NeRF-MS: Neural Radiance Fields with Multi-Sequence

Peihao Li[1*] Shaohui Wang[1] Chen Yang[3] Bingbing Liu[2] Weichao Qiu[2†] Haoqian Wang[1,4†]
[1] Shenzhen International Graduate School, Tsinghua University
[2] Huawei Noah's Ark Lab [3]Shanghai Jiao Tong University
[4]Shenzhen Institute of Future Media Technology

Figure 1: **NeRF-MS** trains neural radiance fields from multiple sequences captured by different sensors and at different times, achieving better scene reconstruction by implicit modeling appearance styles from multi-sequence and separating transient contents from static scenes, such as the rider.

Abstract

Neural radiance fields (NeRF) achieve impressive performance in novel view synthesis when trained on only single sequence data. However, leveraging multiple sequences captured by different cameras at different times is essential for better reconstruction performance. Multi-sequence data takes two main challenges: appearance variation due to different lighting conditions and non-static objects like pedestrians. To address these issues, we propose NeRF-MS, a novel approach to training NeRF with multi-sequence data. Specifically, we utilize a triplet loss to regularize the distribution of per-image appearance code, which leads to better high-frequency texture and consistent appearance, such as specular reflections. Then, we explicitly model non-static objects to reduce floaters. Extensive results demonstrate that NeRF-MS not only outperforms state-of-the-art view synthesis methods on outdoor and synthetic scenes, but also achieves 3D consistent rendering and robust appearance controlling. Project page: https://nerf-ms.github.io/.

1. Introduction

Neural radiance field [20] demonstrates great success for novel view synthesis when given a photo collection. It works best when photos are from a single sequence, which means they are captured by the same camera at the same time. Multiple sequences are collections of photos captured by different cameras at different times as shown in Fig 1. Using multiple sequences is important for creating a better 3D reconstruction of a scene. By combining images captured from different cameras and at different times, we can fill in the gaps and improve the quality of poorly reconstructed regions. Furthermore, we can use videos captured at different seasons or times to show the changes or variations in the scene over time. Additionally, videos captured by different visitors and cameras at the same event can provide different perspectives or angles of the scene.

Multi-sequence data raise challenges for existing methods. The first is **appearance variance**. Sequences are captured under different conditions, e.g. sensor, lighting, and weather. Assigning a per-image appearance code [19, 5, 33] can handle extreme appearance changes. However, this gives the model too big flexibility, so the learned appearance code will overfit image content and camera pose, which

[*] Work done during an internship at Huawei Noah's Ark Lab.
[†] Co-corresponding Author.

results in 3D inconsistent rendering and loss of complex texture in novel views, see Fig. 3. Moreover, it leads to ghosting during interpolation between different appearance codes. The second is **sequence transient**. Previous methods split a scene into static and transient parts [30]. For multi-sequence data, an object can be static in the whole sequence (parked vehicle, standing pedestrian) but absent from other sequences. This kind of object can not be successfully categorized as transient, leading to a floater in the scene as shown in Fig. 5.

We propose NeRF-MS, a novel method for building neural radiance fields using multiple sequences. Firstly, to address the overfitting appearance code due to high flexibility, we introduce a triplet loss that assumes similarity between images within a sequence and diversity between sequences. With the triplet loss that constrains the distribution of appearance latent codes, our method can reduce ambiguity in the reconstruction process and improve texture and reflection fidelity. Since building a better latent space, we can perform more natural appearance interpolation across different sequences. Secondly, We propose a transient decomposition module to separate transient objects from static scenes better. By explicitly modeling sequence and image transients separately, we prevent overfitting to sequence transients and improve geometric reconstruction quality.

Our experiments on real-world outdoor dataset [29] and synthetic dataset [20] demonstrate that our approach achieves state-of-the-art performance in multi-sequence scenes by reconstructing 3D consistent appearance and reducing ghosting artifacts. Moreover, by controllable experiments on the synthetic dataset, we show that our method is robust against various multi-sequence settings and outperforms the baseline consistently.

In conclusion, our contribution can be summarized as follow:

- A novel framework enabling neural radiance fields to perform novel view synthesis with multi-sequence images captured in the wild.
- A triplet loss to regularize appearance code for reducing the ambiguity of appearance variation, allowing high-fidelity rendering and controllable appearance.
- A sequence transient decomposition module to separate transient objects and static scenes for reducing floaters in novel view synthesis.

