GDNeRF: Generalizable Depth-based NeRF for sparse view synthesis.

Sergio Montoya de Paco
i2CAT Foundation
Barcelona, Spain
sergio.montoya@i2cat.net

Ivan Huerta
i2CAT Foundation
Barcelona, Spain
ivan.huerta@i2cat.net

Josep Escrig
i2CAT Foundation
Barcelona, Spain
josep.escrig@i2cat.net

***Abstract*—Generalizable NeRFs typically aim to synthesize novel views of unseen scenes, often requiring multiple closely positioned input views. GDNeRF addresses this limitation by enabling high-quality view synthesis with sparse input views that are widely spaced. Leveraging depth map information, GDNeRF constructs a multilevel probabilistic feature volume to generalize to new scenes effectively. A generative module enhances the volumetric renderer by synthesizing occluded or ambiguous areas, ensuring realistic results even in challenging conditions. Tested under sparse camera settings with up to 60-degree rotation differences along the Y-axis, GDNeRF outperforms state-of-the-art models by up to 60%. Additionally, it achieves real-time rendering at approximately 16 FPS, making it ideal for free-viewpoint video streaming applications with minimal camera setups. GDNeRF's innovative approach demonstrates significant advancements in sparse input scene rendering with a simple camera setup.**

***Index Terms*—Neural Radiance Fields, Few-shot New View Synthesis, Real-time Rendering, Sparse Cameras**

I. Introduction

Neural Radiance Field (NeRF) approaches have demonstrated exceptional performance in highly realistic novel view synthesis. The primary objective is to learn a renderable particle volume, facilitating novel view synthesis through raymarching. Many advancements have been made since their introduction to enhance their training, inference speed and rendering quality [1], [2]. However, these approaches learn an implicit representation of the scene with a few hundred images [3], which is a limitation in some applications. Despite numerous advancements, their ability to generalize to new scenes or unseen timeframes remains limited. To utilize generalizable NeRFs for free-viewpoint video generation from a few sparse views, two primary concerns must be addressed. Firstly, setting up a multi-view capturing environment with numerous views often poses a barrier due to the complexity and expenses associated with hardware. Compared to common scenes used by NeRFs, simple multi-view camera settings have few cameras, which are far from each other. Secondly, for such applications, the lengthy training times of NeRFs are prohibitive for real-time deployment on unseen scenes.

To model dynamic scenes, a line of works models dynamic scenes by proposing a deformation model to relate the scene at different time instants [4], [5], which achieve highly realistic results. However, they require a complex multi-camera setting. Other models are capable of learning the scene in the time domain using a single camera and assuming the type of deformation [6], [7]. Additionally, some provide good results on sequences exhibiting limited deformation without making any assumptions about the scene [8]. Nonetheless, these methods are unable to learn the deformation model in real-time and cannot generalize to unseen scenes.

Other works address real-time synthesis by using few multi-view image information, they are designed to train networks that create radiance fields from a few multi-view images, without requiring any training on the target scene [9], [10]. These methods demonstrate generalization capabilities to previously unseen scenes. These algorithms predict new views given a set of input images in a single forward pass, but they utilize NeRF's volumetric rendering at their core. However, these methods have struggled to generate new views at real-time speed due to the time-consuming volumetric rendering process. Recently, ENeRF [10] proposes a coarse-to-fine image-based rendering approach that utilizes estimates alpha-blending weights for the input images to generate the new view. Nonetheless, image-based rendering methods require a dense camera setup with a considerable number of cameras.

