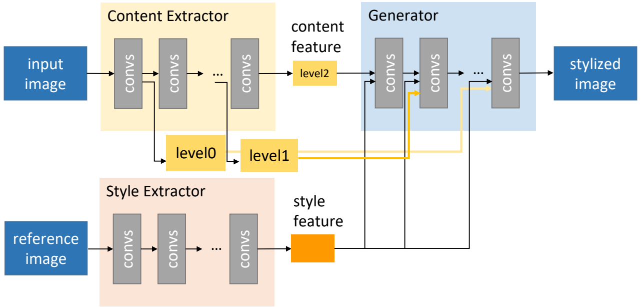


Figure 1: DRIT++ modification. To improve the content extractor in DRIT++, we extract content features at three levels from the convolutional layers and merge each of them with the style feature in the generator.

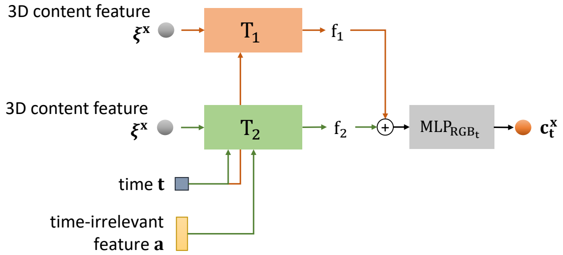


Figure 2: Time-Dependent Radiance Field Constructor. We have developed a two-branch network for the Time-Dependent Radiance Field Constructor to adequately model the temporal variations.

The specifics of T_1 and T_2 are shown in Fig. 3. We use content features at three levels $\{\xi^{(l)}\}_{l=0}^{2}$ and integrate them from the high-level features (i.e, capturing more global information) to the low-level features (i.e., focusing on local details) along with time-related style information.

1.2 Training details

1.2.1 Stage 1 training

To manipulate the time, one possible way is to disentangle time-related information from images in Stage 1. However, doing so

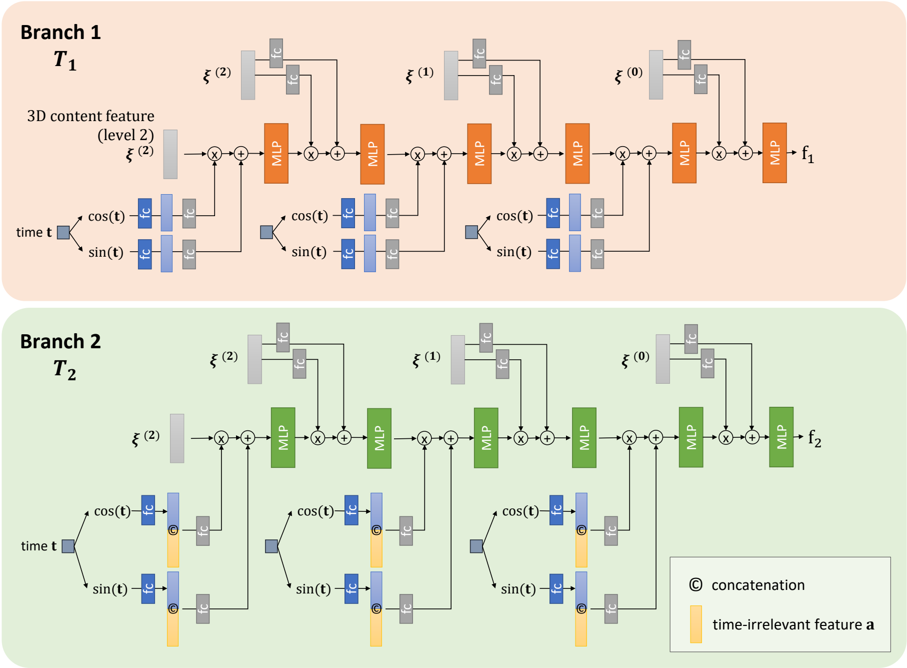


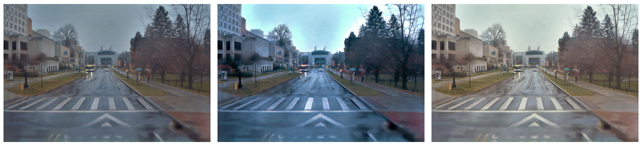
Figure 3: Details of T_1 and T_2. The figure illustrates the architecture design of T_1 and T_2 used in the time-dependent radiance field constructor (Fig. 2).

