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
nadeeshan.21@cse.mrt.ac.lk

Uthayasanker Thayasivam
Department of Computer Science and Engineering
University of Moratuwa
Sri Lanka
rtuthaya@cse.mrt.ac.lk

***Abstract*—Plenoxels provide 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 study, the impact of key hyperparameters including spherical harmonics degree, learning rate schedules, and total variation regularization on reconstruction quality and training stability is systematically investigated. Through a series of controlled experiments, an empirical analysis is presented to examine how these factors influence both convergence behavior and perceptual fidelity, even in cases where quantitative accuracy does not consistently improve. The findings offer insights into the strengths and limitations of Plenoxels, highlighting potential directions for future research in radiance field reconstruction without neural networks.**

***Index Terms*—Plenoxels, Radiance Fields, Neural Rendering, Spherical Harmonics, Total Variation Regularization, Cosine Annealing, 3D Reconstruction.**

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.

This study investigates the impact of hyperparameter tuning and spherical harmonics (SH) order adjustments on Plenoxels. Adaptive learning rates, weight initialization strategies, and progressive regularization are systematically examined, along with variations in SH order, to evaluate their effects on reconstruction quality. Although the observed accuracy improvements were limited, the experiments reveal important trade-offs between reconstruction fidelity, computational efficiency, and hyperparameter sensitivity. By documenting these findings, the work provides practical insights to guide 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) [1], [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], [4] combined voxel grids with neural feature interpolation, significantly reducing both memory usage and training time. PlenOctrees [5], [6] 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 [7], [8] 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 [9]–[11], improves memory efficiency and reconstruction quality. Similarly, FastNeRF [12], [13] 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 [14]. 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 [15], [16]. 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 [17], [18] or Structural Similarity Index Measure (SSIM) [19], [20] 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, the NeRF Synthetic Dataset was employed, which contains multiple scenes captured from a variety of viewpoints, providing both RGB images and corresponding camera poses. The Lego scene was selected for experimentation, as it exhibits 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 across models. Training and evaluation splits followed the original dataset configuration, with attention given to maintaining sufficient diversity for generalization assessment. Experiments were performed on a Google Colab T4 GPU, and training parameters such as batch size, learning rate, and optimization schedule were systematically varied to analyze their influence 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 is based on iterative optimization of voxel densities and spherical harmonic coefficients using RMSProp or stochastic gradient descent (SGD). Regularization techniques, including total variation (TV) and sparsity penalties, are applied to promote smoothness and reduce artifacts in the reconstructed scenes. Progressive regularization is implemented, wherein stronger penalties are applied during

the early stages of training and gradually reduced as optimization progresses. Adaptive learning rate schedules, such as exponential decay and delayed-start schemes, are employed to stabilize convergence. Evaluation is performed using mean squared error (MSE) and peak signal-to-noise ratio (PSNR) on held-out test images, supplemented by qualitative visual inspection.

The codebase for reproducing these experiments, including all training scripts and evaluation utilities, is publicly available at GitHub repository.

IV. Hyperparameter Exploration

Several key hyperparameters influencing Plenoxel training and reconstruction quality were systematically analyzed. Different learning rate schedules for the density field and spherical harmonics (SH) coefficients were tested. Fixed rates provided stability but often led to slower convergence. To address this, cosine annealing was applied to gradually reduce the learning rate for smoother optimization, while cyclical learning rates were explored to allow periodic escapes from shallow minima and better loss landscape exploration.

The role of total variation (TV) regularization was also examined, as it serves as an important prior for enforcing smoothness in both the density and color fields of the representation. Two approaches were considered: separate TV weights for spatial and directional components, and a single unified weight applied across both. The separate weighting strategy allows 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. In contrast, a unified TV weight simplifies tuning but may result in suboptimal trade-offs between sharpness and smoothness.

The spherical harmonics (SH) order was tuned to control the complexity of view-dependent appearance modeling. Lower orders efficiently capture broad lighting but miss fine details and specular highlights. Higher orders improve realism on reflective surfaces but increase computation and risk overfitting. Balancing SH order with other hyperparameters proved crucial for achieving high-quality reconstructions efficiently.

V. Loss Function Analysis

In the experiments, mean squared error (MSE) was used as the baseline reconstruction loss, as it 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. However, capturing perceptual qualities such as sharpness and structural fidelity would be beneficial, as their absence can result in over-smoothed reconstructions in visually complex regions.

