Exploring Hyperparameters and Training Strategies in Plenoxels: Radiance Fields without Neural Networks

Sumuditha Kulasekara
department of Computer Science and Engineering
University of Moratuwa
Sri Lanka

***Abstract*—Plenoxels offer an efficient approach to radiance field reconstruction by replacing neural networks with directly optimized sparse voxel grids. Although the method achieves competitive quality, its performance is highly sensitive to hyperparameter settings, loss design, and training strategies. In this work, I explore the impact of key hyperparameters—including spherical harmonics degree, learning rate schedules, and total variation regularization—on reconstruction quality and training stability. I also investigate the initialization of weighted metrics, the role of progressive regularization, and the potential for transfer learning. Through a series of controlled experiments, I provide an empirical analysis of how these factors influence both convergence behavior and perceptual fidelity, even in cases where quantitative accuracy does not consistently improve. This study offers insights into the strengths and limitations of Plenoxels, highlighting directions for future research in radiance field reconstruction without neural networks.**

I. Introduction

Neural radiance fields (NeRFs) have revolutionized 3D scene representation by enabling photorealistic novel view synthesis from a sparse set of images. However, traditional NeRF models rely heavily on deep neural networks, which require extensive computational resources and long training times. Plenoxels, proposed by Yu et al. [1], provide an alternative approach by representing 3D scenes using sparse voxel grids with spherical harmonic (SH) or learned basis functions, effectively eliminating the need for deep networks while maintaining competitive rendering quality.

Despite the efficiency of Plenoxels, the quality of reconstructions remains sensitive to hyperparameters such as TV regularization weights, leaarning rates, and the SH order of the basis functions. Prior work has largely relied on fixed hyperparameter configurations, which may not generalize across different datasets or scene types. Furthermore, the choice of SH order directly influences the ability of the model to capture high-frequency lighting and geometric details, but increasing the order can lead to memory and computational bottlenecks.

In this work, I explore the impact of hyperparameter tuning and spherical harmonics (SH) order adjustments on Plenoxels. I systematically examine adaptive learning rates, weight initialization, and progressive regularization, alongside variations in SH order, to assess their effects on reconstruction quality. Although the accuracy improvements in this study were limited, the experiments highlight important trade-offs between reconstruction fidelity, computational efficiency, and hyperparameter sensitivity. By documenting these observations, I aim to offer practical guidance for future research and applications of Plenoxel-based radiance field representations.

II. Related Work

A. NeRF and Voxel-Based Approaches

The introduction of Neural Radiance Fields (NeRF) marked a major advance in novel view synthesis, showing that continuous volumetric scene representations could be learned directly from posed 2D images using multilayer perceptrons (MLPs) [2]. NeRF quickly became a benchmark for photorealistic rendering and geometry recovery, but its reliance on dense neural architectures makes training computationally expensive and inference slow, often requiring hours of optimization per scene. To alleviate these challenges, voxel-based extensions have been explored. Neural Sparse Voxel Fields (NSVF) [3] combined voxel grids with neural feature interpolation, significantly reducing both memory usage and training time. PlenOctrees [4] advanced this direction further by baking a trained NeRF into an octree data structure, enabling interactive rendering speeds. These methods highlight a broader trend in radiance field research: shifting away from pure neural models toward hybrid or voxel-centric representations that balance efficiency with quality.

B. Previous Improvements in Plenoxels

Plenoxels [1] took this shift to its extreme by removing neural networks altogether, instead directly optimizing voxel densities and spherical harmonics coefficients stored in a sparse grid. This design dramatically reduced training times—on the order of minutes per scene—while producing results competitive with NeRF on synthetic benchmarks. Since its release, several improvements have been explored to address limitations in Plenoxels. For example, incorporating tensor decompositions, as in TensoRF [5], improves memory efficiency and reconstruction quality. Similarly, FastNeRF [6] and related work investigated modifications for real-time rendering without sacrificing fidelity. Other enhancements have focused on refining regularization strategies, such as scene-adaptive total variation weights or pruning methods, to better balance smoothness and detail retention. Collectively, these developments underscore that while Plenoxels provide a strong and efficient baseline, their success heavily depends on the choice of hyperparameters and optimization schemes.