without supervision across diverse weather conditions is challenging. Weather factors such as lighting and shadows can interfere with accurate time information extraction. While training the model on images from a single weather condition might simplify the task, it risks overfitting to that specific weather condition. For example, CoMoGAN [8], which has the capability of continuous time translation, is trained on a dataset of images captured at different times but only on sunny days. As shown in Fig. 4, when an input image captured on a rainy day is transferred to daytime, it produces a color-biased sky, which is unreasonable.

To avoid this, we first extract style features from images in Stage 1 that encompass all environmental change factors, including both time and weather (time-irrelevant) information. In Stage 2, we then disentangle these factors. By training on a diverse set of weather conditions, our training approach prevents the model from overfitting to a specific weather condition and makes it easier to separate time and weather information from the style feature.

1.2.2 *The inputs of the factor extraction module*

As mentioned in the main paper, we hypothesize that the extracted style encompasses both time and weather information and propose to disentangle these two factors utilizing the factor extraction module. As detailed in Sec. 3.5 of the main paper, we employ two Multi-Layer Perceptrons (MLPs) to predict time t and extract time-irrelevant information (e.g. weather) denoted as a. In the following, we explain our reasoning for extracting t from a reference image and a from an input image during training.



Input image (a) CoMoGAN (b) TimeNeRF

Figure 4: CoMoGAN and TimeNeRF under a rainy day. (a) is the stylized result from CoMoGAN, which converts the rainy scene to the daytime condition. However, it generates a color-biased scene (i.e., the sky). (b) In contrast, TimeNeRF is able to translate images according to different weather conditions without a color bias. More examples can be found in Fig. 6.

The objective of our model is to learn the time transitions occurring throughout the day while preserving the weather conditions present in the input data. To this end, we extract the time-irrelevant feature a from the input view to preserve the weather conditions

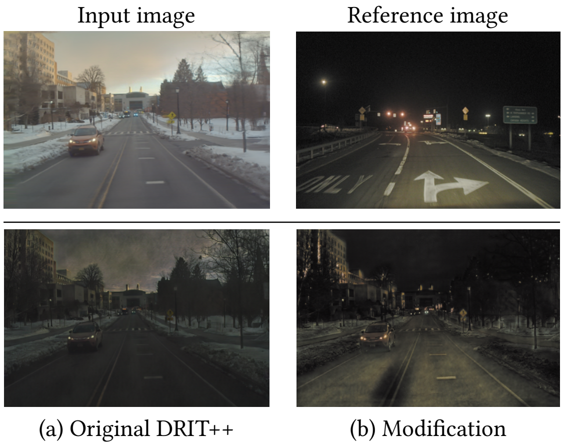


Figure 5: Ablation study on DRIT++ modification. According to the reference image, we transformed the input image into a stylized night-time image using (a) the original DRIT++ and (b) our modified model.

and remove the time-relevant part from the style feature. For learning time transitions, we utilize reference images captured at various times to provide the model with additional time information during training. Specifically, we extract time-related factors t from these reference images and learn to map the style features of reference images into $[0, 2\pi)$. In the testing phase, our model is capable of simulating time transitions by specifying the time t directly without referring to any image, making our method different from the reference-based style transfer methods like DRIT++ [6] and HiDT [1].