To enhance perceptual fidelity, LPIPS values were calculated and observed throughout training to assess the quality of the reconstructions, although they were not directly incorporated into the loss function. LPIPS measures perceptual similarity based on deep feature embeddings, offering a complementary perspective to pixel-wise metrics like MSE. By monitoring LPIPS, it was possible to evaluate how well the model preserved sharpness, fine geometry, and high-frequency details in the reconstructed scenes. This approach allowed for the identification of training stages or hyperparameter settings that yielded perceptually superior results, even if traditional quantitative metrics did not fully reflect these improvements. Overall, tracking LPIPS provided a practical and informative way to gauge visual quality and guide the tuning of reconstruction parameters.

VI. Training Strategies

Beyond hyperparameters and loss functions, several training strategies were explored to improve robustness and generalization across scenes. One such approach was progressive regularization, in which the weight of total variation (TV) was gradually decreased over the course of training. This method prevented oversmoothing during early stages, allowing the model to first capture the coarse structure of the scene before regularization was fully applied. As a result, reconstructions exhibited sharper details and reduced noise, while maintaining stability during optimization. This strategy proved particularly effective in balancing the trade-off between capturing global scene geometry and preserving fine-scale visual features.

Overall, these training strategies highlight that adaptive regularization can meaningfully improve reconstruction quality and training stability, but careful tuning is required to avoid unwanted artifacts or scene-specific biases.

VII. Experimental Results

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

A. Effect of hyperparameters

A baseline was first established by running 10 epochs with default settings, resulting in a PSNR progression from 10.39dB in the first iteration to 34.35dB at convergence, with MSE decreasing from 0.0931 to 0.00053. This served as a reference point for subsequent comparisons.

When a uniform TV regularization weight of 0.001 (λ_tv, λ_tv_sh, λ_tv_basis) was applied, stable convergence was observed, with final performance remaining comparable. Over 5 epochs, PSNR steadily improved from 10.39dB to 34.10dB, while MSE decreased from 0.0931 to 0.00055. Intermediate evaluations showed consistent gains, for example, 30.25dB at epoch 1 and 33.92dB at epoch 4, indicating that uniform TV helps maintain reconstruction stability without substantially affecting the accuracy ceiling. Training time per epoch was slightly reduced, reflecting a modest efficiency improvement compared to more complex regularization schedules.

When introducing a learning rate schedule for density (lr_σ 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 (lrσ 25.0 and lrσ 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. Finally, decreasing the spherical harmonics order (sh dim = 4) reduced runtime (837.1 s) while maintaining competitive performance (final PSNR 33.79 dB), highlighting the efficiency–accuracy trade-off.

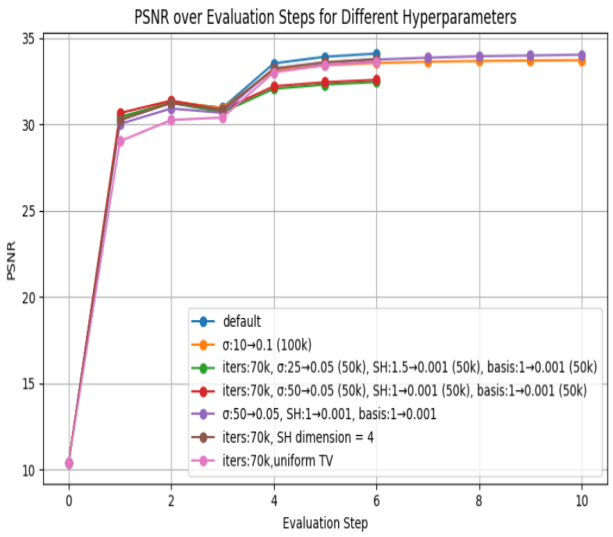

Fig. 1. PSNR value of different hyperparameter sets

Cosine annealing was applied with different configurations to evaluate its impact on convergence and reconstruction quality. Using a maximum period of seventy-six thousand eight hundred iterations, a minimum learning rate of 0.0001, and two thousand warmup steps, PSNR increased steadily from 10.39dB to 34.04dB over seven evaluation points, with MSE decreasing from 0.0931 to 0.00055, and total training time recorded as 883.99seconds. A second configuration with a maximum period of one hundred fifty thousand iterations, a minimum learning rate of 0.0003, and one thousand warmup steps achieved a final PSNR of 33.78dB, with MSE decreasing to 0.00057 over a slightly shorter training time of 867.10seconds. Finally, with a maximum period of forty thousand iterations, a minimum learning rate of 0.00001, and four thousand warmup steps, convergence was slightly slower, reaching 31.77dB PSNR, while MSE reduced to 0.00080 across 862.19seconds. These results indicate that cosine annealing provides stable intermediate performance, with final reconstruction quality comparable across moderate variations in the scheduling parameters.