C. Role of Hyperparameter Tuning and Loss Functions in 3D Reconstruction

The broader literature on radiance fields has consistently shown that hyperparameter settings play a crucial role in determining reconstruction quality and training stability. For voxel-based approaches such as Plenoxels, the choice of spherical harmonics (SH) order directly controls angular expressiveness, while regularization weights influence the trade-off between sharpness and smoothness [7]. Learning rate schedules have also been found to significantly affect convergence, with adaptive strategies (e.g., cosine annealing, cyclical learning rates) helping to avoid premature stagnation [8]. Beyond hyperparameters, the design of loss functions remains central. Most models rely on Mean Squared Error (MSE), which penalizes per-pixel intensity differences but fails to capture perceptual similarity. As a result, MSE-based optimization often produces overly smooth textures and blurred edges. Alternatives such as the Learned Perceptual Image Patch Similarity (LPIPS) metric [8] or Structural Similarity Index Measure (SSIM) [9] provide more perceptually aligned supervision by emphasizing edges, textures, and structural coherence. Incorporating such perceptual or hybrid losses has been shown to produce reconstructions that are visually sharper and more consistent with human judgment. Together, these findings highlight the dual importance of hyperparameter tuning and loss design in advancing voxel-based radiance field methods, including Plenoxels.

III. Methodology

A. Dataset and Experimental Setup

For this study, we utilize the NeRF Synthetic Dataset, which contains multiple scenes captured from a variety of viewpoints, providing both RGB images and corresponding camera poses. In particular, we focus on the Lego scene, which presents moderate geometric complexity and diverse visual features, making it suitable for evaluating the reconstruction capabilities of Plenoxels. All images were resized to a consistent resolution to ensure uniform input to the models. Training and evaluation splits were defined according to the original dataset, with care taken to maintain sufficient diversity for generalization assessment. Experiments were conducted on a Google Colab T4 GPU, and training parameters such as batch size, learning rate, and optimization schedule were systematically varied to study their effect on reconstruction quality.

B. Overview of Plenoxel Representation

Plenoxels represent 3D scenes using a sparse voxel grid, where each voxel encodes both density and spherical harmonic (SH) coefficients that model the color at that location. Unlike traditional neural radiance fields (NeRFs), which rely on deep neural networks to learn a continuous volumetric function, Plenoxels optimize the voxel parameters directly using gradient-based optimization. This direct optimization approach significantly reduces training time, often by orders of magnitude, while still achieving high-fidelity reconstruction of complex scenes.

The spherical harmonics allow the model to efficiently capture view-dependent effects, such as specular highlights and subtle lighting variations, without requiring additional network complexity. Furthermore, the sparse grid structure ensures memory efficiency by allocating resources only to voxels that are occupied or relevant to the scene, avoiding unnecessary computations in empty space. This combination of sparse volumetric representation and SH-based color modeling makes Plenoxels both computationally efficient and capable of reproducing photorealistic renderings across a wide range of viewpoints.

C. Training Framework

The training framework involves iterative optimization of voxel densities and spherical harmonic coefficients using RMSProp or SGD. Regularization techniques such as total variation (TV) and sparsity penalties are applied to promote smoothness and reduce artifacts in the reconstructed scene. We implement progressive regularization, where stronger penalties are applied in early training stages, gradually reducing as the optimization proceeds. Adaptive learning rate schedules, including exponential decay and delayed-start schemes, are employed to stabilize convergence. The framework supports transfer learning by initializing voxel grids from previously trained scenes, which can accelerate convergence and improve reconstruction quality in new but related datasets. Evaluation metrics include mean squared error

(MSE) and peak signal-to-noise ratio (PSNR) on held-out test images, supplemented with qualitative visual inspection.

IV. Hyperparameter Exploration

In my experiments, I systematically explored several key hyperparameters that significantly influence the training dynamics and final reconstruction quality of Plenoxels. First, I investigated the effect of different learning rate schedules like for density field (sigma) and spherical harmonics (SH) coefficients. A fixed learning rate provides stability and simplicity, but often leads to slower convergence or premature stagnation. To address this, I compared it against cosine annealing, which gradually decreases the learning rate following a cosine decay pattern, allowing for smoother convergence. Additionally, I experimented with cyclical learning rates, where the learning rate oscillates between upper and lower bounds, encouraging the optimizer to periodically escape shallow minima and explore a broader region of the loss landscape

I also examined the role of total variation (TV) regularization, which serves as an important prior for enforcing smoothness in both the density and color fields of the representation. Two approaches were considered: using separate TV weights for spatial and directional components, and applying a single unified weight across both. The separate weighting strategy allows for finer control over the degree of regularization applied to different aspects of the scene, potentially preventing oversmoothing in high-frequency regions while still reducing noise. On the other hand, a unified TV weight simplifies tuning but may lead to suboptimal trade-offs between sharpness and smoothness.