2 Additional Experimental Results

2.1 Ablation study

DRIT++ modification. As shown in Fig. 5(a), some residual light remains in the sky within the style-transferred image generated by the original DRIT++ [6]. This is likely because the content feature in DRIT++ still contains some daytime information. In contrast, our modified DRIT++, considering features in three levels, produces a more accurate transferred result (Fig. 5(b)).

2.2 Under diverse weather conditions

Besides the results shown in Fig. 3 of the main paper, to evaluate our model's performance of novel view synthesis across times under varied conditions, we test TimeNeRF with views captured in diverse weather scenarios (i.e., rainy, snowy, and cloudy days). The synthesized results with texture consistency and temporal smoothness shown in Fig. 6 demonstrate the robustness of TimeNeRF to diverse weather conditions.

2.3 Analysis of Possible Alternative Approaches and Their Weaknesses

Alternative approaches that aim to achieve a similar system goal as the proposed TimeNeRF can be considered. However, these alternatives have certain weaknesses that make them unsuitable solutions. Below, we detail these issues.

Novel view synthesis, then style transfer: Synthesizing novel views and subsequently transferring styles can lead to view/geometry inconsistency issues, as discussed in Section 4.4 of the main paper. This issue arises because the style transfer model lacks awareness of 3D geometry, which results in the introduction of unrealistic effects in the scene without considering its underlying 3D structure. Consequently, these methods may struggle with artifacts due to their reliance on 2D information.

Style transfer, then novel view synthesis: On the other hand, applying style transfer to input images before reconstructing a 3D scene and synthesizing novel views may prove ineffective. Given that style transfer operates in 2D space, it often merges inconsistent styles captured from different viewpoints into the 3D scenes. This inconsistency can lead to inaccuracies in NeRF geometry, resulting in imprecise synthesized novel views. Some examples of these inaccuracies are illustrated in Fig. 10.

2.4 Additional Qualitative and Quantitative Studies

2.4.1 *Qualitative results.*

In Fig. 7, we showcase additional synthesized novel views for the Family, Horse, Playground, and Train scenes from the TT dataset [5]. These results demonstrate our model's ability to smoothly generate novel views over time in diverse scenarios. Additionally, we present more qualitative results from the LLFF dataset [7] and the Ithaca365 dataset [3] in Fig. 8. This provides a comparative analysis against state-of-the-art (SOTA) NeRF-based methods focused purely on novel view synthesis without temporal variations. Compared to these SOTA methods, TimeNeRF is able to produce clearer details.

2.4.2 *Quantitative results.*

To further assess the quality of synthesized images across various times, we conduct two additional analyses and comparisons. First, we evaluate color consistency across different periods by calculating the mean histogram correlation for the Y, Cb, and Cr color channels over time. This analysis helps determine how well color properties are maintained throughout different phases of the day. Second, we measure the style consistency between the synthesized images and reference images to measure the overall style coherence of our image synthesis. Detailed explanations of both methodologies are provided below.

Color Consistency Analysis Using YCbCr Color Space. To demonstrate that our method provides better color consistency over time and produces fewer color biases, we analyze the mean histogram correlations in the Y, Cb, and Cr channels over time, as shown in Fig. 9. Here, we measure the correlation between the color histogram at a specific time (daytime) and those at other times. Ideally, the Y channel should show a decrease in correlation as the time difference increases, reflecting changes in illumination. In contrast, the correlations in the Cb and Cr channels should remain

Method	Ithaca365 [3]		T&T [5]	
	Cb	Cr	Cb	Cr
DRIT++[6]	0.409	0.436	0.392	0.461
HiDT[1]	0.315	0.304	0.302	0.294
CoMoGAN[8]	0.398	0.544	0.428	0.612
Ours	**0.702**	**0.722**	**0.614**	**0.769**

Table 1: Analyzing the Mean Histogram Correlation of the Cb and Cr Color Channels Over Time. This table presents the calculated mean histogram correlation of the Cb and Cr channels, which reflects color consistency over different periods. Color consistency should be maintained across varying time intervals. Therefore, higher mean correlation values for the Cb and Cr channels are anticipated

