FEW TENSORF: ENHANCE THE FEW-SHOT ON TENSORIAL RADIANCE FIELDS

Thanh-Hai Le
School of Computer Science and Engineering
The Saigon International University
Ho Chi Minh City 700000, Vietnam
lethanhhai@siu.edu.vn

Hoang-Hau Tran
Department of Informatic Technology
FPT University
Ho Chi Minh City 700000, Vietnam
hauth151239@fpt.edu.com

Trong-Nghia Vu
Department of Informatic Technology
FPT University
Ho Chi Minh City 700000, Vietnam
nghiavt150934@fpt.edu.com

March 27, 2026

ABSTRACT

This paper presents Few TensoRF, a 3D reconstruction framework that combines TensorRF's efficient tensor based representation with FreeNeRF's frequency driven few shot regularization. Using TensorRF to significantly accelerate rendering speed and introducing frequency and occlusion masks, the method improves stability and reconstruction quality under sparse input views. Experiments on the Synthesis NeRF benchmark show that Few TensoRF method improves the average PSNR from 21.45 dB (TensorRF) to 23.70 dB, with the fine tuned version reaching 24.52 dB, while maintaining TensorRF's fast $\approx 10 - 15$ minute training time. Experiments on the THuman 2.0 dataset further demonstrate competitive performance in human body reconstruction, achieving 27.37-34.00 dB with only eight input images. These results highlight Few TensoRF as an efficient and data effective solution for real-time 3D reconstruction across diverse scenes.

***K*eywords** 3D Reconstruction · Few-shot Improvement · FreeNeRF · NERF · Real-time Rendering · TensorRF

1 Introduction

The ability to create accurate and versatile 3D reconstructions has become increasingly vital in today's technological landscape. From applications in computer vision to healthcare, and even in the world of entertainment, 3D reconstruction plays a pivotal role. The history of 3D reconstruction methods has witnessed significant advancements, with a continuous pursuit of techniques that can transform single or multiple 2D images into detailed 3D representations. This study, inspired by the success of the renowned Neural Radiance Field (NeRF) [1, 2] for synthesizing novel views of complex scenes, aims to incorporate two improved methods related to NeRF: the Tensorial Radiance Field (TensorRF) [3] and the enhancement of few-shot neural rendering with Free Frequency Regularization (FreeNeRF) [4]. TensoRF introduces a novel approach using 4D tensors and decompositions [5, 6], for improved rendering quality, reduced memory usage, and faster reconstruction times compared to NeRF. On the other hand, FreeNeRF simplifies few-shot neural rendering by introducing costeffective regularization terms, outperforming complex alternatives with a single line of code change, and emphasizing the significance of frequency in NeRF's training. This study is motivated by the pressing need for a 3D reconstruction method that is both versatile and efficient. Our primary objective is to develop a versatile 3D reconstruction method, leveraging the combined capabilities of TensorRF and FreeNeRF. This approach is designed to address the shortcomings of previous methods and promote broader applications in various domains, especially experience on human body datasets. In developing our approach, we build upon the foundation of TensoRF

and introduce two primary regularization techniques, frequency, and occlusion, inspired by the FreeNeRF paper. These techniques are incorporated as a positional encoding step to map each input 5D coordinate within the tensor component of the scene. Our methodology will be thoroughly compared with previous NeRF-related methods, not only on typical objects but also on the more complex task of 3D human body reconstruction. The THuman2.0 dataset [7] will be used and trained on different setups for experiments and comparison.

2 Related works

2.1 Neural Radiance Fields (NeRF)

Neural Radiance Fields (NeRF) [1] has transformed the approach to 3D reconstruction. Utilizing fully connected deep networks, NeRF processes single continuous 5D coordinates, including spatial location *(x, y, z)* and viewing direction (θ, ϕ) to generate volume density and view-dependent emitted radiance. These properties facilitate the synthesis of novel views by querying 5D coordinates along camera rays and utilizing volume rendering techniques. NeRF's applications range from novel view synthesis [2], 3D generation [8, 9] to deformation [10, 11] and video [12, 13]. Despite its remarkable capabilities in high-quality scene representation, NeRF's significant drawback is its demand for a substantial number of input images, rendering it ineffective at synthesizing novel views from a limited set of input views, such as 3, 6, or 9 views [4]. This limitation not only hinders its potential real-world applications but also contributes to its relatively slow rendering speed, with NeRF requiring approximately 35 hours of training for completeness [3].

2.2 Tensorial Radiance Field (TensorRF)

Tensorial Radiance Fields (TensorRF) [3] presents a novel approach to 3D reconstruction by representing the radiance field as a 4D tensor. TensorRF utilizes the tensor decomposition technique [14] to achieve improved rendering quality, reduced memory usage, and faster reconstruction times compared to NeRF. Although the experience and comparison between the classic CP decomposition [15] and a new vector matrix (VM) decomposition, TensorRF focuses mainly on the reconstruction through gradient descent rather than the decomposition itself. This introduces low-rank regularization in the optimization process, ultimately leading to enhanced rendering quality. TensorRF's advantage lies in its potential to make 3D reconstruction more memory-efficient while maintaining rendering quality.

2.3 FreeNeRF: Few-shot Neural Rendering

Few-shot neural rendering has posed a challenge, often requiring complex external information or pretraining. FreeNeRF [4] addresses this challenge by introducing cost-effective regularization terms, notably frequency and occlusion regularization. Remarkably, these modifications can be incorporated with minimal changes to the original NeRF code, maintaining the same computational efficiency. This approach underscores the importance of geometry in few-shot neural rendering, achieving impressive results without expensive pre-training or complex training-time patch rendering.