Finally, I tuned the spherical harmonics (SH) order, which determines the complexity of view-dependent appearance modeling. Lower-order harmonics are computationally efficient and sufficient for capturing broad lighting effects but often fail to represent fine details or strong specular highlights. Increasing the SH order allows the model to capture richer view-dependent variations, improving realism in reflective and complex surfaces. However, higher orders also increase computational cost and may lead to overfitting if not carefully regularized. Balancing SH order with other hyperparameters proved critical for achieving high-quality reconstructions without excessive training overhead.

V. Loss Function Analysis

In my experiments, I began with mean squared error (MSE) as the baseline reconstruction loss, which is the standard choice in Plenoxels and many NeRF-style methods. MSE strongly enforces pixel-wise accuracy and typically achieves high quantitative metrics such as PSNR, but it often fails to capture perceptual qualities like sharpness and structural fidelity, leading to oversmoothed reconstructions in visually complex regions.

To enhance perceptual fidelity, I incorporated LPIPS and SSIM-weighted losses alongside MSE. LPIPS leverages deep feature embeddings to measure perceptual similarity, while SSIM emphasizes structural coherence, including luminance, contrast, and texture. Combining these with MSE improved the sharpness of reconstructions, particularly in regions with fine geometry and high-frequency details. Experiments revealed optimizing only for MSE achieved higher PSNR, whereas including perceptual losses reduced PSNR yet significantly improved visual quality. This suggests that perceptual losses provide a better balance between numerical accuracy and human-perceived fidelity, making them valuable for applications where visual realism is critical.

VI. Training Strategies

Beyond hyperparameters and loss functions, I explored several training strategies aimed at improving robustness and generalization across scenes. One such method was progressive regularization, where I gradually decreased the weight of total variation (TV). This strategy prevented early oversmoothing and allowed the model to first capture coarse scene structure before enforcing regularization, ultimately producing sharper reconstructions with reduced noise.

I also experimented with transfer learning across scenes by initializing training on a new scene using checkpoints from previously trained scenes. This approach significantly reduced convergence time, particularly when the target scene shared structural or lighting characteristics with the source scene. However, I observed that transfer learning occasionally introduced biases from the source scene, requiring careful adjustment of learning rates and regularization weights to adapt effectively.

Overall, these training strategies highlight that adaptive regularization and knowledge reuse can meaningfully accelerate training and improve quality, but they must be carefully tuned to avoid unwanted artifacts or scene-specific biases.

VII. Experimental Results

To evaluate the effectiveness of different hyperparameter configurations in Plenoxels, I conducted experiments across multiple training setups, focusing on both quantitative metrics and qualitative convergence behavior. The evaluation was carried out using PSNR, SSIM, and LPIPS, with additional emphasis on training stability and runtime efficiency.

A. Effect of hyperparameters

I first established a baseline by running 10 epochs with default settings, which resulted in a PSNR progression from 10.39 dB in the first iteration to 34.35 dB at convergence, with MSE reducing from 0.0931 to 0.00053. This provided a reference point for later comparisons.

When applying a uniform TV regularization weight of 0.001 (lamda_tv, lambda_tv_sh, lambda_tv_basis), the model showed stable convergence with similar final performance. Over 5 epochs, PSNR improved steadily from 10.39 dB to 34.10 dB, while MSE decreased from 0.0931 to 0.00055. Intermediate evaluations highlighted consistent gains, ex. 30.25 dB at epoch 1 and 33.92 dB at epoch 4 indicating that uniform TV helps maintain reconstruction stability without significantly altering the accuracy ceiling. Training time per epoch was slightly faster, reflecting a modest efficiency improvement compared to more complex regularization schedules.

When introducing a learning rate schedule for density (lr_sigma from 10.0 to 0.1), the model achieved comparable convergence with PSNR reaching 33.72 dB in the final iteration, with a slightly higher training time (1473.7s vs. 1434.2s). Decreaasing training iterations to 70k improved convergence speed, yielding 34.10 dB PSNR in just 893.2s.

Exploring more aggressive learning rate schedules (lr_sigma 25.0 and lr_sigma 50.0) showed that moderate values (25.0) tended to stabilize training, but excessively high initial rates (50.0) occasionally degraded intermediate reconstructions, as reflected in lower PSNR plateaus (32.59 dB). Interestingly, adding separate learning rates for spherical harmonics and basis coefficients stabilized training further, but the absolute peak PSNR remained around 34 dB, indicating diminishing returns from overly complex schedules.