Method	Ithaca[3]		T&T [5]	
	CoMoGAN	Ours	CoMoGAN	Ours
Day	9.2933	**8.9319**	9.0664	**8.8019**
Dusk/Dawn	9.5158	**9.3940**	9.1345	**8.7307**
Night	10.4778	**10.3871**	10.6849	**10.4031**

Table 2: Style Consistency Analysis. In this table, we calculate Fréchet Inception Distance (FID) between the style features of the synthesized image set and the reference set to evaluate the level of style consistency across different time periods. Smaller distance values indicate that the corresponding style is more similar to the style of the reference image at the time period. Our TimeNeRF demonstrates better style consistency compared to CoMoGAN [8].

more consistent. Our results indicate that all methods produce a "U" shaped curve in the Y channel, which mirrors the real-world relationship between illumination and time difference. However, our method demonstrates greater consistency in the Cb and Cr channels. We quantify this by calculating mean correlations, with results detailed in table 1. Notably, our method achieves higher mean correlations in both the Cb and Cr channels compared to previous methods. This suggests that the lower mean correlations observed in these state-of-the-art methods may result from artifacts in the generated images.

Style Consistency Analysis. In this section, we evaluate the quality of image synthesis by measuring the style consistency between the synthesized images and the reference images. The style extractor module from DRIT++ [6] is employed to extract style features from an image. As outlined in section 4.3 of the main paper, we synthesize novel views at three different times of the day: day, dusk/dawn, and night. For each time period, novel images are generated using TimeNeRF, and their style features are subsequently extracted. We then measure the Fréchet Inception Distance (FID) between the style features of the synthesized image set and the reference set to evaluate the level of style consistency across different time periods. The results are presented in table 2. We assess our method using the Itheca365 [3] and TT datasets [5]. Our proposed TimeNeRF method demonstrates better style consistency compared to the state-of-the-art CoMoGAN [8], showcasing its efficacy in synthesizing visually consistent images over time.

3 The implementation code

We are prepared to share the source code for implementation. However, due to limitations on the submission file size exceeding the model's weight, we are unable to provide the pretrained model at this time. If there are any requests regarding pretrained models for cross-checking purposes, please feel free to reach out to us

References

[1] Ivan Anokhin, Pavel Solovev, Denis Korzhenkov, Alexey Kharlamov, Taras Khakhulin, Alexey Silvestrov, Sergey Nikolenko, Victor Lempitsky, and Gleb Sterkin. 2020. High-Resolution Daytime Translation Without Domain Labels. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

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

[3] Carlos A. Diaz-Ruiz, Youya Xia, Yurong You, Jose Nino, Junan Chen, Josephine Monica, Xiangyu Chen, Katie Luo, Yan Wang, Marc Emond, Wei-Lun Chao, Bharath Hariharan, Kilian Q. Weinberger, and Mark Campbell. 2022. Ithaca365: Dataset and Driving Perception Under Repeated and Challenging Weather Conditions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 21383–21392.

[4] M. Johari, Y. Lepoittevin, and F. Fleuret. 2022. GeoNeRF: Generalizing NeRF with Geometry Priors. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR).*

[5] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. 2017. Tanks and Temples: Benchmarking Large-Scale Scene Reconstruction. *ACM Transactions on Graphics* 36, 4 (2017).

[6] Hsin-Ying Lee, Hung-Yu Tseng, Qi Mao, Jia-Bin Huang, Yu-Ding Lu, Maneesh Kumar Singh, and Ming-Hsuan Yang. 2020. DRIT++: Diverse Image-to-Image Translation via Disentangled Representations. *International Journal of Computer Vision* (2020), 1–16.