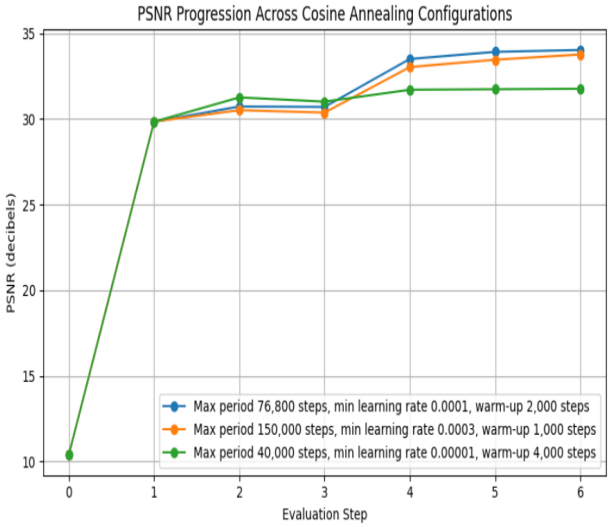

Fig. 2. PSNR value of different cosine annealing configs

In contrast, applying cyclical learning rates resulted in noticeably lower performance and longer training times. With a base learning rate of 0.0001, a maximum learning rate of 0.01, a step size of nineteen thousand two hundred iterations, and a triangular mode, PSNR increased from 10.39dB to 25.54dB over seven evaluation points, while MSE decreased from 0.0931 to 0.00320 across a total training time of 2,515.17seconds. A second configuration with a base learning rate of 0.00005, a maximum learning rate of 0.005, a step size of nine thousand six hundred iterations, and a triangular2 mode reached a final PSNR of 23.26dB with MSE of

0.00510 over 2,754.70seconds. These results indicate that cyclical learning rate schedules destabilized training in this setup, leading to substantially slower convergence and lower reconstruction quality compared to fixed or cosine annealing schedules.

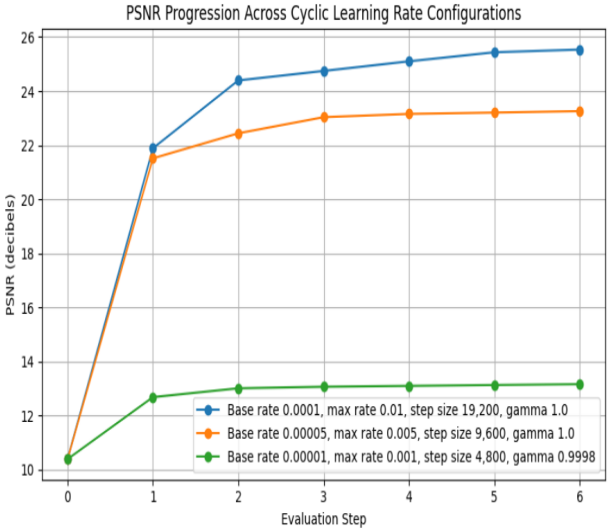

Fig. 3. PSNR value of different cyclic lr configs

.

B. Effect of Initialization (Gaussian vs. Default)

The impact of weight initialization was further tested by setting both the spherical harmonics coefficients and density values to follow a Gaussian distribution with a mean of 0 and a standard deviation of 0.01. A noticeably different training dynamic was produced compared to the 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.

As training progressed, performance continued to improve consistently across epochs. By the fourth epoch, the PSNR had increased significantly to 35.80 dB, accompanied by a corresponding drop in MSE to 0.00060, indicating enhanced reconstruction fidelity. The model achieved its peak performance around the seventh epoch with a PSNR of 38.03 dB, demonstrating strong convergence and effective optimization. Afterward, the results gradually stabilized, maintaining a PSNR of approximately 34.07 dB and an MSE of 0.00056 by the tenth epoch, suggesting that the network had reached a balanced state between reconstruction accuracy and generalization.