2.4 Human Body 3D Reconstruction

Reconstructing the 3D Human Body is the focus of our latest experiment, aimed at expanding the capabilities of NeRF-like methods beyond their traditional applications on standard datasets like Blender, DTU [14, 16], or LLFF [17]. Unlike the objects found in these datasets, the challenges posed by a comprehensive Human Body dataset such as Thuman 2.0 [7] are distinct, ranging from the diverse human shapes and clothing variations to the multitude of poses encountered in real-world scenarios. Our innovative approach, rooted in NeRF-like principles, prioritize generating novel views over constructing the 3D mesh or voxel itself— a departure from earlier methods like ICON [18], ECON [19] and others. This unique perspective holds promise, particularly in scenarios with limited resources, offering a viable solution to address the complexities inherent in capturing the richness of the human form.

3 Method

The proposed Few-TensoRF is still a 3D reconstruction method, capable of exporting 3D object meshes due to its special 3D point coordinate and RGB with density mapping. However, it functions as just one core process within the broader NeRF application-level pipeline, as depicted in Fig. 1. The Few TensoRF method mainly takes advantage of the strong point in 2 methods: faster training time with TensoRF [3] and improvement of the FreeNeRF [4] in sparse input cases.

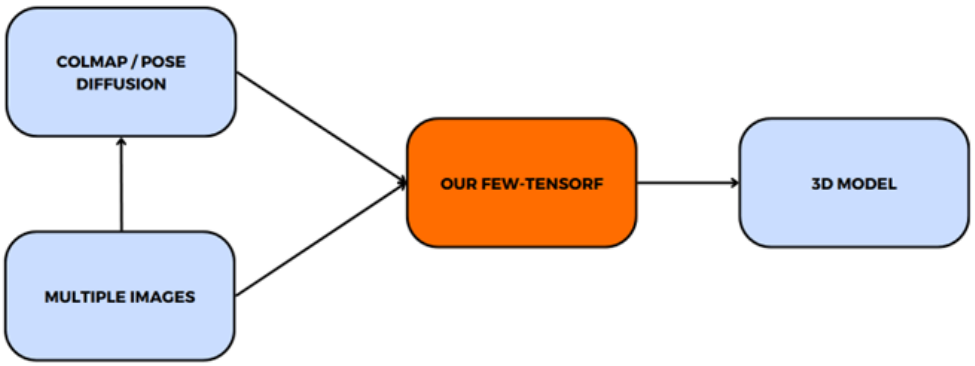

Figure 1: **Few TensoRF in the 3D NeRF-like Pipeline.** The diagram shows where Few TensoRF fits into the core of the 3D NeRF-like reconstruction pipeline. It all begins with multiple scene images, and either COLMAP or Pose Diffusion calculates precise camera details. These ones, along with the images, train Few TensoRF. The trained Few TensoRF model then takes center stage in the final steps, ensuring a streamlined and accurate 3D model reconstruction within the NeRF-like framework. This visual guide underscores the seamless integration of Few TensoRF at the heart of the 3D NeRF-like process.

3.1 Fast training time with Tensorial Radiance Field (TensoRF) Base

TensoRF builds upon the NeRF framework, introducing crucial improvements in the form of a neural network architecture. While both TensoRF and NeRF share the overarching goal of mapping any 3D location x and viewing direction d to their respective volume density σ and, view-dependent color c, TensoRF distinguishes itself by adopting a unique approach. In contrast to NeRF, which relies solely on Multi-Layer Perceptrons (MLPs), TensoRF models the radiance field of a scene as a 4D tensor. This innovative methodology uses a regular 3D grid G with per-voxel multi-channel features to represent the scene's radiance field. In simple terms, TensoRF separately models the volume density σ and view-dependent color c by two distinct grids: a geometry grid G and an appearance grid G_c. The continuous grid-based radiance field is expressed as:

$$\sigma, c = G_\sigma(x), S(G_c(x), d) \tag{1}$$

Where $G(x)$ and $G_c(x)$ represent the trilinear interpolated features from the two grids at location x. Both $G(x)$ and $G_c(x)$ are modeled as factorized tensors, employing the VM decomposition method. This relationship can be expressed using the following formulas (or as shown in Fig. 2):

$$G_\sigma(x) = \sum_{r=1}^{R_\sigma} \sum_{m \in XYZ} A_{\sigma,r}^m \tag{2}$$

$$G_c(x) = \sum_{r=1}^{R_o} A_{c,r}^X \circ b_{3r-2} + A_{c,r}^Y \circ b_{3r-2} + A_{c,r}^Z \circ b_{3r} \tag{3}$$

As detailed in the original TensoRF paper [3], the function S (in Eqn. 1) is introduced as a pre-selected function. This function, which can be a small MLP or a spherical harmonics (SH) function, is employed to convert an appearance feature vector and a viewing direction d into the final color c. However, in this study, we will take advantage of the variant of MLP structures to get more customization for few-shot scenarios. Those A and A_c tensor components play a pivotal role in the revolution of our Few TensorRF technique.