2. Related Work

2.1. Novel View Synthesis

Novel view synthesis (NVS) is a popular research area in computer vision and graphics, involving generating target views from input images. Initial approaches use image-based rendering (IBR), in which models generate target views from a set of input images. Later some methods constructed light fields [15] or proxy geometry [6, 9, 28, 27] from posed inputs and synthesized views through resampling or blending warped inputs. However, these methods required dense input images and were limited by the quality of 3D reconstruction and sparse input data. Recent advancements in deep learning have facilitated the use of neural networks for computing radiance values corresponding to a given 3D position and direction, leading to the synthesis of high-quality novel views. Specifically, Neural Radiance Fields (NeRF) [20] techniques, which employ coordinate Multi-Layer Perceptrons (MLP), model a scene and render it using volumetric rendering, yielding exceptional quality view synthesis results. NeRF achieves impressive view synthesis results and has inspired a lot of related researches for further improvements, such as accelerating rendering and optimization [7, 8, 25, 38, 21, 4], handling color variations [19, 33], optimizing camera poses [16, 37, 35, 1], and improving performance with few input images [12, 14, 22].

2.2. Calibrated Volume Rendering

NeRF directly optimizes a neural volumetric scene representation to match all input images using gradient descent on a rendering loss. Thus, imperfect input data can negatively impact synthesized novel views. To address this, several methods have been proposed to improve performance with relaxed assumptions in volume rendering, such as varying camera conditions [19, 5, 30, 26], occluded objects [5, 17], and distractors [30]. NeRF in the Wild (NeRF-W) [19] uses web images in a wild setting for reconstruction and introduces a handling mechanism by appearance and transient embeddings to deal with varying camera conditions and occluded objects. Block-NeRF [33] adds appearance embeddings, learned pose refinement, and controllable exposure to NeRF to make it robust to data captured over months under different environmental conditions. Ha-NeRF [5] introduces an appearance hallucination module and an anti-occlusion module to handle time-varying appearances and complex occlusions in tourism images. RobustNeRF [30] models distractors as outliers in the optimization problem used for NeRF training, effectively removing them from the scene and reducing artifacts.

Some methods aim to address images with varying exposures and achieve high dynamic range (HDR) rendering from low dynamic range (LDR) images. RawNeRF [20] modifies NeRF to train directly on linear raw images, preserving the scene's full dynamic range and enabling HDR view synthesis tasks from extremely noisy input images captured in near-darkness. HDR-NeRF [11] and HDR-Plenoxels [13] recover the high dynamic range neural radiance field from a set of low dynamic range images with varying exposures using a differentiable tone mapper, achieving high-quality rendering for both HDR and LDR.

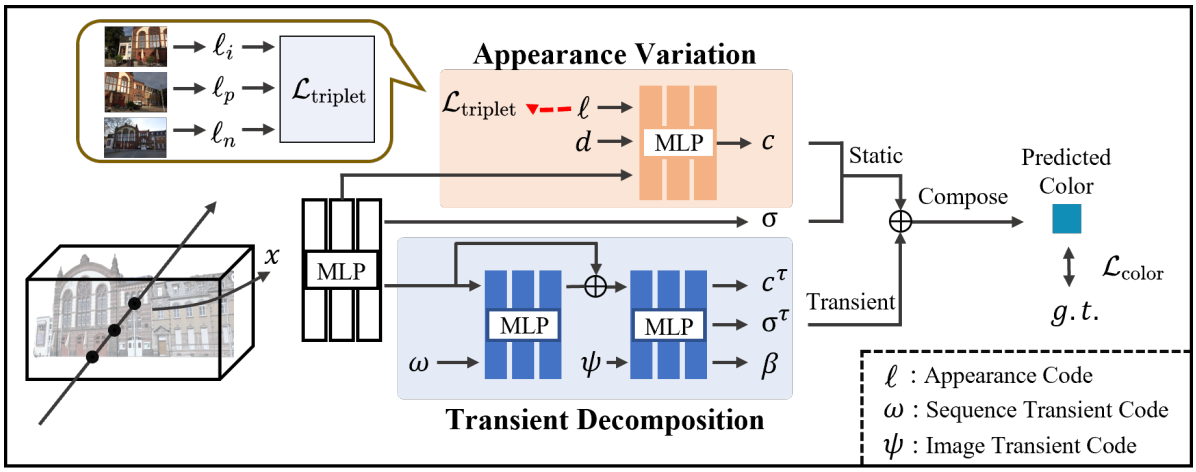

Figure 2: **Overview of NeRF-MS.** Our appearance variation module outputs the static color c based on the per-image appearance code ℓ_i. We use triplet loss to regularize the distribution of appearance code. By utilizing the image transient code ψ_i and sequence transient code ω_k, the transient decomposition module can effectively generate the color, density, and uncertainty for non-static objects. The static and transient components are then integrated to obtain the pixel's color and uncertainty. Finally, we employ a color loss to supervise the radiance field.

Some works target to address image degradation caused by defocus or motion blur. Deblur-NeRF [18] proposes an analysis-by-synthesis approach using a deformable sparse kernel module to recover a sharp NeRF from blurry inputs caused by defocus or motion blur. BAD-NeRF [35] jointly learns the parameters of NeRF and recovers camera motion trajectories during exposure time to be robust to severe motion-blurred images and inaccurate camera poses.