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 explore the effect of progressive spatial regularization, a progressive total variation (TV) decay schedule was implemented and applied to both the density grid and spherical harmonics (SH) coefficients, with the sparsity fraction decayed in parallel. In the first experiment, the TV weight for the density grid was set to decay from 0.001 to 0.00001, the SH TV weight from 0.001 to 0.000001, and the sparsity fraction from 0.01 to 0.001. An initial lag in reconstruction accuracy was observed, but performance improved steadily as regularization weakened, reaching a final PSNR of 34.10 dB with MSE = 0.00055 over 959 seconds of training.

In a second experiment, higher initial regularization values were employed: density TV from 0.005 to 0.000005, SH TV from 0.01 to 0.000001, and sparsity fraction from 0.02 to 0.001. Convergence was slower, with a final PSNR of 33.73 dB and MSE = 0.00059 achieved after 1068 seconds. The stronger initial TV constraints suppressed high-frequency details during early iterations, delaying peak performance relative to the first configuration.

In a third configuration, the SH TV weights were reduced from 0.0005 to 0.0000005 while the density TV decay remained from 0.001 to 0.00001 and the sparsity fraction from 0.015 to 0.002. Smoother convergence was observed compared to the second experiment, and a final PSNR of 33.40 dB with MSE = 0.00063 was reached after 1041 seconds of training. The smaller SH TV values were found to help preserve fine details while still benefiting from regularization.

In a fourth experiment, the decay values were set from density TV 0.002 to 0.00002, SH TV from 0.002 to 0.000002, and sparsity fraction from 0.008 to 0.0005. Training lasted 1019 seconds and produced a final PSNR of 33.66 dB with MSE = 0.00060. Moderate regularization on both density and SH coefficients was found to enable efficient convergence while maintaining reconstruction quality.

Across all four experiments, it was observed that progressive TV schedules initially suppressed high-frequency content, causing slower early convergence. However, as TV weights decayed, PSNR values steadily improved, and final MSE values were comparable across configurations. Total training times were influenced by

the magnitude of initial regularization and sparsity, reflecting the computational overhead of applying progressive gradient updates.

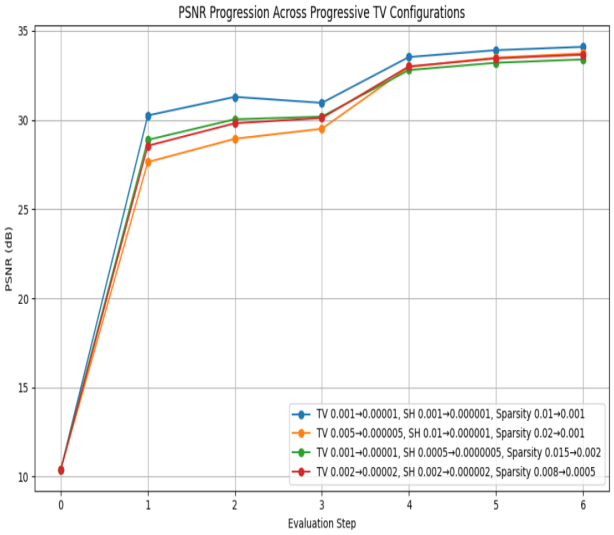

Fig. 4. PSNR value of different progressive tv configs

D. Perceptual Evaluation using LPIPS

To assess the perceptual quality of reconstructions, LPIPS values were calculated for each epoch, without incorporating them into the loss function. This evaluation was carried out using the default training configuration. Initial evaluation on the first epoch yielded an average LPIPS of 0.057, which gradually decreased over training, reaching 0.0066 in the final epoch. This trend indicates that, even though only mean squared error was optimized, perceptual fidelity steadily improved. Observed LPIPS values for each epoch were: 0.0570, 0.0481, 0.0463, 0.0426, 0.0075, 0.0108, 0.0066. These results provide complementary insight into visual quality, demonstrating alignment between quantitative convergence and human-perceived realism.

E. 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.

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.