[7] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. 2019. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Trans. Graph.* 38, 4, Article 29 (jul 2019), 14 pages. https://doi.org/10.1145/3306346.3322980

[8] Fabio Pizzati, Pietro Cerri, and Raoul de Charette. 2021. CoMoGAN: continuous model-guided image-to-image translation. In *CVPR.*

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17).* Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

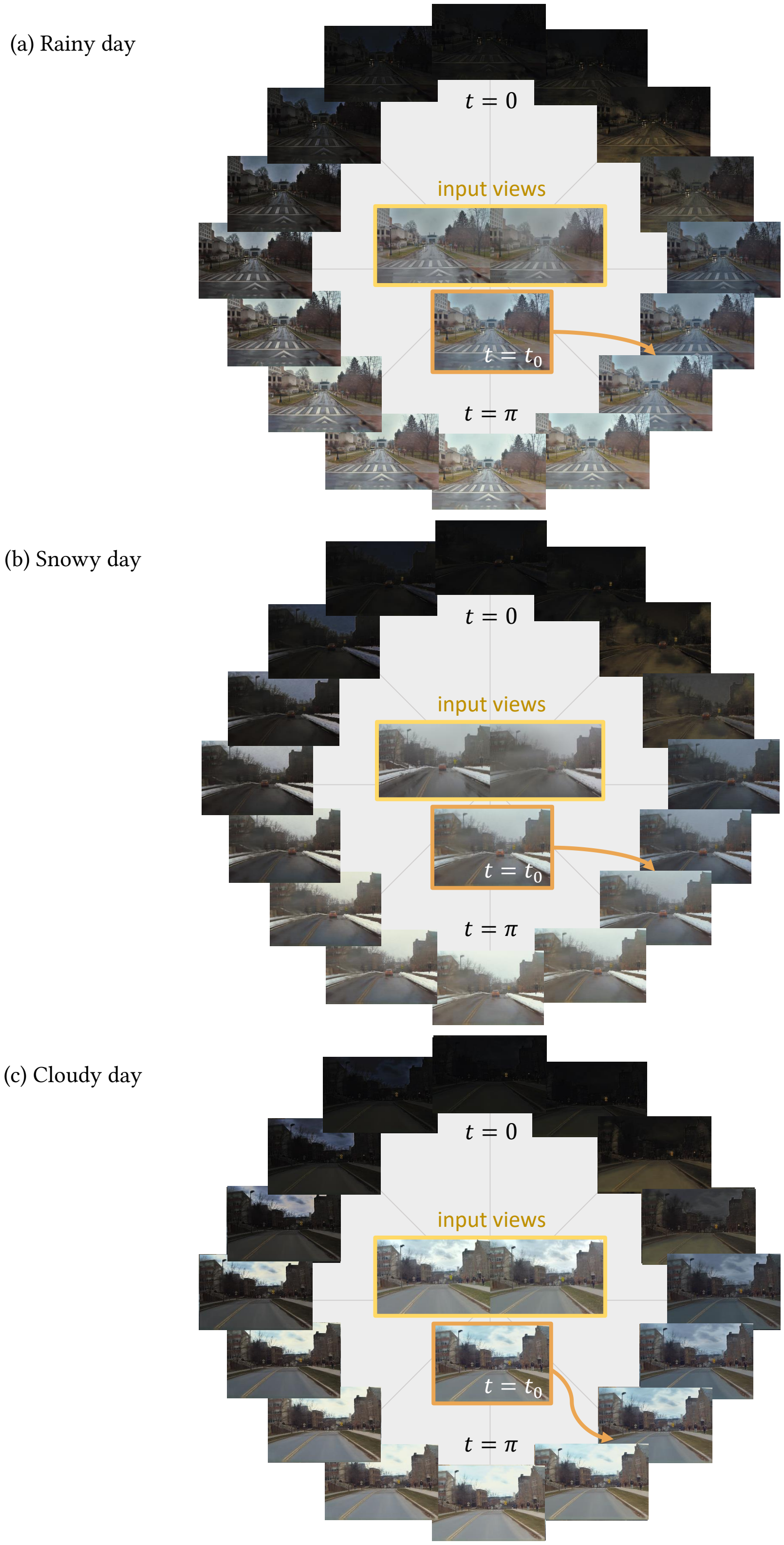


Figure 6: Synthesis under diverse weathers. We show the synthesis results at 16 different time points. The input