In this paper, we propose GDNeRF, a method capable of generating new views at high rendering quality using a sparse and simple camera setup with very few cameras. Our algorithm leverages depth map information to create a probabilistic feature volume that combines information from a few source images. Figure 1 provides an overview of GDNeRF. To process ambiguous and occluded information on the scene, we employ a 3D CNN as a generator of feasible feature volumes. We utilize style codes, akin to StyleGAN [11], to gradually incorporate new information into the feature volume. Following previous work [10], we adopt a coarse-to-fine strategy to esti-

This work has been mainly funded by by the European Union's Horizon Europe program under agreement nº 101092875 (DidymosXR project). This work has been partially funded by Agencia Estatal de Investigación (AEI), in the framework of Proyecto Estratégico Orientado a la Transición Ecológica y a la Transición Digital (TED) 2021, under agreement TED2021-131690B-C32 (REVOLUTION project), in the framework of Proyectos Generación de Conocimiento 2022, under agreement PID2022-140749OBI0 (EVOLVE project). It has funded by the Ministry for Digital Transformation and of Civil Service and by the "European Union NextGenerationEU/PRTR" within the call "UNICO-5G I+D: Programa de Universalización de Infraestructuras Digitales para la Cohesión – 2021" with Grant TSI-063000-2021-31 (6GTwin-Road). Likewise, this work has been partially funded by the European Union's Horizon Europe program under agreement nº 101070250 (XRECO project).

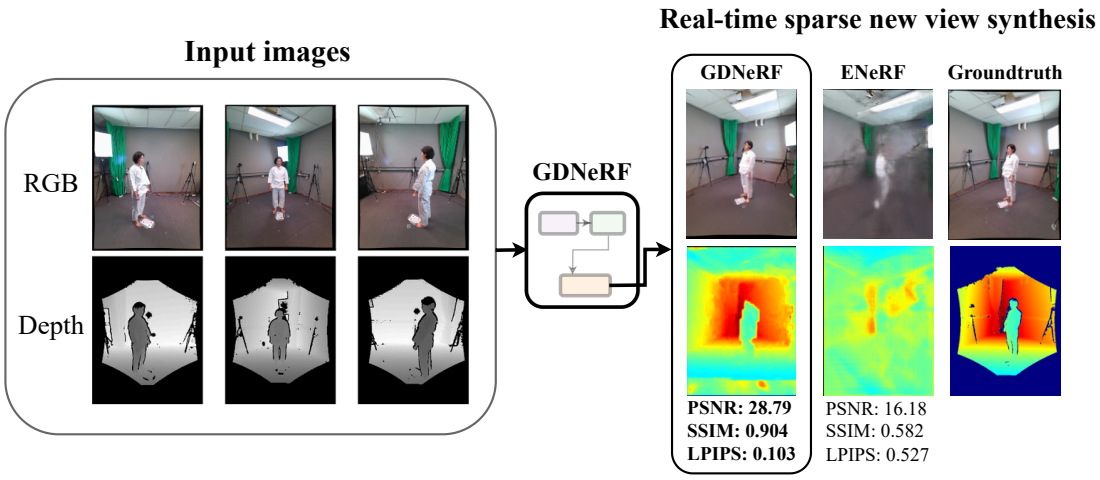

Fig. 1: **GDNeRF intro.** Given few sparse input views, GDNeRF is capable of synthesizing highly realistic target views. The example shows 3 sparse views of the CWI dataset and how our method is capable of synthesizing a new view. Our model leverages input depth information using a multilevel probabilistic feature volume and incorporates a generative component to produce feasible scene features rendered through volumetric rendering. It can generate new views in real time without needing to train the neural representation of the scene.

mate a depth interval for each pixel at lower resolutions, which improves its rendering efficiency. We evaluate our approach on the CWI [12] and ActorsHQ [13] datasets. The CWI dataset contains only 7 cameras, posing a challenge to state-of-the-art generalizable NeRFs. However, when synthesizing a new view, the top three closer views are enough to synthesize the target view. In such sparse camera settings, current approaches are unable to produce feasible results, often resulting in artifacts and blurred renders. Our approach significantly outperforms state-of-the-art methods and produces highly realistic results even in challenging camera setups. In summary, this work makes the following contributions:

- We propose a new algorithm that outperforms previous methods in generalized real-time new view synthesis in sparse camera settings.
- We introduce a new 3D multilevel probabilistic scene representation that models depth uncertainty. This representation integrates information from any number of source views.
- We introduce 3D feature supervision based on GANs for generalizable NeRFs, which leverages seen features while inferring occluded or ambiguous regions.
- We analyze the impact of camera sparsity on the CWI and ActorsHQ datasets for generalizable methods and demonstrate how our GDNeRF largely outperforms the current state-of-the-art in these settings.