TABLE I
Execution Time Comparison for Different Hyperparameters (seconds)

Experiment	Time per 1000 iters (s)
default	12.76
Cosine annealing	11.289
cyclic lrs	32.747
SH dimension = 4	11.959
Progressive TV & sparsity	15.859

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, hyperparameter tuning strategies for Plenoxels were systematically explored, with a focus on learning rate schedules, regularization methods, initialization schemes, and loss functions. It was demonstrated through the experiments that, while advanced techniques such as Gaussian initialization, progressive regularization, and alternative loss formulations influenced training stability and runtime behavior, consistent improvements in reconstruction accuracy over the baseline were not achieved. In fact, strong and efficient performance was provided by fixed learning rates combined with MSE optimization, 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] A. Yu, S. Fridovich-Keil, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," arXiv preprint arXiv:2112.05131, 2021.

[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 preprint arXiv:2003.08934, 2020.

[3] L. Liu, J. Gu, K. Z. Lin, T. S. Chua, and C. Theobalt, "Neural sparse voxel fields," in Advances in Neural Information Processing Systems, vol. 33, pp. 15651–15663, 2020.

[4] L. Liu, "Neural Sparse Voxel Fields," Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), 2020.

[5] A. Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, "PlenOctrees for real-time rendering of neural radiance fields," Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Montreal, Canada, Oct. 2021, pp. 5752–5761, doi: 10.1109/ICCV48922.2021.00570 .

[6] L. Wang, J. Zhang, X. Liu, F. Zhao, Y. Zhang, Y. Zhang, M. Wu, L. Xu, and J. Yu, "Fourier PlenOctrees for dynamic radiance field rendering in real-time," Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), New Orleans, LA, USA, Jun. 2022, pp. 13514–13524, doi: 10.1109/CVPR52688.2022.01316

[7] C. Yang, M. Xiao, H. Jia, Y. Xu, H. Duan, and X. Liu, "P-3.4: Plenoxels-based Parallax Map Generation for Flexible Scale Ultra-High Resolution in 3D Imaging," SID Symposium Digest of Technical Papers, vol. 55, no. S1, pp. 725–728, Apr. 2024, doi: 10.1002/sdtp.17187.

[8] K. Jun-Seong, K. Yu-Ji, M. Ye-Bin, and T.-H. Oh, "HDR-Plenoxels: Self-Calibrating high dynamic range radiance fields," in Lecture notes in computer science, 2022, pp. 384–401. doi: 10.1007/978-3-031-19824-3_23.

[9] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "TeNSORF: Tensorial Radiance Fields," in Lecture notes in computer science, 2022, pp. 333–350. doi: 10.1007/978-3-031-19824-3_20.

[10] H. Jang and D. Kim, "D-TensoRF: Tensorial radiance fields for dynamic scenes," arXiv.org, Dec. 05, 2022. https://arxiv.org/abs/2212.02375

[11] A. Luthra, J. Shi, X. Song, Z. Lin, and H. Yu, "Generalized TensoRF: Efficient multi-scene radiance fields and view-consistent 3D editing," Companion Proc. ACM SIGGRAPH Symp. Interact. 3D Graphics Games (I3D), May 2025, pp. 1–3, doi: 10.1145/3722564.3728385 .

[12] S. J. Garbin, M. Kowalski, M. Johnson, J. Shotton, and J. Valentin, "FastNeRF: High-fidelity neural rendering at 200FPS," Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Montreal, Canada, Oct. 2021, pp. 14346–14355, doi: 10.1109/ICCV48922.2021.01408

[13] K. Wadhwani and T. Kojima, "SqueezeNeRF: Further factorized FastNeRF for memory-efficient inference," Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops, New Orleans, LA, USA, Jun. 2022, pp. 2717–2725, doi: 10.1109/CVPRW56347.2022.00307

[14] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-NeRF 360: Unbounded anti-aliased neural radiance fields," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5470–5479, 2022.

[15] M. Debbagh, "Neural Radiance Fields (NeRFs): A review and some insights," arXiv preprint arXiv:2305.00375, 2023.

[16] A. K. M. Rabby, "BeyondPixels: A comprehensive review of the evolution of NeRF," arXiv preprint arXiv:2306.03000, 2023.

[17] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[18] R. Zhang, "Learned Perceptual Image Patch Similarity (LPIPS): A perceptual metric for evaluating image similarity," TransferLab Blog, 2023. [Online]. Available: https://transferlab.ai/blog/perceptual-metrics/

[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[20] T. Oanda, "A review of the image quality metrics used in image synthesis models," Paperspace Blog, 2023. [Online]. Available: https://blog.paperspace.com/review-metrics-image-synthesis-models/