DiscoRF : Discriminator on TensoRF

Byungwoo Jeon
Dept. of Computer Science & Engineering
Korea University
Seoul, South Korea
ipcs@korea.ac.kr

Seoyeon Byeon
Dept. of Computer Science & Engineering
Korea University
Seoul, South Korea
byunhwan8832@korea.ac.kr

Suhyeok Kim
Dept. of Computer Science & Engineering
Korea University
Seoul, South Korea
rlatn219@korea.ac.kr

Seonghu Jeon
Dept. of Computer Science & Engineering
Korea University
Seoul, South Korea
jsh0423@korea.ac.kr

Hoongyu Chung
Dept. of Electrical Engineering
Korea University
Seoul, South Korea
kttx3267@korea.ac.kr

***Abstract*—We propose a novel approach to enhance TensoRF, a radiance field model, utilizing a Generative Adversarial Network (GAN) architecture. Our approach positions TensoRF as the generator and employs a shallow CNN as the discriminator within the GAN structure. This configuration, which interprets GAN loss as a regularization mechanism, significantly enhances the model's performance with fewer steps. Our experimental results underscore the efficiency of this method in generating high-quality 3D synthetic images.**

***Index Terms*—radiance field model, TensoRF, GAN, 3D synthesis**

I. Introduction

In this paper, we present a novel approach to enhance the performance of TensoRF, a model for radiance fields, by incorporating a Generative Adversarial Network (GAN) architecture. Recent advancements have leveraged radiance fields for the synthesis of new viewpoints in a single scene, overcoming their high computational cost. Unlike NeRF, which solely utilizes MLPs, TensoRF models the radiance field of a scene as a 4D tensor, providing superior rendering quality with significantly less memory usage through Vector-Matrix (VM) decomposition.

In our research, we propose a GAN architecture where TensoRF serves as the generator, creating synthetic views of the scene, which are then distinguished by a discriminator based on a shallow CNN. We utilize the grid sampling technique and append a shallow CNN as a discriminator with scheduling details, enables TensoRF to achieve superior performance with approximately one-tenth of the steps. We propose that the GAN loss serves as a regularization for the original model for radiance fields, contributing to the enhanced performance.

Our experimental results demonstrate that a simple classifier, tasked with distinguishing whether the synthesized view is new or not, allows the generator to achieve high-quality PSNR with fewer steps. This research contributes to the field by enhancing the performance of TensoRF, paving the way for more efficient and high-quality 3D synthetic image generation.

II. Related Works

A. Scene Representations and Radiance Fields

Various ways to represent 3D scenes, including meshes, point clouds, volumes, and implicit functions have been studied quickly these days. NeRF [1] proposed a neural radiance field to synthesize novel views of real-world scenes from posed 2D images with high fidelity. These representations are widely used and applied in diverse graphics and vision tasks. To achieve realistic and real-time novel-view synthesis, NeRF with pure MLP-based representation is too slow for scene reconstruction and view rendering. Many recent methods [2]–[4] are proposed to improve rendering speed by leveraging a voxel grid of features in the radiance field. However, these voxel grid-based rendering still has limitations on low reconstruction speed and high memory costs. TensoRF [5], the tensorial radiance fields, leverages tensor factorization techniques to resolve those issues, and it improved reconstruction speed and memory costs. We applied tensorial radiance fields as generator and shallow CNN as discriminator to achieve high fidelity of reconstruction.

B. Tensor Factorization

Tensor decomposition has been studied for diverse applications in vision, graphics, and machine learning. The most widely used tensor decomposition is Tucker decomposition [6] and CP decomposition [7], [8]. Both methods are considered matrix singular value decomposition(SVD). By combining Tucker and CP decomposition, block term decomposition(BTD) has been proposed and utilized in many vision applications. Inspired by BTD decomposition, a new VM(Vector-Matrix) decomposition [5] has been proposed and contributed to more efficient radiance field reconstruction. In this work, we leveraged VM decomposition which factorizes a tensor into multiple vectors and matrices, reducing memory complexity. More detailed information can be found in section III-B.

