NeRF-based 3D Reconstruction and Orthographic Novel View Synthesis Experiments Using City-Scale Aerial Images

Dylan Callaghan Chapell[2], Edward Ruien Shang[3], Taci Kucukpinar[1], Joshua Fraser[1],
Jaired Collins[1], Vasit Sagan[4], Prasad Calyam[1], Kannappan Palaniappan[1],
[1]Dept. of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO
[2]Colorado College, Colorado Springs, CO
[3]Purdue University, West Lafayette, IN
[4]Dept. of Earth and Atmospheric Sciences, Saint Louis University, St. Louis, MO

***Abstract*—City-scale 3D reconstruction of drone images has many benefits in creating dynamic digital twin models for geospatial and remote sensing applications. We experiment with Neural Radiance Fields (NeRF) to generate novel orthorectified views, point clouds, and 3D meshes using our city-scale image dataset captured from drones and crewed aircraft flights in a circular orbit. We report on the impact of using different parameters related to the NeRF network architecture, ray sampling density, and input image view sampling on the quality of the results. We compare these results with traditional Structure from Motion (SfM) techniques and lidar point clouds. NeRFs can generate highly competitive top-down novel views of city environments compared to traditional SfM techniques, but the underlying 3D structure tends to be less accurate with large-scale scenes. NeRFs can also capture more detail, such as side walls of the buildings, compared to lidar data collections. Finally, we propose a patch-based region of interest training approach to generate high-quality novel top-down views of the large city environments more efficiently for georegistration purposes.**

***Index Terms*—neural radiance fields, NeRF, 3D reconstruction, Gaussian splatting, geospatial, georegistration, aerial images**

I. Introduction

NeRFs have shown great success in generating synthetic novel views, but questions remain regarding the accuracy of the underlying 3D structure, especially for large-scale scenes. Most studies on NeRF focus on smaller-scale settings. We reconstruct 3D point clouds and meshes to assess the accuracy of the 3D structure with aerial images ranging in altitude from 120 meters, where a drone flies in a circular track above a building, to 2 kilometers, where crewed aircraft capture urban environments by flying in a circular track above a city. We use the Transparent Sky [1] dataset for our city-scale 3D reconstruction experiments, which includes multiple cities such as Albuquerque, Berkeley, Columbia, Ferguson, Los Angeles, St. Louis, San Francisco, Syracuse, and more.

We also leverage the high-quality synthetic novel view rendering advancements for georegistration purposes. Our images are captured using an oblique angle. These oblique images need to be orthorectified to improve matching performance with satellite images. Using an orthographic camera and nadir view to render 3D radiance field reconstructions can be a good alternative approach for the orthorectification process as the capabilities of novel view synthesis continue to advance. This paper explores the orthographic nadir view rendering approach with NeRF and Gaussian splatting methods. In addition, we developed a PatchNeRF extension for our NeRF reconstruction pipeline to optimize the georegistration process.

The next section will focus on related work. Section III will detail our pipeline and approach. Section IV will explain our evaluation process. Finally, Section V will present our experiments and results.

979-8-3503-5952-7/23/$31.00 ©2023 IEEE

II. Related Work

NeRFs train a neural network to create a volumetric representation of a scene using images and corresponding camera poses as input. This volumetric representation allows rendering novel views of the scene with traditional volume rendering techniques [2]. The original NeRF model [3] sends rays through pixels of the input images, samples points along these rays, and uses multilayer perceptrons (MLP) for mapping these spatial coordinates to color and density values. Many researchers used grid-based representations of a scene to improve the speed of the training process [4]–[7]. Mip-NeRF [8] and Mip-NeRF 360 [9] trace conical frustums instead of rays to solve the aliasing problem that exists in the original approach. Zip-NeRF [10] and PyNeRF [11] combine these acceleration and anti-aliasing efforts to provide the best quality with high performance. The 3D Gaussian splatting [12] method uses 3D Gaussians for the scene representation and eliminates neural networks in the pipeline with differentiable rendering methods. They achieve better novel view synthesis quality compared to NeRF, and results can be rendered efficiently in real time.