II. Related work

New view synthesis. Light field rendering [14] and view-morphing techniques [15] were the primary new view synthesis techniques. NeRF [16] has garnered significant attention and has seen fast-paced improvement [1], [2]. Since then, several scene representations have been proposed to reduce the memory usage of these methods while retaining the training and inference advantages [17], [18]. ZipNeRF [2] learns pre-filtered features to address z-aliasing by weighting features at different scales based on the distance from the ray origin. Recently, 3D Gaussian Splatting (3DGS) methods [19]–[21] have emerged as a competitor to NeRF-based approaches. Nevertheless, these techniques require prohibitively long training times for real-time free-viewpoint video applications.

Dynamic view synthesis. NeRF has also been extended to non-rigid scenes by deforming the observed points to a canonical space or over time [5], [8]. D-NeRF [22] proposed deforming points along a ray at a given time to obtain the corresponding points in a canonical scene. NSFF [23] models the scene as a time-variant continuous function encoded in an MLP; however, this approach is limited by the length of the videos it can encode. DynIBaR [24] proposes to use motion trajectory fields to achieve temporal consistency on video recordings. Nevertheless, these methods are incapable of generalizing to future time instants or unseen scenes.

Generalizable view synthesis. Generalizable methods are based on multi-view stereo techniques, where the target view is estimated given a limited number of views. We are particularly interested in those that, without a lengthy optimization of the scene, enable real-time view synthesis [10], [25]. One of the pioneering approaches is MVSNet [26], which constructs a cost volume of 2D image features to estimate the depth of the target view. However, these methods do not scale well at high resolutions. Coarse-to-fine methods [10], [27] have emerged to scale better to higher resolutions by estimating depth at increasing resolutions. LatentSplat [28] introduces variational 3D Gaussians for efficient and generalizable 3D reconstruction from few input views. However, its performance degrades when reconstructing scenes at high resolution. ENeRF [10]

achieves real-time high-resolution new view synthesis generalization employing a coarse-to-fine scheme. However, ENeRF is unable to synthesize novel views from sparse inputs, as it requires a dense camera setting. Our model differs from previous approaches in that we leverage generalizable methods and depth information to create a consistent probabilistic feature volume, enabling volumetric rendering. Additionally, we model unseen or occluded information in the target view with a generative prior, filling unseen content with coherent regions based on contextual views.

III. Methodology

Coarse-to-fine rendering schemes enable real-time rendering; however, current approaches struggle to synthesize realistic views given a sparse multi-view setting in a generalizable manner. Figure 2 provides a general overview of our proposed GDNeRF. Our approach leverages depth information to generate probabilistic feature volumes (Section III-A) from the limited scene information. Previous methods particularly encounter difficulties with missing information in the target view due to occluded content in the source views. We address this issue with a generative 3D CNN (Section III-B) based on StyleGAN [11], which introduces new feature information to generate feasible features for neural rendering. Finally, the renderer conducts volumetric rendering with a small set of samples on each ray, enabling generalizable real-time rendering. It is important to note that optimization is only performed during network pretraining. The new view synthesis of an unseen scene is accomplished through a single forward pass given the sparse input views.

A. Probabilistic feature volume

Given a set of M sparse source views $s = (s_0, s_1, ..., s_M)$, the objective is to synthesize a new view t that appears realistic and is coherent in geometry and texture with the context of the available sparse views. We begin by extracting information from the source views s to create a probabilistic feature volume F. If H and W denote the target image resolution and D defines the depth resolution, we create a 3D feature volume $F \in \mathbb{R}^{H \times W \times D}$ relying on depth maps d_i of the source views. Visual features are extracted from the source sparse views, resulting in depth maps f_i, which are then projected from a viewpoint into grid cells. The feature at position $p \in \mathbb{R}^3$ in each cell $F(p)$ is weighted proportionally to its likelihood of being projected into the correct position. We model this probability with a Gaussian distribution:

$$w_i^p = N(p - c_i; d(p)_i, \sigma(p)) \cdot \gamma \tag{1}$$

where $d(p)_i$ is the distance given by the depth map d_i when p is projected into the i-th image plane, c_i is the i-th camera center, and γ is a scaling factor. Thus, features where the cell position is closer to the estimated depth on camera i will be more represented at that position. The standard deviation $\sigma(p)$ is greater on depth discontinuities where the estimation might contain more errors. For more details on the probabilistic feature volume, please see Appendix I.