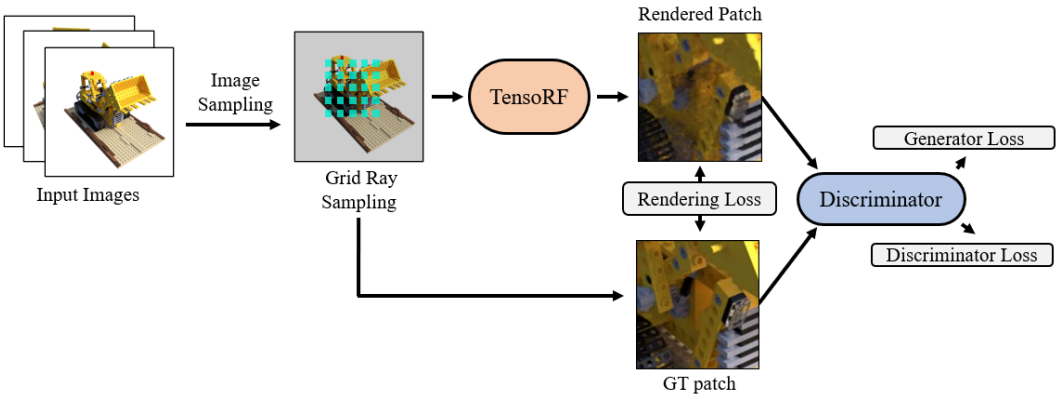

Fig. 1. **Method Overview.**

C. Ray Sampling

Our model employs a patch-based discriminator. When using patches instead of individual pixels, it is important to sample patches at random scales. Ray sampling is the method employed for this purpose. This process enables the use of a convolutional discriminator independent of the image's resolution. This approach is similar to the one used in GRAF [9], but it differs. It is a method that allows for the continuous evaluation of radiance fields. However, the grid structure should be maintained after the sampling in our tensorial model. Further details are explained in section III-D.

III. Method

In the following, we first briefly review the Tensorial Radiance Field (TensoRF) as the generator for our model.

A. Tensorial Radiance Field Representation

The Tensorial Radiance Field (TensoRF) is a novel scene representation that models the 5D plenoptic function as a 4D tensor. This tensor is factorized into a set of 3D spatial tensors and a 1D feature tensor. The spatial tensors capture the spatial structure of the scene, while the feature tensor encodes the appearance information. The 4D tensor is represented as:

$$T = \sum_{r=1}^{R} (v_{\sigma,r} \otimes v_{x,r} \otimes v_{y,r} \otimes v_{z,r}) \otimes M_r \tag{1}$$

where $v_{\sigma,r}$, $v_{x,r}$, $v_{y,r}$, and $v_{z,r}$ are the vector components, M_r is the matrix component, and R is the rank of the tensor.

B. Vector-Matrix (VM) Decomposition

The Vector-Matrix (VM) decomposition is a novel tensor decomposition technique introduced in the TensoRF framework. It decomposes the 4D tensor into a set of 3D spatial vectors and a set of 2D feature matrices. This decomposition leads to better reconstruction quality and faster optimization speed compared to classical tensor factorization techniques.The VM decomposition is represented as:

$$\begin{aligned} T = &\sum_{r=1}^{R} (v_{\sigma,r} \otimes v_{x,r} \otimes v_{y,r} \otimes v_{z,r}) \\ &\otimes (M_{\sigma,r} \otimes M_{x,r} \otimes M_{y,r} \otimes M_{z,r}) \end{aligned} \tag{2}$$

where $v_{\sigma,r}$, $v_{x,r}$, $v_{y,r}$, and $v_{z,r}$ are the vector components, $M_{\sigma,r}$, $M_{x,r}$, $M_{y,r}$, and $M_{z,r}$ are the matrix components, and R is the rank of the tensor.

C. Volume Rendering and Reconstruction

The volume rendering process in TensoRF involves integrating the radiance along each ray to produce the final image. Using the tensorial radiance field representation, the radiance at each point in the scene is computed by:

$$\sigma(x) = \sum_{r} \sum_{m} A^{m}_{\sigma,r}(x) \tag{3}$$

$$c(x,d) = S(B \oplus [A^{m}_{c,r}(x)]_{m,r}, d) \tag{4}$$

where σ represents the volume density at location x, c represents the view-dependent color at location x with viewing direction d. Based on VM decomposition described in sections III-B, $A_{\sigma,r}(x)$ and $A_{c,r}(x)$, S, B can be derived. Short description of the notice is as follows:

- $A_{\sigma,r}(x)$ and $A_{c,r}(x)$ are the interpolated values of the component tensors for the density and color grids, respectively.
- S is the shading function that converts an appearance feature vector and a viewing direction to color.
- B is a global appearance dictionary that abstracts the appearance commonalities across the entire scene.