3.2 Enhance Few-shot on Tensorial Radiance Field (Few TensoRF)

Given the findings depicted in Figure 3, showcasing the rendered novel views with varying amounts of training images in the TensoRF baseline, it becomes evident that TensoRF encounters challenges in delivering satisfactory results under sparse input conditions. The visual comparison highlights a noticeable disparity in the quality of novel views, particularly as the number of training images decreases. As discussed in the FreeNeRF [4], we anticipate encountering

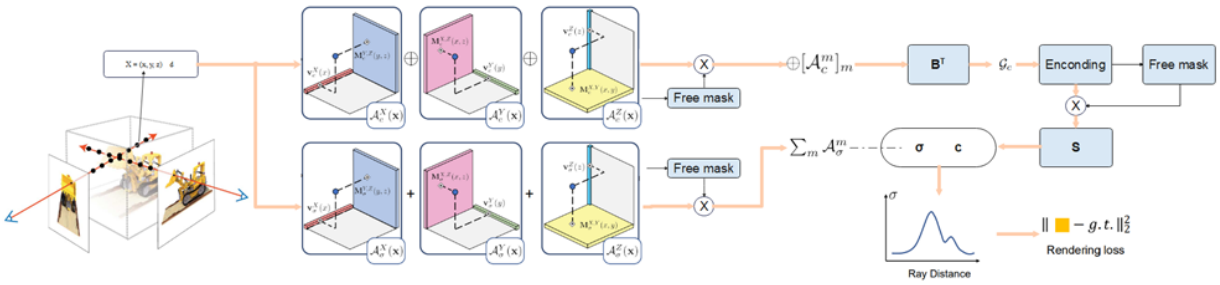

Figure 2: **The Few TensoRF overview.** This illustration was adapted from the original TensoRF paper [3]. It highlights two significant enhancements as we move from left to right: incorporating density frequency masks and integrating appearance color frequency on each rendered voxel. Additionally, a supplementary frequency mask is applied to refine the positional encoding fed into the neural networks.

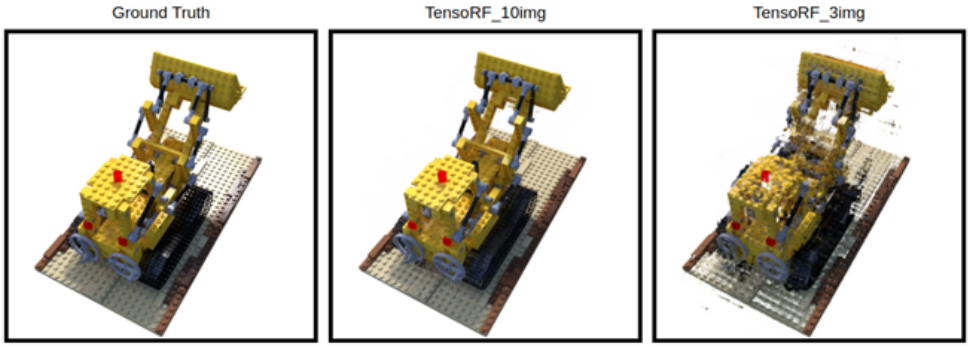

Figure 3: **Comparing the rendered novel views quality of TensoRF baseline across different setups.** To assess TensoRF's performance on sparse input scenarios, we retrain the default TensoRF setup on different amounts of training image inputs. Following the image order, we have the original testing image (Ground Truth), and the novel view images rendered with TensoRF after training with 10 and 3 images (TensoRF_10img and TensorRF_10img) respectively.

issues arising from over-fast convergence during the optimization process. This rapid convergence, particularly in high-frequency components, hinders TensorRF's ability to effectively explore low frequency information. Consequently, it introduces a quite bias in TensoRF, leading to the manifestation of undesired high-frequency artifacts (TensoRF 3img in Fig. 3).

Therefore, to achieve better results in sparse input cases, the proposed Few TensoRF method proposes three improvements on the TensoRF [3] baseline, as depicted in Figure 2. Frequency Masking Tensor Components. The goal here is to decrease the sensitivity of each tensor component in the high-frequency domain during the initial stages of training. This approach guides our model to concentrate on enhancing low-frequency structures, ensuring the stability of tensor components against undesired high-frequency artifacts. As mentioned earlier, there are two types of components: A for the density component and Ac for the appearance color feature component. Two frequency masks will be applied to them according to the following equations:

$$A_L^{'}(t,T;\mathbf{x}) = A_L(\mathbf{x}) \odot \alpha(t,T,L) \tag{4}$$

with

$$\alpha_i(t,T,L) = \begin{cases} 1 & \text{if} \quad i \leq \frac{t \cdot L}{T} \\ \frac{t \cdot L}{T} - \lfloor \frac{t \cdot L}{T} \rfloor & \text{if} \quad \frac{t \cdot L}{T} + 3 < i \leq \frac{t \cdot L}{T} + 6 \\ 0 & \text{if} \quad i > \frac{t \cdot L}{T} + 6 \end{cases}$$

Where $\alpha_i(t,T,L)$ denotes the i^{th} bit value of $\alpha(t,T,L)$; t and T are the current training iteration and the total iterations, respectively, L corresponds to either the density component length or the appearance feature length, depending on the desired tensor component to mask. This dynamic frequency mask, a unique feature introduced in FreeNeRF [4],

calculates the mask values based on the current iteration if t is less than T. The calculation involves both an integer and a fractional part, ensuring a smooth transition. If t is greater than or equal to T, the method returns a tensor of ones. This dynamic frequency mask is particularly beneficial for enhancing the high-frequency details of rendering novel views in the later stages of training. Additionally, we have implemented a simplified version of Eqn. 4, allowing us to set a fixed visible ratio for the frequency mask. This can be expressed as follows:

$$Freq\_mask[: int(L * v\_ratio)] = 1 \tag{5}$$

Frequency Masking Appearance Grid G_c. Similar to the process of masking tensor components, we once again apply the frequency mask, this time to the entire appearance grid G_c, along with the viewing direction d as described in Fig. 4. The masking technique employed here is analogous to the frequency masking tensor component, intended to mitigate overfitting in the MLP neural network associated with high-frequency signals. This frequency regularization acts as a filter for the positional encoding of the appearance grid G_c and view direction d (akin to Eqn. 4). Subsequently, the output is fed into a streamlined neural network, or MLP, responsible for predicting the RGB value of the input point.