B. Generative rendering volume

When addressing sparse new view synthesis, it is crucial to incorporate some kind of generative prior to infer missing information based on the contextual source images. To incorporate generative information into our framework, we adopt the generator from StyleGAN [11] and the discriminator from StyleGANv2 [29], which have been studied in-depth [30], [31]. Specifically, the generator G is a 3D CNN that processes the probabilistic feature volume F_i as described in Section III-A and produces multiple outputs at different resolutions, which are then combined as detailed in Appendix II. These multilevel render volumes are denoted as $R = (R_0, R_1, ..., R_l)$, where l is the number of resolution levels. We employ a 3D CNN with a UNet-like architecture as the generator, incorporating AdaIN [32] layers to embed the generative prior. We employ the Non-Saturating GAN objective [33], as denoted in Eq. 2 and Eq. 3. The generator should generate feasible and realistic images given the contextual source sparse views s, which can be formulated as follows:

$$\mathcal{L}_{GAN}^{G} = -E_{(t,s),w\sim p_d, MLP_W(x)}[log(D(G(s,w)|s))] \tag{2}$$

where p_d is the data distribution and x, w and MLP_W denote a latent code, a style code and the mapping network as in StyleGAN, respectively. x is a random variable drawn from a standard Gaussian distribution. To simplify notation, we encapsulate the generation of the render volume and the volumetric rendering with $G(\cdot)$. Therefore, $G(s, w)$ results in a rendered image that is conditioned on source views s and a latent code w.

We denote p_g as the distribution of images generated by the volumetric renderer using the multilevel render volume generated by the generator. The discriminator, besides distinguishing between samples from p_d and p_g, should analyze if the image is coherent given the context of source images s. This assists G in generating not only realistic samples but also views that are coherent in geometry and texture with the available views. Therefore, the objective of D is to optimize the following function:

$$\begin{aligned}\mathcal{L}_{GAN}^{D} = &-E_{\hat{t},s,w\sim p_g,p_d,MLP_W(x)}[log(1 - D(\hat{t}|s))] \\ &- E_{(t,s)\sim p_d}[log(D(t|s))]\end{aligned} \tag{3}$$

where $\hat{t}$ is an image rendered with the volumetric renderer. Additionally, we experiment with incorporating the synthesized depth information and the groundtruth depth as an extra supervision term, aiming to ensure that the expected density depth is similar to the groundtruth depth images. These variations are denoted as L_{DGAN}^{G} and L_{DGAN}^{D}.

C. Coarse-to-fine generative loss

Similar to previous work, we use an L2 penalization on the differences in color with the groundtruth $\mathcal{L}_c$. To enhance the network's ability to capture important textural information in greater detail, we incorporate a perceptual loss $\mathcal{L}_p$, which compares feature maps extracted from a pretrained VGG16