To compute the color of each pixel, we traverse along a ray and sample Q shading points along the ray. The pixel color is then computed by:

$$\begin{aligned} C &= \sum_{q=1}^{Q} T_q \cdot c_q, \\ T_q &= \tau_q(1 - \exp(-\sigma_q \cdot \Delta_q)), \\ \tau_q &= \exp(-\sum_{p=1}^{q-1} \sigma_p \Delta_p) \end{aligned} \tag{5}$$

where T_q is the transmittance at the q-th shading point, Δ_q is the ray step size.τ_q represents transmittance.

The reconstruction process involves optimizing the tensor factors to minimize the difference between the rendered image

and the ground truth image. This is achieved using a combination of L2 rendering loss and additional regularization terms.

D. Adversarial Training

We employ a Generative Adversarial Network (GAN) framework. In this setup, the TensoRF model serves as the generator, while a separate model, a shallow CNN-based discriminator, is trained to distinguish between the images generated by the TensoRF model and real images.

The training process involves two main steps: training the discriminator and training the generator.

- **Discriminator**: The discriminator is trained to distinguish between real images and images generated by the TensoRF model. The loss function for the discriminator, L_D, is given by:

$$L_D = -\Big(y\log(D(G)) + (1-y)\log(1-D(\hat{C}))\Big) \quad (6)$$

  where G is the ground truth image, $\hat{C}$ is the rendered image determined in section III-C.
- **Generator**: The loss function for the generator, L_G, is given by:

$$L_G = -\log(D(\hat{C})) \quad (7)$$

Also, we utilize the grid ray sampling method. In the original TensoRF, rays were uniformly sampled from NHW candidates, where N indicates a number of images, and $H \times W$ represents the pixel size of each image. However, to provide patches as input for the discriminator, we modified the sampling process to maintain the grid structure between sampled points. Firstly, we sample one image uniformly from all input images, and then we randomly sample a patch from the image.

- **Grid Ray Sampling**: A $K \times K$ patch coordinates, denoted as $\nu(\mathbf{u}, s)$, is subset of 2D image coordinates. It can be defined as follows:

$$\nu(\mathbf{u}, s) = \{(sx+u, sy+v) | x, y \in \{0, 1, ..., K\}\} \quad (8)$$

  where $\mathbf{u}$ represents the top-left point of the image and s indicates the number of unselected points between selected adjacent points. The point $\mathbf{u}$ is selected such that all points in the grid exist within the 2D coordinates of $H \times W$.

Finally, We sample (1) rays and camera direction for the TensoRF input and (2) GT patch image for the discriminator input from the patch coordinates. This sampling method is similar to the one used in GRAF. Still, it differs in that it uses discrete integer coordinates and does not use bilinear interpolation when passing patches as input to the GAN. We exclude the ray-filtering process in our implementations.

Thus, our objective function is modified by:

$$\min_{\theta}\max_{\phi}\Big[\mathbb{E}_{I\sim p_D}[\mathbb{E}_{\nu\sim p_\nu}[f(D_\phi(\mathbf{P}(I,\nu)))] - \mathbb{E}_{\nu\sim p_\nu}[f(D_\phi(G_\theta(\nu)))] + \lambda\mathbb{E}_{I\sim p_D}[\mathbb{E}_{\nu\sim p_\nu}[\|\nabla_{\mathbf{P}(I,\nu)}D_\phi(\mathbf{P}(I,\nu))\|_2^2]]\Big] \quad (9)$$

where $f(t) = -\log(1+\exp(-t))$, λ controls the strength of the regularizer, D_ϕ is the discriminator, G_θ is the generator, $\mathbf{P}(I,\nu)$ is the sampled patch, I is an image from the data distribution p_D, ν is the distribution over random patches.