In this paper, we address the challenges of optimizing NeRF with multi-sequence, which relaxes the requirements on photometric consistency in trainging data.

3. Preliminaries

3.1. NeRF

NeRF [20] represents the object to be reconstructed as a neural radiance field, which takes 3D coordinates (x, y, z) and viewing direction (θ, ϕ) as input and map them to color $\mathbf{c}$ and density σ with MLP:

$$[\sigma(t), \mathbf{z}(t)] = F_{\theta_1}(\gamma(\mathbf{x})), \tag{1}$$

$$\mathbf{c}(t) = F_{\theta_2}(\gamma(\mathbf{d}), \mathbf{z}(t)), \tag{2}$$

where $\theta = \{\theta_1, \theta_2\}$ is the parameter of MLP, the $\gamma(\cdot)$ represents a positional encoding. Given rays denoted as $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ pass from camera origin $\mathbf{o}$ through pixels of images along ray direction $\mathbf{d}$. A pixel's predicted color can be represented in the following form:

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{k=1}^{K} T(t_k)\, \alpha(\sigma(t_k)\, \delta_k)\, \mathbf{c}(t_k), \tag{3}$$

where $\alpha(x) = 1 - \exp(-x)$, $\delta_k = t_{k+1} - t_k$ is the sampling interval and $T(t_k) = \exp\left(-\sum_{k'=1}^{k-1} \sigma(t_{k'})\, \delta_{k'}\right)$ is the transmittance of sampled point.

All parameters of models are optimized by decreasing the difference in pixels' color between generated images and reference images:

$$\mathcal{L} = \sum_{\mathbf{r} \in R} \|\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r})\|_2^2, \tag{4}$$

where R denotes a set of rays in each batch.

3.2. NeRF in the Wild

NeRF-W [19] performs better when synthesizing novel views of complex in-the-wild scenes using unstructured image sets. Combining the unique learned appearance code and transient code of each image with viewing direction and sampled points, NeRF-W produces static and transient colors and densities as well as uncertainty β, which are encoded to eliminate temporary occlusions.

To disentangle static and transient components, not only uncertainty masks β are considered, but the rendering equation is also modified as follows:

$$\hat{\mathbf{C}}_i(r) = \sum_{k=1}^{K} T_i(t_k) \left(\alpha(\sigma_i(t_k)\, \delta_k)\, \mathbf{c}_i(t_k) + \alpha\left(\sigma_i^{(\tau)}(t_k)\, \delta_k\right) \mathbf{c}_i^{(\tau)}(t_k)\right), \tag{5}$$

$$\text{where} \quad T_i(t_k) = \exp\left(-\sum_{k'=1}^{k-1} \left(\sigma_i(t_{k'}) + \sigma_i^{(\tau)}(t_{k'})\right) \delta_{k'}\right), \tag{6}$$

in which σ_i, c_i are density and radiance, and $\sigma_i^{(\tau)}$, $c_i^{(\tau)}$ are their transient counterparts. This modification results

Figure 3: **Effectiveness of triplet loss.** Our method can reconstruct fine details (window dividers) and 3D consistent reflection (on windows and bulldozer tracks) by utilizing triplet loss to prevent appearance code overfitting.

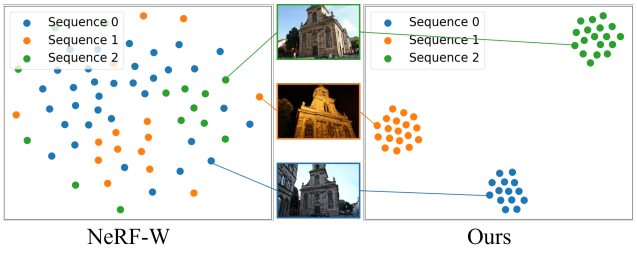

Figure 4: **T-SNE visualization of appearance codes.** We show that, without triplet loss, the appearance codes from different sequences are overlapped due to overfitting.

in larger density clustering near static objects, which helps remove transient objects.

4. Method

Our method aims to reconstruct neural radiance fields with an image collection $\{I_i\}_{i=1}^{N}$ containing multiple sequences $\{S_k\}_{k=1}^{M}$, which are captured at different times or with different sensors. Specifically, we build neural radiance fields by optimizing sequence-consistent appearance latent code with triplet loss in Sec. 4.1. To better separate non-static objects, we model different categories of non-static objects using sequence transient code and image transient code in Sec. 4.2, which reduced artifacts in the rendered results. Fig. 2 presents an overview of the proposed method's workflow and structure.