images are captured on a (a) rainy, (b) snowy, and (c) cloudy day. For each scene, two input images are utilized for 3D reconstruction. The images in the yellow box represent the two input views of a test scene. The images around the circle are novel views at different times. The image in the orange box is synthesized for the time of input views t_0.

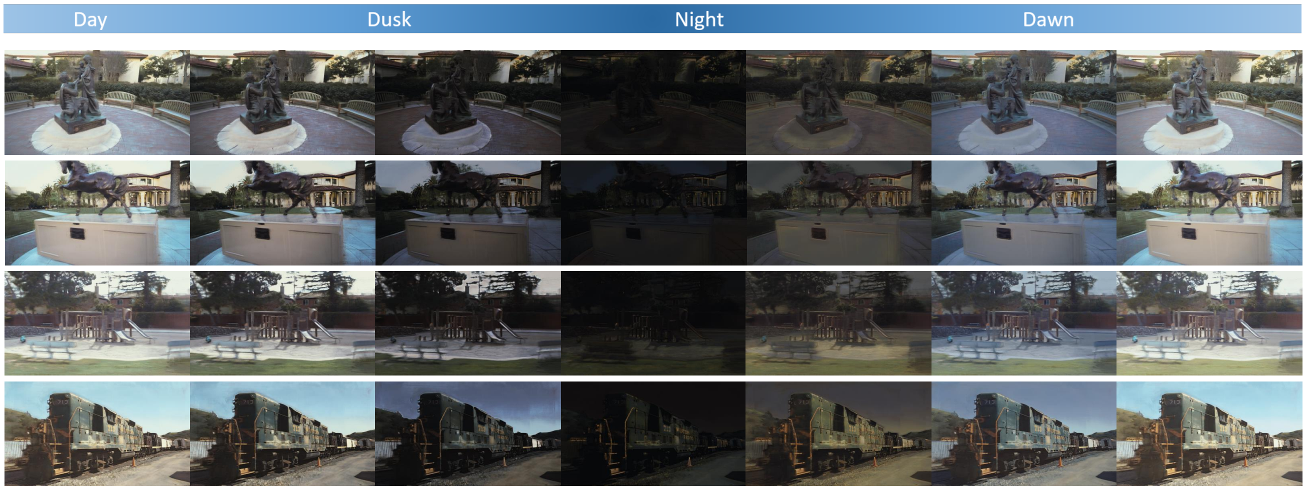


Figure 7: Qualitative results on the T&T [5] dataset. We generate novel views at 7 different times to show the cyclic changes of a day. For each scene, 3 input images are utilized for 3D reconstruction in this experiment.