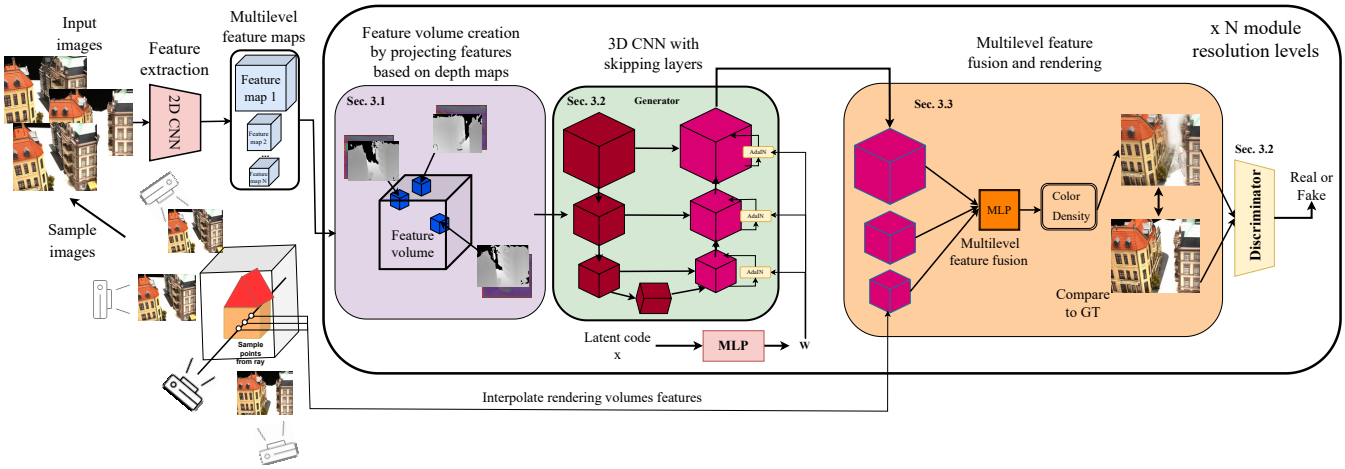

Fig. 2: **GDNeRF approach.** From sparse few views, our method extracts a set of feature maps which are projected into our probabilistic feature volume based on the source views depth maps. Our generator module processes the probabilistic feature volume to generate a feasible multilevel feature volume. These volumes contain features that the volumetric renderer uses to synthesize the target view in real time, without the need of per-scene training.

network on ImageNet. Since we have access to the groundtruth depth information during training, we supervise the predicted depth map by the NeRF MLP with an L1 loss $\mathcal{L}_d$. The objective at a resolution level r is to minimize the sum of all the previously described losses:

$$\mathcal{L}_r = \lambda_c \mathcal{L}_c + \lambda_d \mathcal{L}_d + \lambda_p \mathcal{L}_p + \lambda_g (\mathcal{L}^D_{DGAN} + \mathcal{L}^G_{DGAN}) \quad (4)$$

where λ_c, λ_d, λ_{dreg}, λ_p and λ_g are hyperparameters weighting each loss term. We omit the subscript r in the loss terms for clarity. The total loss will be the sum of the losses $\mathcal{L}_r$ at the Q cascade levels of the coarse-to-fine approach.

IV. Experiments

We conduct both qualitative and quantitative evaluations, along with comparisons against competing approaches. Our quantitative evaluation involves measuring the PSNR, SSIM [34] and LPIPS [35]. First, in Section IV-A, we describe the details of each dataset and why they have been chosen. Then, in Section IV-B, we perform extensive evaluation and conduct an ablation study to assess the impact of each component of GDNeRF. For implementation details and hyperparameters settings, please see Appendix III.

A. Datasets

For multiview static scenes, we pretrained our GDNeRF model using the DTU dataset [37], following the evaluation protocol of previous work. Our model follows the training and evaluation protocol from MVSNeRF [36]. The model is pretrained on a set of scenes that are different from those used for evaluation. In this setup, 16 views were used as input for generating new scene views, while 4 were reserved for evaluating the rendered results. In order to evaluate the DTU dataset in a sparser setting, we perform an additional experiment where we filter source views with a cosine similarity greater than 0.8 relative to the target view. While we primarily utilized the DTU dataset for comparison with prior work, it is worth noting that this dataset features a dense camera setting, which does not align with the specific problem addressed by our method. This is why, in order to evaluate DTU in a sparser setting, we perform an additional experiment where we filter source views with a cosine similarity greater than 0.8 relative to the target view.