4.1. Triplet Loss for Appearance Regularization

For reconstructing 3D scenes with multiple sequences, it is important to disentangle the varying appearance from the static geometry. We follow [19] to fit appearance latent code $\ell_i \in \mathbb{R}^{n_1}$ for each image I_i using Generative Latent Optimization [2]. Thus, with the appearance embedding as MLP's input, we extend the Eq. 2 as follows:

$$\mathbf{c}_i\left(t\right) = F_{\theta_2}\left(\gamma\left(\mathbf{d}\right), \mathbf{z}\left(t\right), \ell_i\right) \tag{7}$$

Due to the high degree of freedom, optimizing one appearance code for each image leads to overfitting when the training dataset lacks appearance diversity, which results in poor generalization performance in test views. Additionally, Per-image appearance cannot model the similarity between images within a sequence, preventing 3D consistent novel view synthesis.

To address this issue, we propose a triplet loss, which is largely used in metric learning [32, 10, 31, 3], to preserve the sequence appearances disentanglement:

$$\mathcal{L}_{\text{triplet}} = \frac{1}{N}\sum_{i=0}^{N-1}\max\left(\|\ell_i - \ell_p\| - \|\ell_i - \ell_n\| + m, 0\right), \tag{8}$$

where m is a hyper-parameter to adjust the constant margin between inner-sequence distance and inter-sequence distance. Appearance codes ℓ_p and ℓ_n represent the positive sample and negative sample, which means that (ℓ_i, ℓ_p, ℓ_n) corresponds to images (I_i, I_p, I_n), while $I_i, I_p \in S_k$ and $I_n \notin S_k$.

Compared to optimizing per-sequence appearance code trivially, the triplet loss function is robust against subtle appearance variance between images within the same sequence, which is common due to white balance, exposure, and lens flare. Table. 2 shows that such a method won't work.

Optimizing per-image appearance code weakens multi-view consistency constraints and leads to overfitting. Our method overcomes this drawback with the regularization of the appearance code. Therefore, as illustrated in Fig. 3, our method can reconstruct high-frequency texture and view-dependent effects such as reflections by integrating multi-view information.

Our method can construct reasonable appearance latent space as shown in Fig. 4, which is consistent with the prior that the appearance of pictures between sequences is similar. To regularize the latent space more concisely is essential to make robust novel view synthesis and smooth appearance interpolation, as shown in Sec.5.3.

4.2. Transient Decomposition

In a multi-sequence task, a scene comprises three parts, "static components", "sequence transients" and "image transients". Since sequence transient keep static within a sequence such as a parked vehicle, current works [19, 33, 5] can't separate the sequence transients from the static scene as shown in Fig. 5, which leads to artifacts and ghosting in the novel view.

GT NeRF-W Ha-NeRF Ours

Figure 5: **Transient Decomposition**. In the provided GT diagram, we label sequence transient region as yellow and image transient region as green. Subsequently, we present the decomposition performance of various algorithms alongside their corresponding depth maps. Our proposed method not only addresses the challenging yellow region that SOTA methods struggle with, but also improves the details in green area. Furthermore, our method effectively preserves the intricate geometric details of the static region, resulting in more accurate depth estimation.

We introduce a transient decomposition module to better disentangle the static geometry and transients by explicitly modeling the sequence transients. We leverage two kinds of learned transient latent code to model non-static objects: sequence transient code $\omega_k \in \mathbb{R}^{n_2}$ for each sequence and per-image transient code $\psi_i \in \mathbb{R}^{n_3}$. We encode the transient field in two MLP with parameter $\{\theta_3, \theta_4\}$ as follows:

$$\mathbf{z}'(t) = \mathbf{z}(t) + F_{\theta_3}(\mathbf{z}(t), \omega_k), \tag{9}$$

$$\left[\mathbf{c}_i^{(\tau)}(t), \sigma_i^{(\tau)}(t), \beta_i(t)\right] = F_{\theta_4}(\mathbf{z}'(t), \psi_i), \tag{10}$$

where $\mathbf{z}(t)$ is the output of Eq. 1, ω_k corresponds to sequence S_k and ψ_i corresponds to image $I_i \in S_k$. By disentangling the sequence transient and image transient, our method encourages NeRF to fit the non-static objects with transient codes instead of frame-dependent appearance.

The output of transient decomposition module along the ray $\mathbf{r}$ will be composed with the static radiance c_i and density σ_i to obtain the pixel color $\hat{\mathbf{C}}_i(\mathbf{r})$ and uncertainty $\hat{\beta}_i(\mathbf{r})$. Following [19], the color loss is proposed as follows:

$$\mathcal{L}_{\text{color}} = \sum_{\mathbf{r}\in R}\left(\frac{\|\hat{\mathbf{C}}_i(\mathbf{r}) - \mathbf{C}_i(\mathbf{r})\|_2^2}{2\hat{\beta}_i(\mathbf{r})} + \frac{\log\hat{\beta}_i(\mathbf{r})}{2} + \frac{\lambda_u}{K}\sum_{k=1}^{K}\sigma_i^{(\tau)}(t_k)\right), \tag{11}$$

where λ_u is a loss weight to adjust the penalty for the transient amount. In the test stage, we omit the transient decomposition module and only render $\mathbf{c}_i$ and σ_i for novel view synthesis.