Urban Radiance Fields [13] uses lidar information to improve RGB novel view synthesis and 3D surface reconstruction quality. Transient NeRF [14] uses lidar scans to render novel views of transient images. DS-NeRF [15] and Roessle [16] use sparse point clouds generated from structure from motion

(SfM) methods to achieve a similar or better novel view rendering quality with less number of input images. ReconFusion [17] also focuses on reducing required number of input images by using a diffusion prior to render novel views.

NeRF models require accurate camera poses as input. Camera pose estimation can be an additional time-consuming step in the workflow. NoPe-NeRF [18], NeRF– [19], and COLMAP-Free 3D Gaussian Splatting [20] do not require known camera parameters as an input by jointly optimizing the scene reconstruction and camera parameters.

NeRFMeshing [21] and Neuralangelo [22] optimize the NeRF pipeline to reconstruct accurate 3D meshes.

While most methods primarily focus on smaller-scale environments, some tackle the city-scale reconstruction problem. BungeeNeRF [23] addresses the challenge of modeling city-scale scenes at varying scales by progressively growing the NeRF model and training set, allowing it to adapt to different levels of detail from satellite to ground-level views. Mega-NeRF [24] and Block-NeRF [25] divide city environments into submodels and train these small NeRF models in parallel. Li *et al.* [26] generate a city-scale synthetic dataset using Unreal Engine [27] and test the novel view synthesis performance of multiple NeRF models.

III. Approach

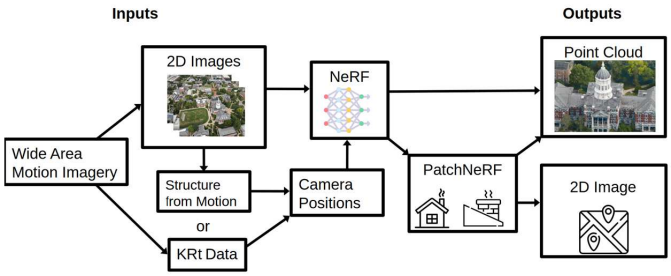


Fig. 1: Our NeRF reconstruction pipeline overview. The pipeline is used to reconstruct 3D point clouds to observe the accuracy of the underlying 3D structure. The pipeline also includes our PatchNeRF extension, which offers specific optimizations for georegistration applications.

NeRF can generate impressive-looking synthetic novel views, which could be useful for georegistration algorithms. We were also interested in whether their reconstruction was physically accurate in the 3D space. Therefore, we conducted 3D reconstruction experiments by using the Nerfstudio framework [28] and its Nerfacto network architecture with our aerial images. Nerfstudio is a modular Python framework that allows modifying each module throughout the NeRF training pipeline according to specific needs. Nerfacto is their custom pipeline that combines the recent advancements in NeRF research, which include ideas from NeRF– [19], Instant-NGP [5], NeRF-W [29], and Ref-NeRF [30].

NeRFs require accurate camera positions and associated images as input. Currently, the usual practice for estimating camera positions and bundle adjustment is using COLMAP [31]. Unfortunately, COLMAP can take several hours to process on a high-end computer. COLMAP also failed to estimate a sufficient amount of camera positions in our Columbia, MO city-scale aerial images from the Transparent Sky dataset. Instead, we obtaiend optimized camera intrinsic and extrinsic (KRt) data for the images in the Transparent Sky dataset using our bundle adjustment software BA4S [32]. This allowed us to have accurate camera positions without the performance cost or failure of COLMAP.

After training NeRFs using our aerial images, we rendered novel orthographic images with nadir views, which could be used for georegistration by matching these orthographic views with satellite images. We also compared the novel view synthesis performance with the recent Gaussian splatting method. We used the original code published by the authors outside of the Nerfstudio environment for our Gaussian splatting experiments. We imported these reconstruction results into Unreal Engine by using a plugin developed by Luma AI [33]. We used the orthographic camera featured in Unreal Engine to render nadir view orthographic images.