Methods	Camera setting	FPS	DTU PSNR↑	DTU SSIM↑	DTU LPIPS↓
PixelNeRF [3]	Dense	0.019	19.31	0.789	0.382
IBRNet [9]		0.217	26.04	0.917	0.191
MVSNeRF [36]		0.416	26.63	0.931	0.168
ENeRF [10]		25.29	**27.61**	**0.956**	**0.091**
Ours		15.75	**26.82**	**0.922**	**0.099**
ENeRF [10]	Sparse	25.29	**16.598**	**0.523**	**0.489**
Ours		15.75	**22.485**	**0.869**	**0.167**

TABLE I: **View synthesis comparison.** Comparison among various generalizable methods on DTU in the dense and sparse camera setting. Despite our method not being specialized for dense camera settings, it ranks as the second best performer in this scenario. However, in the sparse setting, it significantly outperforms the current state-of-the-art. Best result. Second best result.

In order to assess the generalization capabilities in a sparse camera setting, we turned to the CWI [12] and ActorsHQ [13] datasets. The CWI dataset comprises 45 sequences recorded by 7 Kinect cameras arranged in a sparse configuration, forming a 360° camera setup where each camera is positioned 60° apart about the Y-axis. Finetuning of the model was performed on this sequence using the first 200 frames, with evaluation conducted on the subsequent 100 frames. Random cameras were sampled as target views, and the 3 closest views were

Methods	DTU			CWI			ActorsHQ		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
ENeRF	16.598	0.523	0.489	14.523	0.471	0.466	21.046	0.807	0.177
GDNeRF	22.153	0.825	0.236	18.504	0.665	0.427	26.445	0.900	0.116
GDNeRF-Multi	21.420	0.816	0.233	18.328	**0.666**	0.433	**26.568**	0.901	0.112
GDNeRF-GANs	**22.485**	**0.869**	**0.167**	**18.897**	0.648	**0.381**	25.446	**0.911**	**0.097**

TABLE II: **GDNeRF ablation study.** Results on DTU, CWI and ActorsHQ for the different components of GDNeRF and ENeRF. Our model significantly outperforms ENeRF. Even the most basic version of GDNeRF, utilizing only the probabilistic feature volume, already surpasses its performance.

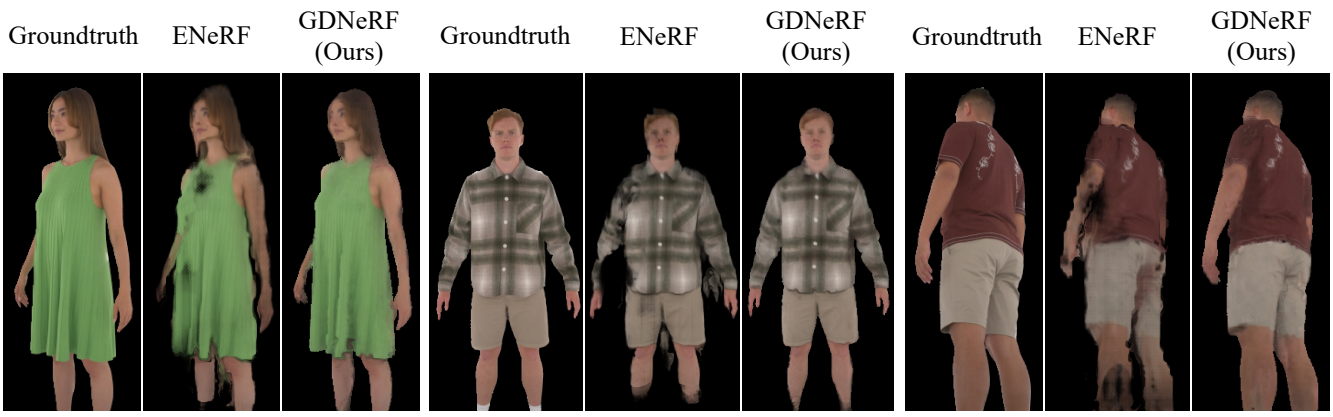

Fig. 3: **Generalizable sparse new view synthesis in ActorsHQ.** Comparison between ENeRF and GDNeRF in ActorsHQ. ENeRF generates many artifacts and holes compared to our GDNeRF.

utilized to render the target. The ActorsHQ dataset provides high-quality recordings of clothed humans in motion, featuring multi-view captures from 160 cameras arranged in a 360° setup, along with accurate depth maps. Due to the denser camera setting of this dataset, we filtered source views with a cosine similarity greater than 0 relative to the target view. This filtering process resulted in sparser and more distant cameras relative to the target view. We leave the *Actor08* sequence for evaluation while pretraining in the rest.