4.3. Optimization

To obtain NeRF-MS, we integrate the constraints mentioned above and perform joint training of network parameters $\{\theta_1, \theta_2, \theta_3, \theta_4\}$ and the learned latent codes $\{\ell_i\}_{i=1}^N, \{\psi_i\}_{i=1}^N, \{\omega_k\}_{k=1}^M$ to minimize the full loss function:

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \lambda\mathcal{L}_{\text{triplet}}, \tag{12}$$

which is a linear combination of color loss and triplet loss with loss weights λ.

5. Experiment

5.1. Datasets

NeRF-OSR. The NeRF-OSR dataset [29] is a benchmark dataset for outdoor scene relighting, containing 8 outdoor scenes captured from 3240 viewpoints in 110 different recording sessions. We select 3 sequences under different lighting conditions for 4 scenes each. Specifically, we select the first frame of every eighth frames as the testing set. Furthermore, during testing, we utilize semantic segmentation to generate masks for ground truth in order to eliminate transient occlusions, and focus on evaluating the performance of our algorithm in reconstructing the static components.

Synthetic Dataset. For controllable experiments, we construct a synthetic dataset based on NeRF Synthetic dataset [20]. We first randomly split the 100 training views into several sequences $\{S_k\}_{k=1}^M$. Then, similar to NeRF-W [19], we apply random color perturbation for each sequence, except the first sequence in which appearance code of the first image is used to render test views. By adjusting image count per sequence, while fixing the overall number of training images to 100, we show the robustness of our method against different multi-sequence settings in Sec.5.5. Please refer to our supplementary for more controllable experiments and complete details.