E. Training and Inference

As discussed in section III-C, we follow previous experimental results for optimizing the radiance field reconstruction, while adding a GAN Loss. Our general expression for the loss function is:

$$L = L_{\text{img}} + \omega \cdot L_{\text{reg}} + L_G \quad (10)$$

$$L_{\text{img}} = ||G - \hat{C}||^2 \quad (11)$$

where L_{img} represents the L2 rendering loss, L_{reg} denotes the regularization terms, and L_G is the generator loss.

The choice of regularization term L_{reg} depends on the specific dataset and the conditions under which the images were captured. For example, for datasets with very few input images or imperfect capture conditions, a TV loss was found to be more efficient than the L1 sparsity loss. The L1 sparsity loss was found to be effective in improving the quality in extrapolating views and removing floaters/outliers in final renderings. The choice of regularization term is thus a trade-off between the desired quality of the reconstruction and the characteristics of the dataset. The L1 sparsity loss is only applied on the density parameters and is expressed as:

$$L_{\text{reg}} = \frac{1}{N_{\text{param}}}\left(\sum_{r=1}^{R_\sigma}\|M_{\sigma,r}\|_1 + \|v_{\sigma,r}\|_1\right) \quad (12)$$

where $|M_{\sigma,r}|_1$ and $|v_{\sigma,r}|_1$ are simply the sum of absolute values of all elements, and N_{param} is the total number of parameters. In our experiments, L1 sparsity loss is used for the Synthetic NeRF dataset with a $\omega = 0.0004$.

The TV loss is expressed as:

$$L_{\text{reg}} = \frac{1}{N_{\text{param}}}\left(\sum_{r=1}^{R_\sigma}\|\nabla^2 M_{\sigma,r}\|_1 + 0.1\|\nabla^2 v_{\sigma,r}\|_1\right) \quad (13)$$

where ∇^2 is the squared difference between the neighboring values in the matrix/vector factors.

IV. Experiments

A. Baseline.

We evaluate our model on Lego, Flower datasets on Synthetic NeRF dataset, and Truck datasets on Tanks&Temples dataset in our experiments. At the same time, We compare our approach with TensoRF and TensoRF with grid sampling. We use TensoRF-VM as generator and Tiny CNN as discriminator and optimize each model for 50k steps with a batch size of 1024. Fig. 2 shows the rendering PSNRs for each generator iterations. Our approach achieves qualitative renderings with appearance and geometry details, as shown in Fig. 3.

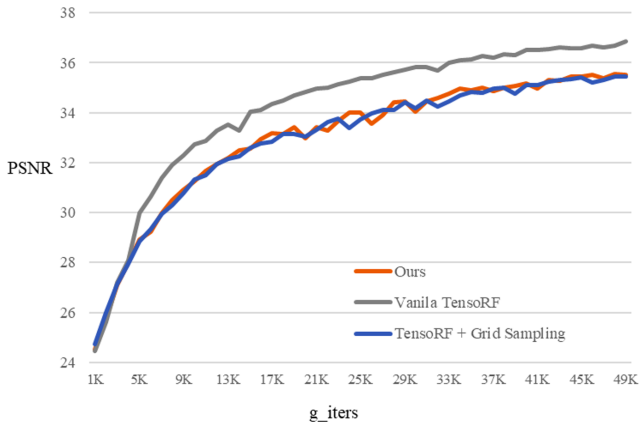

Fig. 2. We compare our method with TensoRF and TensoRF with grid sampling. We report PSNR for each generator iterations for each models.

V. Conclusion

Limitations. Training GAN poses significant challenges and inconveniences. The potential for training disruptions increases as the model goes beyond a specific iteration threshold. To ensure stable GAN training, our model necessitates comprehensive efforts. Moreover, there are pressing need for further research to enhance the speed of GAN training.

Tensor4D [10] exhibited exceptional performance in dynamic scene modeling. The robustness of our GAN architecture in learning dynamic scenes remains as future works.

Contributions. Previous studies [9], [11] have explored models that employ NeRF as a generator. However, these models predominantly concentrate on GAN-based camera pose optimization. In contrast, we understand TensoRF as a generator of GAN and tried to show that it can achieve outstanding performance by distinguishing whether the output is real or fake. If the GAN training process stabilized, we expect that the discriminator's feedback during training will play a crucial role in regularization by imposing penalties on the TensoRF when generating unrealistic samples. This mechanism effectively guides the generator toward capturing meaningful features and prevents it from merely memorization of the training data.