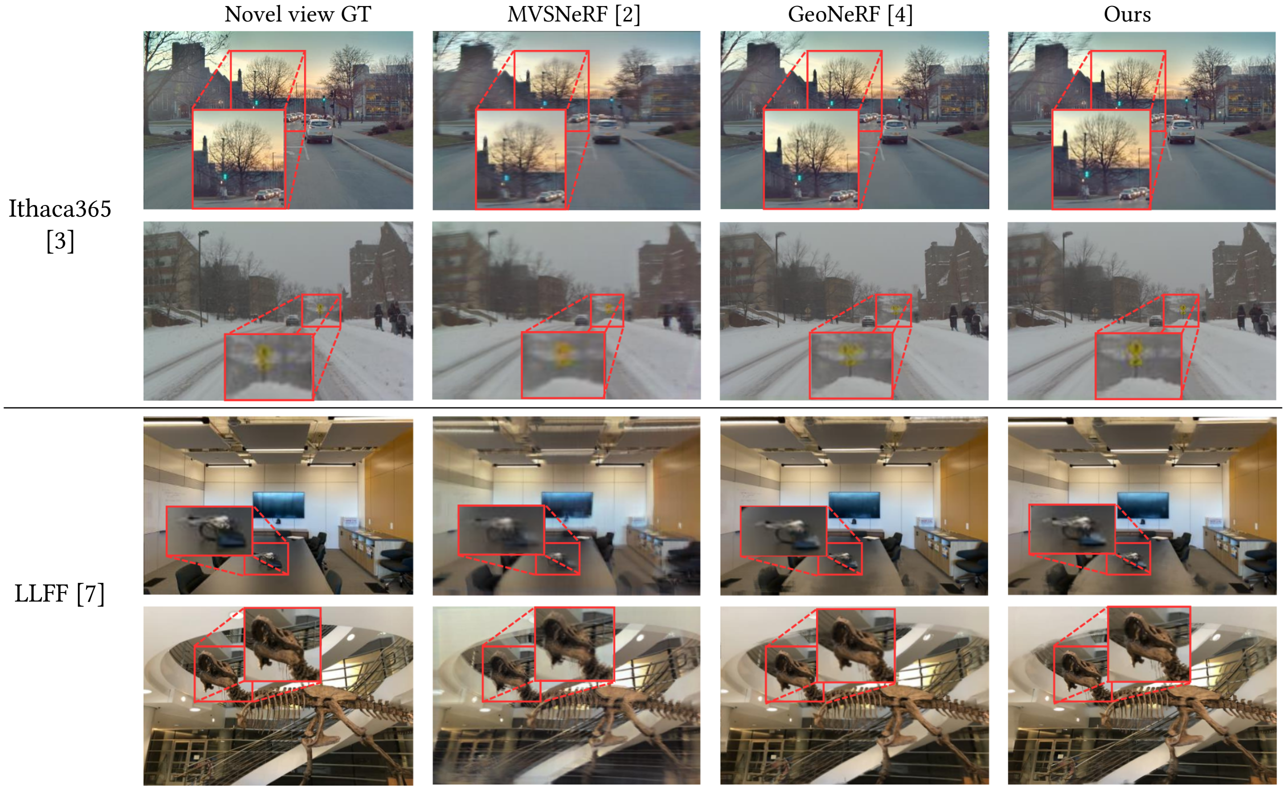


Figure 8: Qualitative results of pure view synthesis. We show the view synthesis results from MVSNeRF, GeoNeRF, and our model on the LLFF dataset and the ithaca365 dataset. Compared to these SOTA methods, TimeNeRF is able to produce clearer details.