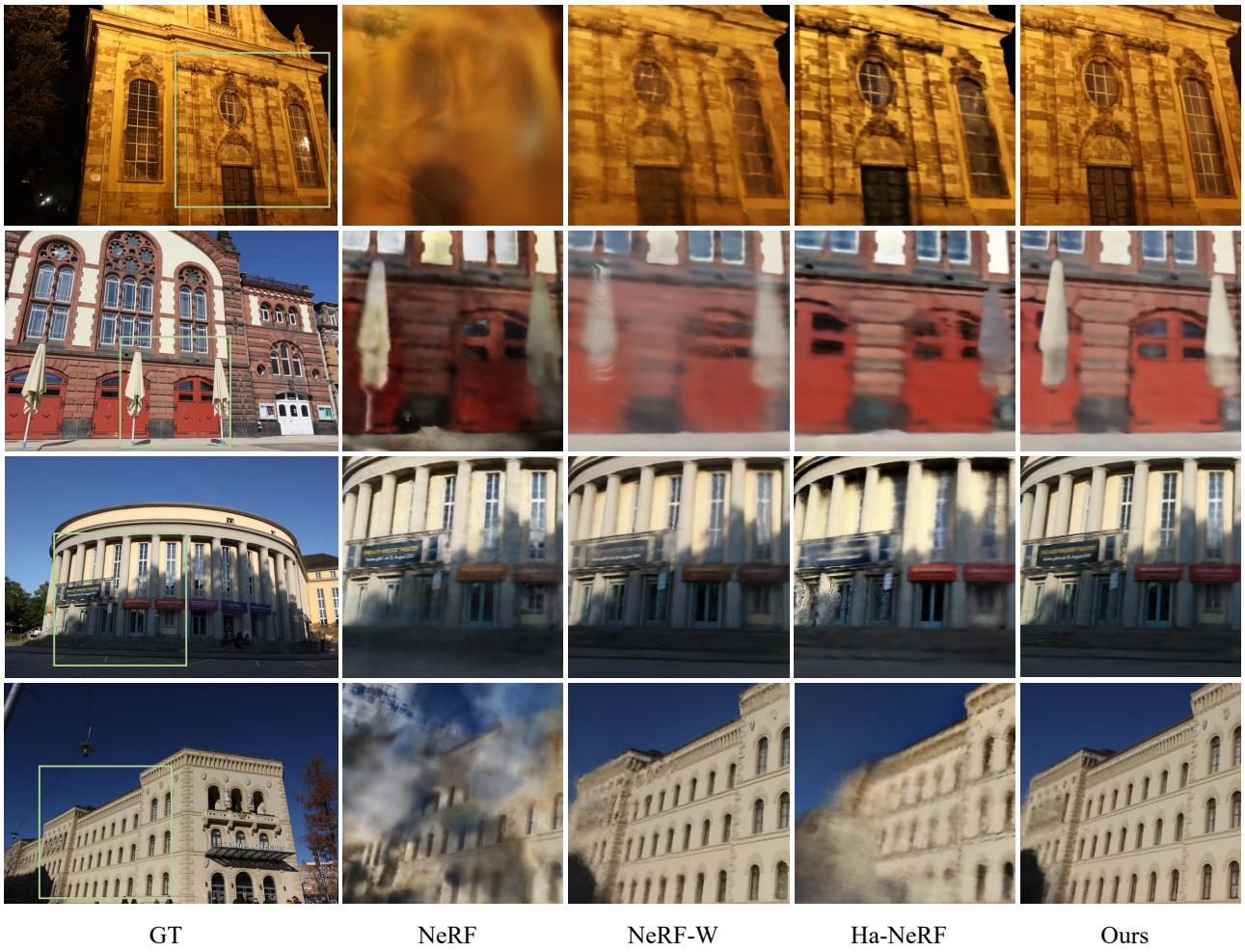

Figure 6: **Qualitative results on NeRF-OSR dataset.**

Figure 7: **Qualitative results of appearance interpolation.** We make interpolation between appearance codes of two training images.With overfitting appearance code, NeRF-W and Ha-NeRF suffer from ghosting in the sky on the right.

	stjohann			lwp			st			europa		
	↑PSNR	↑SSIM	↓LPIPS	↑PSNR	↑SSIM	↓LPIPS	↑PSNR	↑SSIM	↓LPIPS	↑PSNR	↑SSIM	↓LPIPS
NeRF [20]	14.891	0.4319	0.6388	11.510	0.4677	0.5742	17.196	0.5143	0.5017	17.492	0.5508	0.5034
NeRF-W [19]	21.230	0.6673	0.4255	19.607	0.6159	0.4453	20.310	0.6067	0.4380	20.000	0.6899	0.3397
Ha-NeRF [5]	17.187	0.6857	0.3309	20.027	0.6850	0.3647	17.298	0.5378	0.4831	17.790	0.6323	0.4210
Ours	**22.839**	**0.7933**	**0.2347**	**21.904**	**0.7187**	**0.3361**	**20.675**	**0.6304**	**0.4021**	**21.028**	**0.7214**	**0.2939**

Table 1: **Quantitative results on NeRF-OSR dataset.** We compare with state-of-art NeRF-like methods. Our method can significantly improve the performance with multi-sequence images.

5.2. Implementation Details

Our implementation of NeRF and NeRF-W is based on [24], while our implementation of Ha-NeRF follows [5]. The static radiance field consists of 8 layers with 256 channels and a 128 channels layer to generate σ and $\mathbf{c}$. Two MLPs with 2 layers in the Transient Decomposition module are followed by ReLU, and we generate color, density, and uncertainty by one fully-connected layer. Our per-image appearance vector with 48 dimensions is constrained by triplet loss with margin $m = 2$, and the weight of triplet loss λ is set to 0.01. Dimensions of sequence transient code and image transient code are both 16. We train the full model on 8 Nvidia 2080Ti GPUs for 200k steps with a batch size of 1024 and downsample all the images by 2 times during training and evaluation. We use MindSpore [34] and PyTorch [23] for the implementation and code will be released.

5.3. Comparisons with the State-of-the-art Methods

Baselines. We evaluate our NeRF-MS and other SOTA NeRF-like baselines as follows: a) vanilla NeRF which presents volume rendering but requires strict illumination conditions; b) NeRF-W [19] which is able to train radiance field using unstructured image sets; c) Ha-NeRF [5] which uses a convolutional neural network to extract appearance code and an occluded network to predict 2D transient masks.

Evaluation. We use a number of different assessment metrics to measure visual quality of our novel view synthesis, such as the Peak Signal-to-noise Ratio(PSNR), the Structural Similarity Index Measure(SSIM [36]) and the Learned Perceptual Image Patch Similarity(LPIPS [39]). We follow the setting in [5] and optimize the appearance code on the left half of each test image for our method and NeRF-W and report metrics on the right half, while Ha-NeRF takes the whole test image as input to fetch appearance feature.

Quantitative results are shown in Table. 1. With multi-sequence training images, our method outperforms the baselines on PSNR, SSIM and LPIPS on NeRF-OSR dataset. By regularizing per-image appearance code, we strengthen the multi-view consistent constraint so that our method can perform robust rendering and reconstruct 3D consistent appearance, especially some view-dependent effects like specular reflections as shown in Fig. 3. For better geometry reconstruction, we propose a transient decomposition module to enhance the modeling capability of sequence transients, realizing thorough separation of static and transient objects. The quantitative results on the real-world dataset announce our method achieves state-of-art performance in multi-sequence tasks.