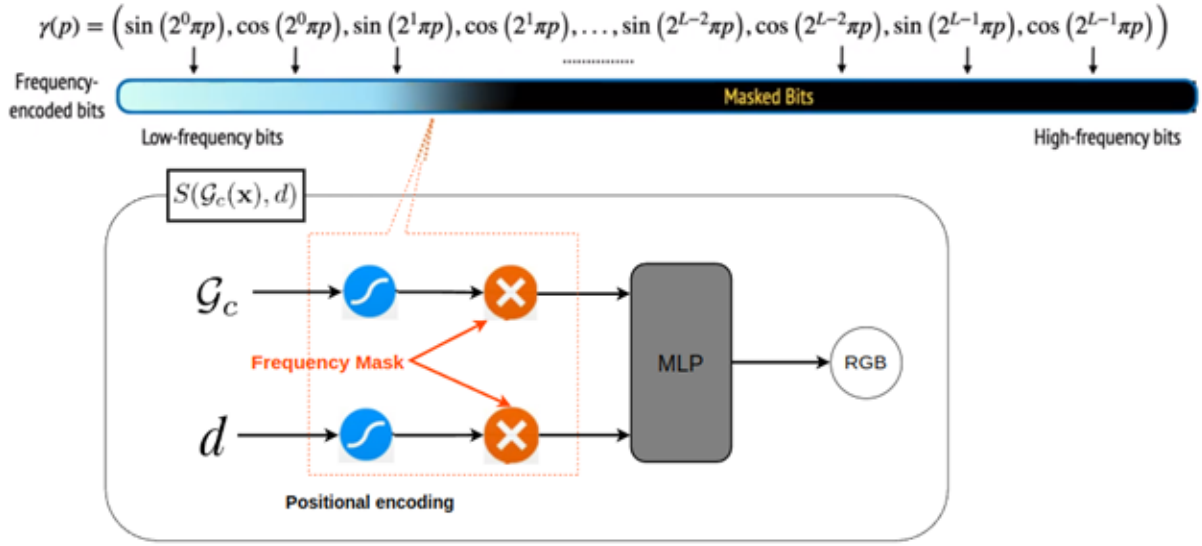

Figure 4: **Diagram illustrating Frequency Masking on the appearance grid G_c and view direction d.**

Occlusion Regularization. Despite efforts like frequency regularization, characteristic artifacts, such as "walls" or "floaters," persist in novel views during few-shot neural rendering. Therefore, the novel "occlusion" regularization is introduced to push the density of floaters in the near-camera region to zeros, and models will learn to explain this area in a farther place. This occlusion loss is directly incorporated into the original TensoRF-based loss, originally designed for overfitting but now tailored to be more adaptable in few-shot scenarios, reducing overfitting tendencies. This integration aims to enhance the model's robustness, providing a more balanced performance in handling sparse input cases.

4 Results and Discussion

4.1 Experiment setup

Dataset and Metric. We utilized two datasets with distinct purposes: Synthesis NeRF and THuman 2.0. For performance evaluation, we replicate the few-shot Synthesis NeRF dataset following the methodology outlined in FreeNeRF. THuman 2.0 is utilized by the Blender Nerf add-on to extract 2D images and corresponding camera information, primarily serving new experiments. We will use PSNR as the core metric to evaluate the qualitative performance of the methods.

Implementation. To ensure fair comparisons among different methods, consistent experimental environments were maintained for each training session. All experiment methods were implemented or reproduced with a PyTorch-based framework. The four main methods explored in this study include the reproduced few-shot state-of-the-art FreeNeRF, the original TensoRF on a few-shot dataset setup, and two variations of the proposed Few TensoRF, facilitating performance comparisons with the former two. Fundamentally, our models were trained with a ray batch size of 1024

over 15,000 iterations. Optimization was performed using the Adam optimizer with a specific learning rate, and training was conducted on Google Colab with a T4 Tesla GPU.

4.2 Comparison

Synthesis NeRF Comparison. As shown in Table 1, the Our Few TensorRF demonstrates significant improvements in average PSNR compared to both FreeNeRF (50k iters, repro.) and TensorRF (repro.). The finetuned version of Our Few TensorRF further enhances performance, achieving the highest PSNR in several scenes. Notably, the raw version of Our Few TensorRF consistently outperforms TensoRF (repo.), underscoring its efficacy in few-shot scenarios and showcasing advancements introduced by Our Few TensorRF. Fine-tuning Our Few TensorRF leads to additional improvements, surpassing both FreeNeRF and TensorRF in almost all scenes. These results underscore the effectiveness of our proposed Few TensorRF methods in achieving superior rendering quality, particularly in challenging few-shot conditions. However, there is a noticeable challenge in the Drums scene. This anomaly may be attributed to the intricate details and complexity of the Drum scene, making it the most challenging in the Synthesis NeRF dataset. In scenarios with numerous hidden angles, our model may struggle to comprehend the intricacies, leading to the reconstruction of some peculiar artifacts.