B. Experimental validation

We begin by comparing GDNeRF to previous methods using their dense camera evaluation settings to validate our approach against prior work. Next, we conduct an ablation study of the various components of our approach and analyze their performance in a generalization setting. Finally, we extensively compare our full approach against prior state-of-the-art.

1) Comparison with other generalizable approaches: In order to compare to previous generalizable NeRF approaches, we use the DTU dataset. In Table I, we provide the quantitative results for the dense and sparse camera settings in DTU. As we can observe, in the dense camera setting, GDNeRF still yields good results despite being designed for sparse camera settings. In the dense evaluation setting, as ENeRF is an image-based renderer, it can reuse many pixels from very close views, which is why it gives better results than our approach. However, when filtering the closest views that have a cosine similarity greater than 0.8 with respect to the target view, we can observe that our method outperforms the current state-of-the-art, which is ENeRF.

2) Generalization ablation study: Our GDNeRF is mainly compared to ENeRF which is the current state-of-the-art for generalizable NeRFs. Unlike ENeRF, we employ a combination of IBR (Image-based rendering) and MBR (Model-based rendering), utilizing our probabilistic feature volume III-A and generative model III-B. ENeRF relies solely on an IBR approach, which exhibits limitations, particularly in synthesizing occluded content and leveraging a sparse camera setting, which is what GDNeRF mainly aims to solve. In Table II, we provide an ablation study assessing the generalization capabilities on the DTU, CWI and ActorsHQ datasets. We refer to GDNeRF with the multilevel feature volume representation as GDNeRF-Multi and to GDNeRF with both the multilevel representation and generative module as GDNeRF-GANs. The results demonstrate that the base model, which only incorporates the probabilistic feature volume (section III-A), outperforms ENeRF significantly. Furthermore, the integration of multiple render volumes with ZipNeRF's rendering enhances some results in both the CWI and ActorsHQ datasets. Incorporating the generative rendering volume further improves the overall performance.

3) Sparse view synthesis experiments: Our model consistently outperforms ENeRF in sparse setting scenarios, particularly on CWI. This difference is especially pronounced in

the CWI dataset, where the closest cameras are 60° apart along the Y-axis. Qualitatively, this difference is evident in Figure 1. While ENeRF struggles to render the person adequately due to the sparse input views, GDNeRF produces high-quality renders. Figure 3 depicts the qualitative results on the ActorsHQ dataset. As can be seen, GDNeRF outperforms ENeRF, exhibiting greater consistency, higher quality and far less artifacts. For additional qualitative results, see Appendix IV. Finally, for a comparison with offline sparse-camera optimization methods, see Appendix V.

V. Conclusions

In this work, we have presented a generalizable new view synthesis approach for sparse camera environments. We address sparse camera environments that are not common to NeRF-based approaches by leveraging depth with a probabilistic feature volume and filling unseen regions with a generative prior. We demonstrate competitive performance on the dense camera setting while outperforming the state-of-the-art in the sparse setting by a large margin. This advancement paves the way for real-time new view synthesis applications across unseen scenes or time instants, such as free-viewpoint video rendering. Moreover, our analysis delves into the impact of camera sparsity using the CWI and ActorsHQ datasets. This comprehensive examination underscores the efficacy of our proposed method in handling sparse camera environments for generalizable view synthesis.

In future work, we aim to investigate the impact of synthetic depth information on our method, particularly for datasets lacking groundtruth depth information. This analysis could provide insights into the robustness and generalization capabilities of our approach in scenarios where depth data is not available. Moreover, to enhance rendering performance, our method could integrate Gaussian Splatting [19] as its rendering method. However, this adaptation would involve working with point clouds instead of 3D feature volumes, presenting both technical and computational challenges that require further exploration.

References

[1] Müller et al., "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1–15, 2022.