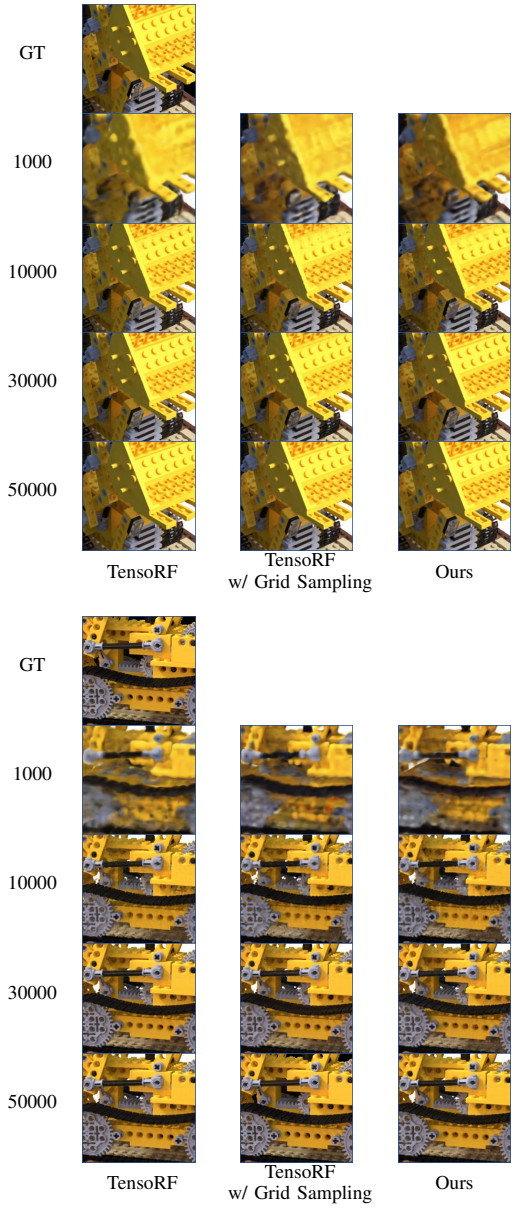

Fig. 3. Qualitative results per iteration of our model and comparison methods on lego datasets.

We introduced a neat architecture that improves the performance of Radiance Fields. By employing grid sampling and leveraging the GAN architecture, we elevated the learning capability of TensoRF. As a result, we open up the possibilities of radiance fields with GAN architecture.

References

[1] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision. pp. 405–421. Springer (2020)

[2] Yu, A., Li, R., Tancik, M., Li, H., Ng, R., Kanazawa, A.: Plenoctrees for real-time rendering of neural radiance fields. arXiv preprint arXiv:2103.14024 (2021)
[3] Hedman, P., Srinivasan, P.P., Mildenhall, B., Barron, J.T., Debevec, P.: Baking neural radiance fields for real-time view synthesis. arXiv preprint arXiv:2103.14645 (2021)
[4] Liu, L., Gu, J., Lin, K.Z., Chua, T.S., Theobalt, C.: Neural sparse voxel fields. NeurIPS (2020)
[5] A. Chen, Z. Xu, A. Geiger, J. Yu and H. Su. TensoRF: Tensorial Radiance Fields. In Proc. of the European Conf. on Computer Vision (ECCV), 2022.
[6] Tucker, L.R.: Some mathematical notes on three-mode factor analysis. Psychometrika 31(3), 279–311 (1966)
[7] Carroll, J.D., Chang, J.J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. Psychometrika 35(3), 283–319 (1970)
[8] Harshman, R.A.: Foundations of the parafac procedure: Models and conditions for an" explanatory" multimodal factor analysis (1970)
[9] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger.: Graf: Generative radiance fields for 3d-aware image synthesis. In Proc. NeurIPS, 2020.
[10] Shao. R., Zheng, Z., Tu, H., Liu, B., Zhang, H., and Liu, Y.: Tensor4D : Efficient Neural 4D Decomposition for High-fidelity Dynamic Reconstruction and Rendering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2023.
[11] Eric R. C., Marco M., Petr K., Jiajun W., and Gordon W.: pi-GAN: Periodic Implicit Generative Adversarial Networks for 3D-Aware Image Synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.