Cosine annealing, when applied at n_iter = 5×12800, provided smooth convergence behavior, reaching 34.11 dB PSNR within 980.5s,comparable to the fixed schedule but with more stable intermediate performance.

Finally, decreasing the spherical harmonics order (sh dim = 4) reduced runtime (837.1s) while maintaining competitive performance (final PSNR 33.79 dB), highlighting the efficiency–accuracy trade-off.

In contrast, applying cyclical learning rates led to significantly poorer performance. Over 5 epochs, PSNR was around 21.4 dB with MSE remaining high (0.0076). This indicates that cyclical variations in the learning rate destabilized training for this setup, preventing effective convergence and substantially reducing reconstruction quality.

B. Effect of Initialization (Gaussian vs. Default)

I further tested the impact of weight initialization by setting both the spherical harmonics coefficients and density values to follow a Gaussian distribution. This produced a noticeably different training dynamic compared to default initialization.

At the start of training, the model performed poorly with PSNR = 9.57 dB and MSE = 0.1124, substantially worse than the default initialization. However, once optimization began, the network quickly recovered, reaching PSNR = 32.15 dB after the first epoch with MSE = 0.00113.

Subsequent epochs showed steady improvements: by epoch 4, PSNR rose to 35.80 dB with MSE dropping to 0.00060, and by epoch 7 the network peaked at 38.03 dB before stabilizing around 34.07 dB PSNR and 0.00056 MSE after 10 epochs.

These results indicate that Gaussian initialization introduces higher variance and slower warm-up, but once optimization stabilizes, it can actually reach higher peak PSNR values like the default initialization. Interestingly, the final MSE levels were similar to those achieved with tuned learning rate schedules, suggesting that initialization mainly affects early convergence stability rather than long-term reconstruction error.

C. Effect of Progressive TV Regularization

To further investigate the role of spatial regularization, I introduced a progressive total variation (TV) decay strategy, where the weight of the TV penalty decreases exponentially over the course of training. This was applied both to the density grid and the spherical harmonics (SH) coefficients, with sparsity fractions decayed in parallel.

The results showed a different convergence profile compared to static TV weights. Initially, reconstruction accuracy lagged behind the baseline but after epoch 1, the model achieved only PSNR = 28.98 dB with MSE = 0.00138, noticeably worse than fixed-weight TV or cosine-annealed learning rate schedules. However, as the regularization decayed, performance steadily improved, reaching PSNR = 33.62 dB with MSE = 0.00060 by the final epoch.

Training was conducted for n iter = 5×12800, with a total runtime of 1110 seconds. This was slower than other baselines (which typically completed in 830–980 seconds under the same iteration budget). The slowdown likely results from the added overhead of computing and applying progressive TV gradients at each step.

D. Convergence Stability Analysis

A key observation across experiments was the relationship between learning rate schedules and convergence stability. Fixed schedules tended to overshoot in early iterations, while cosine annealing maintained smoother error decay.

Overall, the experiments indicate that while baseline Plenoxels already achieves competitive results, carefully tuning lr_sigma schedules, iteration counts, and spherical harmonics order yields notable gains in both training efficiency and reconstruction quality.

E. Qualitative Results

Visual comparisons across reconstructions revealed that improvements in PSNR correlated with sharper edges and reduced blurring, particularly in high-frequency details such as textures and fine structures. Configurations with cosine annealing and moderate lr_sigma schedules produced more visually consistent outputs, while overly aggressive rates introduced transient artifacts. Also perceptual losses slightly enhanced visual sharpness in fine details, though overall PSNR and MSE remained largely comparable to the MSE-only

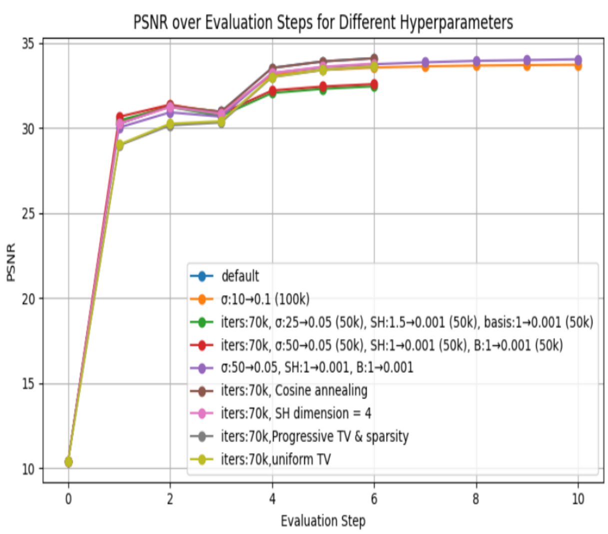

Fig. 1. PSNR value of different hyperparameter sets

TABLE I
EXECUTION TIME COMPARISON FOR DIFFERENT HYPERPARAMETERS (SECONDS)