Qualitative results are shown in Fig. 6. Our method significantly improves the rendering performance in test views with multiple sequences as training data compared with the baselines. NeRF suffers from severe ghosting and artifacts since it has a strong assumption for photometric consistency. NeRF-W optimizes per-image appearance and transient codes for each image without regularization and assumption and suffers from overfitting in multi-sequence tasks due to a lack of diversity of appearance variations and the limited number of training views. The overfitting appearance embedding leads to a weak ability to reconstruct high-frequency texture and geometry details (examples 1, 2, 4, Fig. 6). Ha-NeRF uses a CNN to fetch the appearance features from training images. Similarly, CNN is prone to overfit in image content, resulting in global color bias and blurry rendering (example 1, 3, 4, Fig. 6). Besides, Ha-NeRF leverages MLP to predict 2D masks for transient objects instead of transient fields. The 3D inconsistent masks result in an excessive separation of geometry, leading to the loss of fine details in the reconstruction (example 2 in Fig. 6 and roof in Fig.7).

Controllable Appearance. Our method achieves controllable appearances by interpolation between different sequences. In Fig. 7, we present some images rendered by interpolated appearance code from the leftmost image and rightmost image. Our method performs smooth appearance style transfer by building a better latent space, while NeRF-W and Ha-NeRF suffer from ghosting artifacts in unseen regions (the sky on the right). It reveals that per-image appearance code without regularization is prone to overfitting on the training views and cannot robustly render and interpolate in novel viewpoints.

5.4. Ablation Studies

Results in Table. 2 show that each component is beneficial to multi-sequence fusion tasks. This illustrates the effectiveness of regularization on appearance code distribution and explicitly modeling sequence transients.

	↑PSNR	↑SSIM	↓LPIPS
NeRF-W	20.287	0.6450	0.4121
Ours w/o Triplet Loss	20.848	0.6932	0.3410
Ours w/ Seq App Code	20.366	0.6780	0.3645
Ours w/o Sequence Transient	21.224	0.6846	0.3572
Full Model	**21.611**	**0.7160**	**0.3167**

Table 2: **Ablation study on NeRF-OSR dataset.**

	Label Error	↑PSNR	↑SSIM	↓LPIPS
NeRF-W	☐	21.230	0.6673	0.4255
Ha-NeRF	☐	17.187	0.6857	0.3309
Ours	☐	22.839	0.7933	0.2347
Ours	☑	21.668	0.7315	0.2989

Table 3: **Robustness against sequence label error.** The performance of our method decreases with sequence partition error, but still better than the other methods.

Without Triplet Loss. We remove the triplet loss in this experiment. Without triplet loss, appearance code tends to overfit image content and camera pose, leading to global color shift and loss of high-frequency texture in test views. Furthermore, without the regularization of the embedding distribution, neural radiance fields can't perform 3D consistent rendering and reconstruct view-dependent effects, as shown in Fig. 3.

With Sequence Appearance Code. In this experiment, we replace the per-image appearance code with a per-sequence appearance code. This experiment's result is worse than "w/o Triplet Loss", indicating that sequence appearance code harms multi-sequence tasks. The main reason is that such a trivial method can't model the difference between frames in the same sequence.

Without Transient Decomposition Module. In the experiment, we remove sequence transient code, only optimizing image transient code like NeRF-W. Without explicitly modeling the sequence transients, current methods struggle to disentangle sequence transients from static scenes, causing artifacts and floaters, as shown in Fig. 5.

5.5. Robustness Analysis

Robustness against Error Sequence Label. Sequence label is free to obtain, such as by timestamp. However, in some extreme cases, there existing some inaccurate labels, for example, the scene appearance will change dramatically due to turning on the light.

By labeling sequence 0 and sequence 1 in "stjohann" scene as the same sequence, we evaluate our method's robustness against the sequence label error. As shown in Table. 3, with sequence label error, our method still shows performance beyond baselines, which proves our method's robustness against label error and drastically changes within a sequence. The main reason is that triplet loss allows variance between the images in the same sequence.

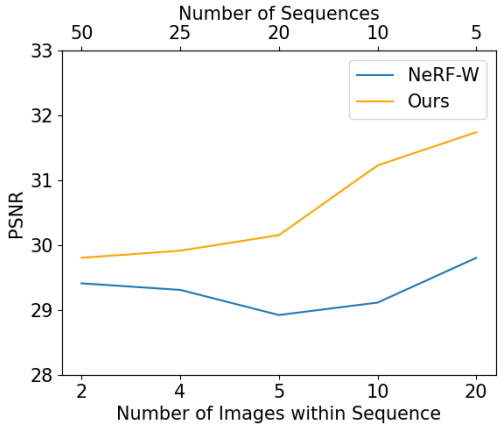

Figure 8: **Evaluation with various multi-sequence settings.** By fixing the total number of training images as 100, we adjust the number of sequences and number of images within a sequence, e.g. 50 sequences with 2 images per sequence and 5 sequences with 20 images per sequence.