Table 1: PSNR metrics comparison on each Synthesis NeRF's scene. The best-performing and second-best results from our experiments are denoted by red and blue highlights, respectively.

PSNR	**Lego**	**Chair**	**Drums**	**Ficus**	**Mic**	**Ship**	**Materials**	**Hotdog**	**Avg.**
FreeNeRF_50k iters (repro.)	24.81	27.01	20.46	20.59	25.54	23.79	21.69	29.40	24.16
Tensorf (repro.)	23.83	23.45	18.32	19.34	23.3	19.96	20.78	21.85	21.45
Our Few Tensorf	25.68	27.70	18.88	21.09	24.56	21.64	22.04	27.99	23.70
Our Few Tensorf (Fine tune)	25.73	27.71	18.04	21.36	27.44	23.77	22.36	29.71	24.52

Table 2: PSNR metrics comparison on each Synthesis NeRF's scene. The best-performing and second-best results from our experiments are denoted by red and yellow highlights, respectively.

Metrics	**PSNR↑**	**Training time↓**
FreeNeRF_50k (repro.)	29.40	04:55:00
FreeNeRF_15k (repro.)	28.02	01:25:40
TensoRF (repro.)	21.85	00:15:00
Our Few Tensorf	27.99	00:15:20
Our Few Tensorf (Fine tune)	29.71	00:10:22

To assess the training efficiency among methods, we adjusted FreeNeRF iterations to 15000 to establish a common reference. Lower training times and higher PSNR values are preferable. After numerous experiments, we observed minimal differences in training times across scenes, typically varying by only 2-3 minutes for each method. Therefore, in Table 2, we opted to reproduce FreeNeRF 15000 iterations on HotDog alone to save time and Fig. 5 shows the result of four methods for Hotdog object. Additionally, as anticipated, FreeNeRF 15000 iterations did not optimize well enough to match the performance of FreeNeRF_50k. Our fine-tuned model demonstrated superior efficiency and rendering quality compared to other methods. In summary, Our Few TensorRF outshines FreeNeRF and TensorRF in average PSNR, with finetuning enhancing performance across scenes. Despite the intricacies of the challenging Drum scene, the proposed model demonstrates superior efficiency and rendering quality. The findings highlight the effectiveness of our Few TensorRF methods in addressing few-shot challenges, making it a promising approach for highquality 3D reconstruction.

Performance on THuman 2.0 dataset. Figure 6 presents a comparison between two objects, 0300 and 0525, from the THuman0.2 dataset. For this analysis, models were trained on either 50 or 8 images using TensoRF, while our Few TensoRF was implemented with a visible frequency ratio of 0.8. All referenced models utilized VM decomposition and were trained for an identical number of iterations (15,000).

Table 3 offers a detailed summary of the performance of three methods on the THuman 2.0 dataset, with particular emphasis on objects 0525 and 0300. As expected, the original TensoRF models—trained with a larger number of input images—exhibited superior image rendering performance, providing high accuracy and intricate detail attributable to increased data availability. To assess few-shot learning capability, we reduced the number of training views to eight, paralleling the methodology used in the Synthesis Nerf experiment. The results indicate that the Few TensoRF models show diminished performance relative to the original TensoRF.

Figure 5: **Comparing the view rendered of Hotdog object between methods.**

The comparative 3D meshes displayed in Fig. 7 further corroborate the PSNR score findings reported in Table 3. Training TensoRF with 50 images enabled accurate reconstruction of object details such as clothing, facial features, fingers, and toes. Conversely, models trained with fewer images generated meshes with numerous holes and exhibited instability during object rendering. Notably, the Few TensoRF model trained on only eight images produced increased noise around the object compared to the original TensoRF model trained with more extensive data.