[2] Barron et al., "Zip-nerf: Anti-aliased grid-based neural radiance fields," *arXiv preprint arXiv:2304.06706*, 2023.

[3] Yu et al., "pixelnerf: Neural radiance fields from one or few images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4578–4587.

[4] Xu et al., "4k4d: Real-time 4d view synthesis at 4k resolution," *arXiv preprint arXiv:2310.11448*, 2023.

[5] Song et al., "Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 5, pp. 2732–2742, 2023.

[6] Kocabas et al., "Hugs: Human gaussian splats," *arXiv preprint arXiv:2311.17910*, 2023.

[7] Weng et al., "Humannerf: Free-viewpoint rendering of moving people from monocular video," in *Proceedings of the IEEE/CVF conference on computer vision and pattern Recognition*, 2022, pp. 16210–16220.

[8] Park et al., "Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields," *arXiv:2106.13228*, 2021.

[9] Wang et al., "Ibrnet: Learning multi-view image-based rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4690–4699.

[10] Lin et al., "Efficient neural radiance fields for interactive free-viewpoint video," in *SIGGRAPH Asia Conference Proceedings*, 2022.

[11] Karras et al., "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[12] Reimat et al., "Cwipc-sxr: Point cloud dynamic human dataset for social xr," in *Proceedings of the 12th ACM Multimedia Systems Conference*, 2021, pp. 300–306.

[13] Işık et al., "Humanrf: High-fidelity neural radiance fields for humans in motion," *arXiv preprint arXiv:2305.06356*, 2023.

[14] Levoy et al., "Light field rendering," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pp. 441–452. 2023.

[15] Ji et al., "Deep view morphing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2155–2163.

[16] Mildenhall et al., "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[17] Chan et al., "Efficient geometry-aware 3d generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16123–16133.

[18] Chen et al., "Tensorf: Tensorial radiance fields," in *European Conference on Computer Vision*. Springer, 2022, pp. 333–350.

[19] Kerbl et al., "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, 2023.

[20] Yu et. al, "Mip-splatting: Alias-free 3d gaussian splatting," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 19447–19456.

[21] Huang et. al, "2d gaussian splatting for geometrically accurate radiance fields," in *ACM SIGGRAPH 2024 conference papers*, 2024, pp. 1–11.

[22] Pumarola et al., "D-nerf: Neural radiance fields for dynamic scenes," in *CVPR*, 2021.

[23] Li et al., "Neural scene flow fields for space-time view synthesis of dynamic scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6498–6508.

[24] Li et al., "Dynibar: Neural dynamic image-based rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4273–4284.

[25] Chibane et al., "Stereo radiance fields (srf): Learning view synthesis for sparse views of novel scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[26] Yao et al., "Mvsnet: Depth inference for unstructured multi-view stereo," in *Proceedings of the ECCV*, 2018, pp. 767–783.

[27] Yu et al., "Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement," in *Proceedings of the IEEE/CVF CVPR*, 2020, pp. 1949–1958.

[28] Wewer et al., "latentsplat: Autoencoding variational gaussians for fast generalizable 3d reconstruction," in *European Conference on Computer Vision*. Springer, 2024, pp. 456–473.

[29] Karras et al., "Analyzing and improving the image quality of StyleGAN," in *Proc. CVPR*, 2020.

[30] Zhou et. al, "Point-stylegan: Multi-scale point cloud synthesis with style modulation," *Computer Aided Geometric Design*, vol. 111, pp. 102309, 2024.

[31] Pehlivan et. al, "Styleres: Transforming the residuals for real image editing with stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 1828–1837.

[32] Chen et al., "On self modulation for generative adversarial networks," *arXiv preprint arXiv:1810.01365*, 2018.

[33] Goodfellow et al., "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[34] Wang et al., "Image quality assessment: from error visibility to structural similarity," *IEEE TIP*, vol. 13, no. 4, pp. 600–612, 2004.

[35] Zhang et al., "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[36] Chen et al., "Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14124–14133.

[37] Jensen et al., "Large scale multi-view stereopsis evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 406–413.