Robustness against Various Multi-sequence Settings. We discuss the robustness of our method against different multi-sequence settings. We evaluate our method and baselines with different numbers of sequences and numbers of images per sequence on synthetic dataset, while fixing the number of whole training views to 100. Fig. 8 shows that our method consistently outperforms NeRF-W even in a NeRF-W like setting (50 sequences with 2 images per sequence). Therefore, the sequence suitable for our method is not only a long video or hundreds of frames collection but also an image set with several images.

6. Conclusion

We present NeRF-MS, a novel framework to train neural radiance fields with multi-sequence images. NeRF-MS utilizes a triplet loss to avoid overfitting for appearance code and proposes sequence transient code for better non-static object separation. Our method can reconstruct 3D consistent reflection and achieve controllable appearance. The sequence transient code effectively separates non-static components, while existing methods are unable to handle this phenomenon. Experiments show that NeRF-MS achieves state-of-art novel view synthesis effects with multiple sequences. We believe our method represents a significant advancement in expanding the potential applications of NeRF.

Acknowledgments. This work was funded through National Key Research and Development Program of China (Project No.2022YFB36066), in part by the Shenzhen Science and Technology Project under Grant (JCYJ20220818101001004, CJGJZD20200617102601004). We thank MindSpore [34] for the partial support to this work.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022. 2

[2] Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. *arXiv preprint arXiv:1707.05776*, 2017. 4

[3] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018. 4

[4] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 333–350. Springer, 2022. 2

[5] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12943–12952, 2022. 1, 2, 4, 7

[6] Inchang Choi, Orazio Gallo, Alejandro Troccoli, Min H Kim, and Jan Kautz. Extreme view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7781–7790, 2019. 2

[7] Forrester Cole, Kyle Genova, Avneesh Sud, Daniel Vlasic, and Zhoutong Zhang. Differentiable surface rendering via non-differentiable sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6088–6097, 2021. 2

[8] Stephan J Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. Fastnerf: High-fidelity neural rendering at 200fps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14346–14355, 2021. 2

[9] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics (TOG)*, 37(6):1–15, 2018. 2

[10] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 4

[11] Xin Huang, Qi Zhang, Ying Feng, Hongdong Li, Xuan Wang, and Qing Wang. Hdr-nerf: High dynamic range neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18398–18408, 2022. 2

[12] 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*, pages 5885–5894, 2021. 2

[13] Kim Jun-Seong, Kim Yu-Ji, Moon Ye-Bin, and Tae-Hyun Oh. Hdr-plenoxels: Self-calibrating high dynamic range radiance fields. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 384–401. Springer, 2022. 2

[14] Mijeong Kim, Seonguk Seo, and Bohyung Han. Infonerf: Ray entropy minimization for few-shot neural volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12912–12921, 2022. 2

[15] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996. 2

[16] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751, 2021. 2

[17] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7824–7833, 2022. 2

[18] Li Ma, Xiaoyu Li, Jing Liao, Qi Zhang, Xuan Wang, Jue Wang, and Pedro V Sander. Deblur-nerf: Neural radiance fields from blurry images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12861–12870, 2022. 3

[19] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM 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*, pages 7210–7219, 2021. 1, 2, 3, 4, 5, 7

[20] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 3, 5, 7

[21] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 2

[22] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM 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*, pages 5480–5490, 2022. 2

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 7

[24] Chen Quei-An. Nerf pl: a pytorch-lightning implementation of nerf. *URL https://github. com/kwea123/nerf_pl*, 2020. 7

[25] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In *Proceedings of the IEEE/CVF In-*

ternational Conference on Computer Vision, pages 14335–14345, 2021. 2

[26] Konstantinos Rematas, Andrew Liu, Pratul P Srinivasan, Jonathan T Barron, Andrea Tagliasacchi, Thomas Funkhouser, and Vittorio Ferrari. Urban radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12932–12942, 2022. 2

[27] Gernot Riegler and Vladlen Koltun. Free view synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, pages 623–640. Springer, 2020. 2

[28] Gernot Riegler and Vladlen Koltun. Stable view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12216–12225, 2021. 2

[29] Viktor Rudnev, Mohamed Elgharib, William Smith, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. Nerf for outdoor scene relighting. In *European Conference on Computer Vision (ECCV)*, 2022. 2, 5

[30] Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J Fleet, and Andrea Tagliasacchi. Robustnerf: Ignoring distractors with robust losses. *arXiv preprint arXiv:2302.00833*, 2023. 2

[31] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 4

[32] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016. 4

[33] 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*, pages 8248–8258, 2022. 1, 2, 4

[34] Huawei Technologies. Mindspore. https://www.mindspore.cn/. 7, 8

[35] Peng Wang, Lingzhe Zhao, Ruijie Ma, and Peidong Liu. Bad-nerf: Bundle adjusted deblur neural radiance fields. *arXiv preprint arXiv:2211.12853*, 2022. 2, 3

[36] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 7

[37] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021. 2

[38] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5752–5761, 2021. 2

[39] Richard Zhang, Phillip Isola, and Alexei A Efros. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7