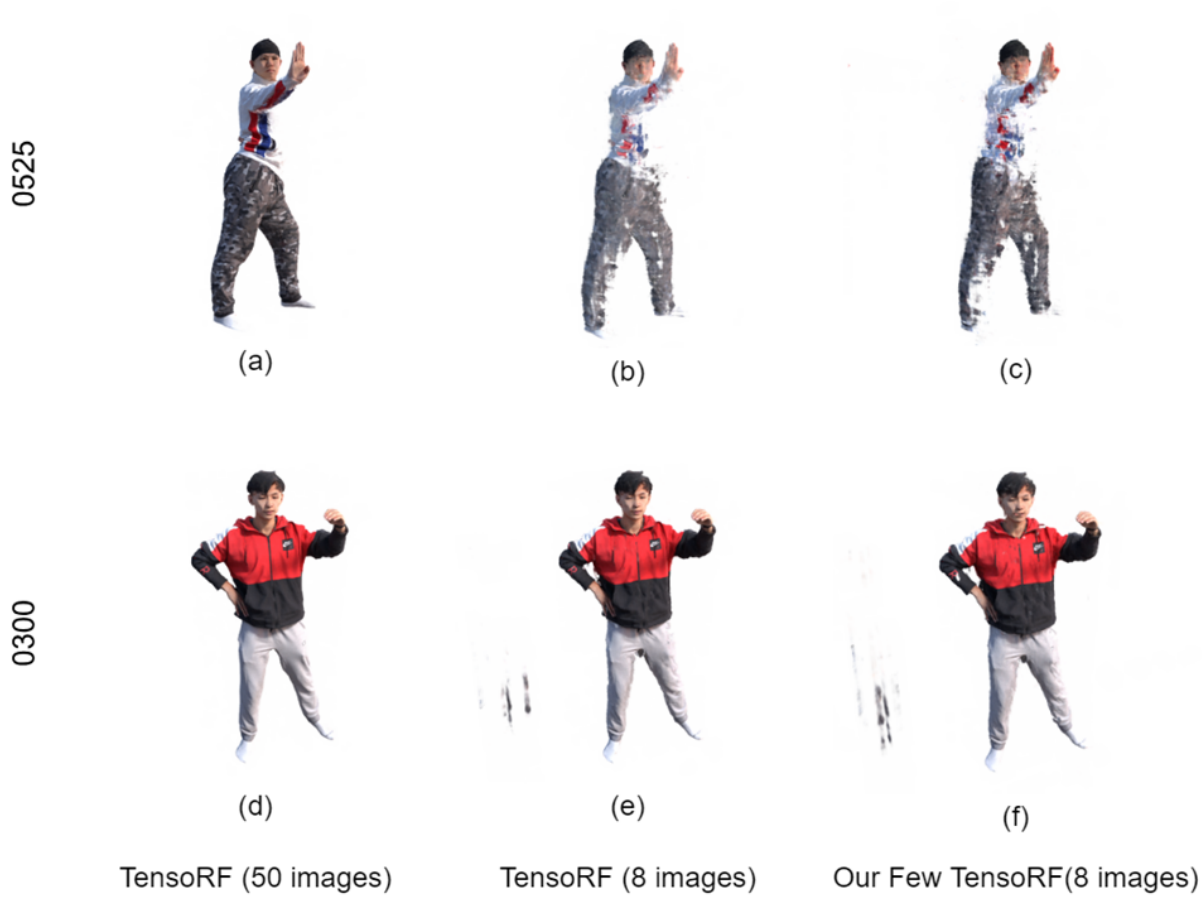

Figure 6: **Comparing the quality of rendered images with different numbers of images between TensoRF and Few TensoRF.**

Table 3: Comparison of PSNR parameters between TensoRF and Few TensoRF with different numbers of input images.

PSNR	**0525**	**0300**
TensoRF (50 images)	40.98	45.58
TensoRF(8 images)	28,37	34.28
Few Tensorf (8 images)	27.37	34.00

Furthermore, Fig. 8 illustrates the underlying reasons for the superior PSNR scores achieved by the proposed methods in these scenes. While the FreeNeRF method tends to blur fine details within the scene, our approach provides more precise refinement across nearly all scenarios evaluated in Synthesis NeRF.

4.3 Discussion

The results provide a positive outlook on the potential development of our method in future applications. The commendable performance on the Synthesis NeRF dataset positions our approach favorably among prior studies in novel view synthesis, establishing a benchmark for subsequent work on the THuman 2.0 dataset. However, despite the strengths exhibited, certain limitations are evident. Challenges emerged in specific scenes, notably the "Drums" scene in Synthesis NeRF, where our method encountered difficulties. The exact nature of this issue remains elusive, possibly attributable to the intricate details of the "Drums" scene.

Interestingly, experiments on the novel THuman 2.0 dataset yielded compelling results, showcasing higher PSNR metrics compared to standard NeRF datasets used in prior studies. Yet, the rendered images exhibited notable noise, highlighting an area for improvement in future endeavors. This underscores the need for a more robust approach to enhance the quality of 3D reconstruction meshes, particularly in the context of the THuman 2.0 dataset. Our method's distinctive strength lies in its ability to achieve efficiency across three pivotal scenarios: fast training time, utilization of fewer images, and maintaining an acceptable level of general quality. This emphasis on efficiency aligns with our primary goal of contributing to the advancement of NeRF-like techniques. However, evaluating the overall performance of Few TensoRF on THuman 2.0 dataset is challenging due to the limited experimentation on only two out of 500 objects in the dataset. Within the scope of this study, we focused on experimenting with our method on specific THuman 2.0 objects instead of training on the entire dataset for a comprehensive benchmark.

Looking ahead, our focus on THuman 2.0 opens avenues for refining methods and addressing the identified limitations. In a broader context, NeRF-like methods, including our Few-TensoRF, hold promise not only for reconstructing 3D models but also for excelling in novel view synthesis—a crucial technique poised to revolutionize the fields of virtual reality (VR) and augmented reality (AR) applications.

5 Conclusion

The proposed Few-TensoRF presents a step forward in the realm of novel view synthesis, showing promise across various scenarios. By addressing challenges and limitations, our approach stands as a valuable asset for real-world applications with resource constraints. The promising results achieved on both Synthesis NeRF and THuman 2.0 datasets, coupled with a focus on overcoming identified limitations, underscore the potential impact and versatility of our Few-TensoRF method in advancing the capabilities of 3D reconstruction and view synthesis technologies.

In conclusion, the Few TensoRF method represents a significant advance in 3D reconstruction, achieving faster training times and improved efficiency. Integrating FreeNeRF and TensoRF capabilities, the proposed Few TensoRF demonstrates versatility across diverse datasets, contributing to the broader field of 3D reconstruction and promising advancements in applications requiring faster training times and dynamic scene captures.