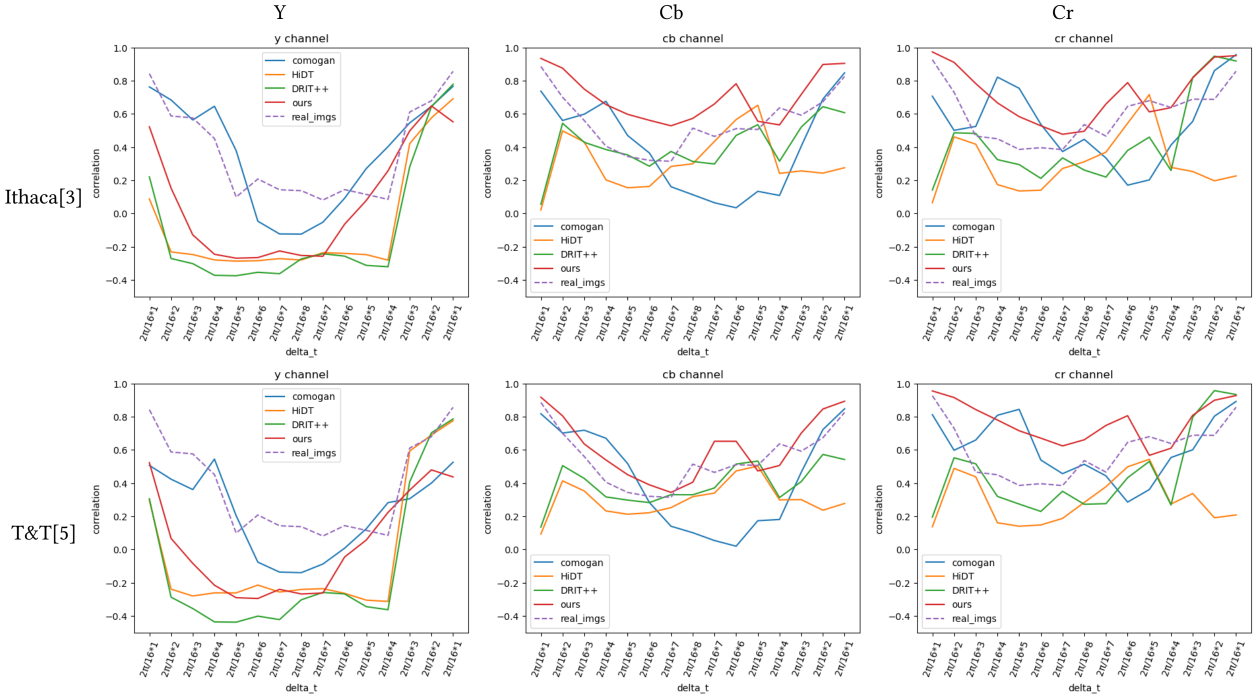


Figure 9: the mean histogram correlations in the Y, Cb, and Cr channels. The x-axis represents time differences, while the y-axis indicates correlation values. “real_imgs” correspond to frames extracted from multiple 24-hour videos. Ideally, the Y channel’s correlation should decrease as the time difference increases, whereas the Cb and Cr channels should exhibit relatively stable correlations compared to the Y channel.

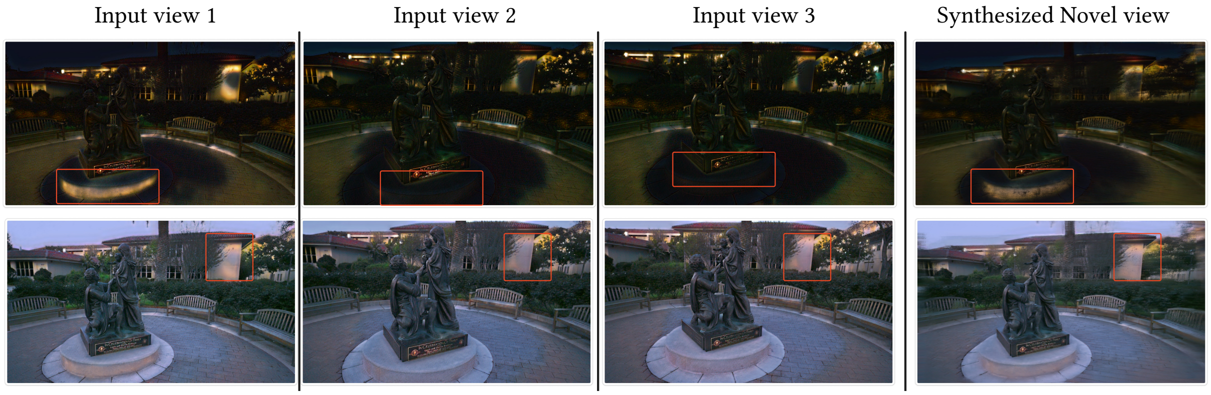


Figure 10: Style transfer, then novel view synthesis. We first transfer the style of the input views using CoMoGAN [8], and then use these style-transferred input views to synthesize a novel view through GeoNerf [4]. The red boxes in the images highlight regions where there are inconsistencies between the input views and the synthesized novel views produced by this alternative approach.