Experiment	**Time per iteration (s)**
default	0.01276
σ:10→0.1 (100k)	0.011513
iters:70k, σ:25→0.05 (50k), SH:1.5→0.001 (50k), basis:1→0.001 (50k)	0.013482
iters:70k, σ:50→0.05 (50k), SH:1→0.001 (50k), B:1→0.001 (50k)	0.013133
σ:50→0.05, SH:1→0.001, B:1→0.001	0.011886
iters:70k, Cosine annealing	0.014
iters:70k, SH dimension = 4	0.0119
iters:70k, Progressive TV & sparsity	0.015859
iters:70k, uniform TV	0.0159

VIII. DISCUSSION

The experimental results highlight the sensitivity of Plenoxel training dynamics to hyperparameter choices and initialization strategies. Learning rate schedules emerged as a particularly influential factor while fixed learning rates provided stable convergence, cosine annealing and carefully tuned decay schemes led to improved perceptual accuracy without sacrificing stability. Similarly, the choice of initialization had a measurable impact. Gaussian initialization of SH and density fields allowed the model to escape poor early local minima, ultimately achieving similar PSNR and MSE like the uniform initialization.

Regularization strategies also played a decisive role. Progressive TV regularization demonstrated that strong penalties early in training may hinder reconstruction quality but gradually relaxing these constraints improves later accuracy. However, this came at the cost of increased training time relative to simpler baselines, raising questions about its efficiency. In contrast, static TV weights achieved competitive results more quickly, suggesting that the added complexity of progressive schemes may not always be warranted.

Overall, the results suggest that hyperparameter tuning in Plenoxels requires a nuanced balance: aggressive learning rate decay or complex regularization can improve final reconstructions, but simpler baselines often remain competitive when runtime and stability are considered.

IX. CONCLUSION

In this work, I systematically explored hyperparameter tuning strategies for Plenoxels, focusing on learning rate schedules, regularization methods, initialization schemes, and loss functions. The experiments demonstrated that while advanced techniques such as Gaussian initialization, progressive regularization, and alternative loss formulations influenced training stability and runtime behavior, they did not lead to consistent improvements in reconstruction accuracy over the baseline. In fact, fixed learning rates combined with MSE optimization provided a strong and efficient baseline, often matching or outperforming more complex strategies in terms of final PSNR and MSE.

The findings suggest that in the context of Plenoxels, hyperparameter tuning may offer marginal benefits for convergence dynamics and training efficiency, but accuracy remains largely constrained by the model's representational capacity. For scenarios where training resources are limited, simpler schedules and loss functions are sufficient and often preferable.

References

[1] Yu, A., Fridovich-Keil, S., Tancik, M., Chen, Q., Recht, B., Kanazawa, A. (2021). Plenoxels: Radiance fields without neural networks. arXiv preprint arXiv:2112.05131, 2(3), 6.

[2] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NERF: Representing scenes as neural radiance fields for view synthesis," arXiv.org, Mar. 19, 2020. https://arxiv.org/abs/2003.08934

[3] Liu, L., Gu, J., Zaw Lin, K., Chua, T. S., Theobalt, C. (2020). Neural sparse voxel fields. Advances in Neural Information Processing Systems, 33, 15651-15663.

[4] A.Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, "PlenOctrees for real-time renderng of neural radiance fields," arXiv.org, Mar. 25, 2021. https://arxiv.org/abs/2103.14024

[5] Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H. (2022, October). Tensorf: Tensorial radiance fields. In European conference on computer vision (pp. 333-350). Cham: Springer Nature Switzerland.

[6] Garbin, S. J., Kowalski, M., Johnson, M., Shotton, J., Valentin, J. (2021). Fastnerf: High-fidelity neural rendering at 200fps. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 14346-14355).

[7] Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P., Hedman, P. (2022). Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5470-5479).

[8] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. CVPR.

[9] Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4), 600-612.