Declaration of conflicting interest

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

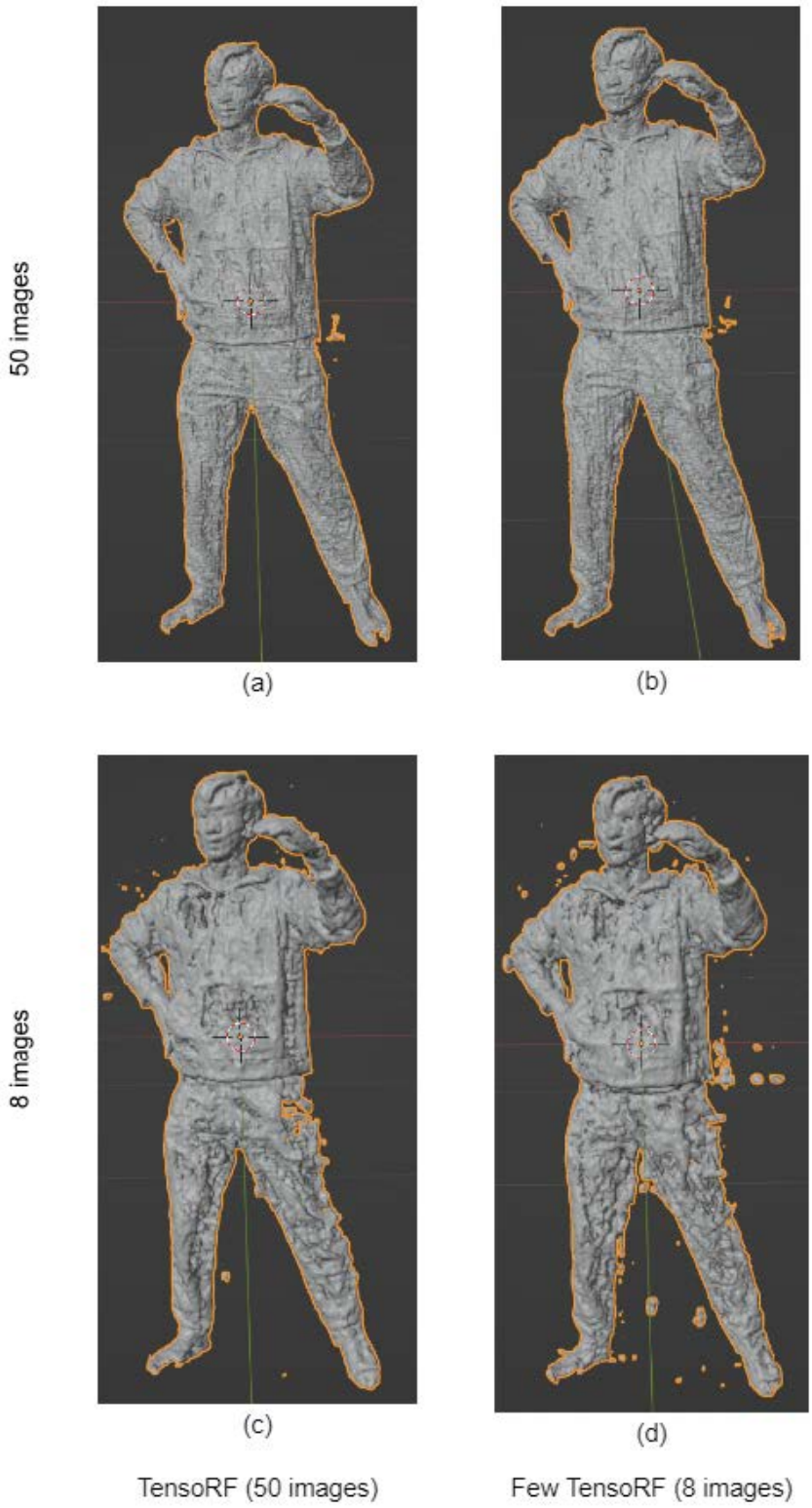

Figure 7: **Comparing 3D meshes rendered for 0300 object.**

Data availability statement

Data supporting the findings of this article are publicly available at `https://www.kaggle.com/datasets/nguyenhung1903/nerf-synthetic-dataset` and `https://github.com/ytrock/THuman2.0-Dataset`.

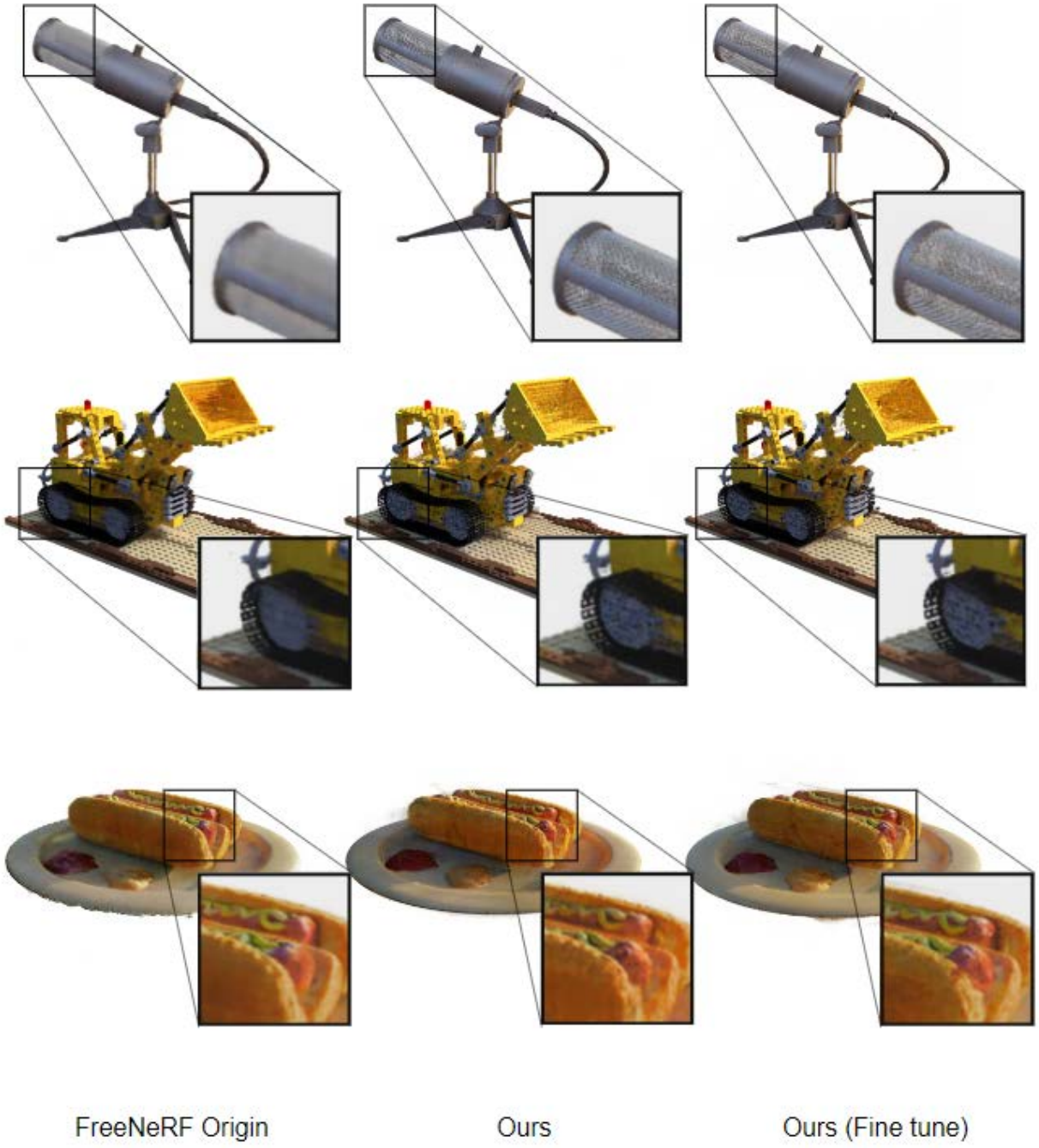

Figure 8: **Visual comparison between methods.**

ORCID iD

Thanh-Hai Le: https://orcid.org/0000-0002-3212-3940

References

[1] B. Mildenhall, P.P. Srinivasan, M. Tancik, J.T. Barron, R. R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[2] R. Martin-Brualla, N. Radwan, M.S.M. Sajjadi, J.T. Barron, A. Dosovitskiy, and Duckworth D. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *CVPR*, 2021.

[3] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su. Tensorf: Tensorial radiance fields. In *ECCV*, 2022.

[4] J. Yang, M. Pavone, and Y. Wang. Freenerf: Improving few-shot neural rendering with free frequency regularization. In *CVPR*, 2023.

[5] J.D. Carroll and J.J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35:283–319, 1970.

[6] L. De Lathauwer. Decompositions of a higherorder tensor in block terms—part ii: Definitions and uniqueness. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1033–1066, 2008.

[7] T. Yu, Z. Zheng, K. Guo, P. Liu, Q. Dai, and Y. Liu. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *CVPR*, 2021.

[8] B. Poole, A. Jain, J.T. Barron, and B. Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprints arXiv: 2209.14988*, 2022. Accessed July 05, 2023.

[9] A. Jain, B. Mildenhall, J.T. Barron, P. Abbeel, and B. Poole. Zero-shot text-guided object generation with dream fields. In *ECCV*, 2022.

[10] K. Park, U. Sinha, J.T. Barron, and et al. Nerfies: Deformable neural radiance fields. In *ECCV*, 2022.

[11] A. Pumarola, E. Corona, G. Pons-Moll, , and F. Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. *arXiv preprints arXiv: 2011.13961*, 2020. Accessed June 10, 2023.

[12] Z. Li, S. Niklaus, N. Snavely, and O. Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. *arXiv preprints arXiv: 2011.13084*, 2020. Accessed June 10, 2023.

[13] W. Xian, J.-B Huang, J. Kopf, and C. Kim. Space-time neural irradiance fields for freeviewpoint video. *arXiv preprints arXiv: 2011.12950*, 2020. Accessed June 5, 2023.

[14] H. Aanæs, R. Jensen, G. Vogiatzis, E. Tola, and A. Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120(2):153–168, 2016.

[15] T.G. Kolda and B.W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[16] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanaes. Large scale multi-view stereopsis evaluation. In *CVPR*, 2014.

[17] B. Mildenhall, P. Srinivasan, R. Ortiz-Cayon, and et al. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. In *the ACM SIGGRAPH*, 2019.

[18] Y. Xiu, J. Yang, D. Tzionas, and M.J. Black. Icon: Implicit clothed humans obtained from normals. In *CVPR*.

[19] Y. Xiu, J. Yang, X. Cao, D. Tzionas, and M.J. Black. Econ: Explicit clothed humans optimized via normal integration. In *CVPR*, 2023.