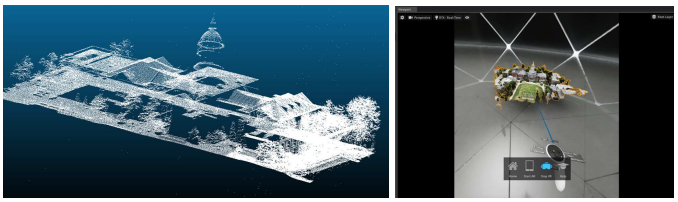
Then, we extracted 3D point clouds for our scenes from Nerfstudio to examine the accuracy of the 3D structure. Additionally, we used the Neuralangelo [22] method to reconstruct 3D meshes outside of the Nerfstudio framework. Neuralangelo is a state-of-the-art surface reconstruction technique that uses a modified NeRF model.

Fig. 2: A binary mask overlayed on an input image for visualization. The mask is generated by an intersection test with rays cast from each pixel and a bounding box. Rays that do not intersect with the bounding box are excluded, allowing a focused and shorter training process.

We developed a PatchNeRF extension that reconstructs only selected regions of interest instead of the whole scene. We do not need to reconstruct the whole scene for georegistration as it would be mostly sufficient to use features from specific landmarks in the scene. Therefore, we can just focus the reconstruction on selected 3D patches in the scene to lower the performance cost by reducing the size of the task. After we get the optimized camera poses, we choose a specific location in the 3D scene and place a 3D bounding box. Then, we send rays through pixels of the input images to check if they intersect with the bounding box. We generate a binary mask for each image according to the results from this intersection test. Fig. 2 shows an example input image with the binary mask

overlayed for visualization. Finally, we exclude the rays that do not intersect with this bounding box from the training, so the network only reconstructs the region inside the bounding box.

IV. Evaluation



(a) CloudCompare lidar visualization (b) Omniverse VR view

Fig. 3: CloudCompare and Omniverse platforms are used for visualization.

We used CloudCompare [34] to visualize, register, and compare the 3D reconstruction results. Fig. 3a shows the visualization of lidar data for the Jesse Hall building located on the University of Missouri campus. It shows that vertical side walls are missed during the data collection flight. We also used NVIDIA's Omniverse [35] software within the virtual reality environment to visually compare the NeRF 3D reconstruction results with our VB3D2 [36] reconstruction algorithm results, which uses traditional Multi-view Stereo (MVS) based computer vision techniques. Visualizing the reconstruction results side-by-side using a virtual reality headset with intuitive controls and 3D perception significantly helps in giving a better idea about the quality of the reconstructed models.

For the quantitative analysis, we followed the approach in the Tanks and Temples [37] dataset. We tested the limits of the NeRF point cloud reconstruction by gradually reducing the number of input images and reported the effect on the reconstructed point clouds.

Let G represent the ground truth point cloud and R denote the reconstructed point cloud, with r being a point in R and g a point in G. For each point in a point cloud, the minimum distance to any point in the other point cloud is calculated.

$$e_{r \to G} = \min_{g \in G} \|r - g\| \tag{1}$$

$$e_{g \to R} = \min_{r \in R} \|g - r\| \tag{2}$$

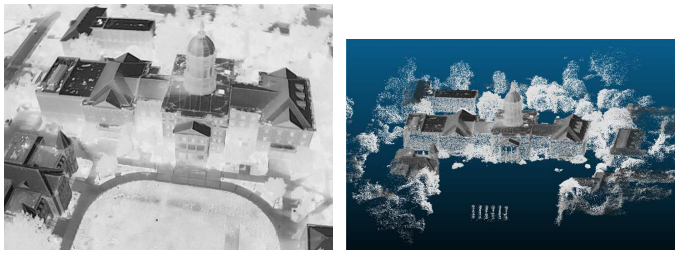
These distances are aggregated to compute precision (P) and recall (R) metrics by choosing a threshold distance d.

$$P(d) = \frac{100}{|R|} \sum_{r \in R} [e_{r \to G} < d] \tag{3}$$

$$R(d) = \frac{100}{|G|} \sum_{g \in G} [e_{g \to R} < d] \tag{4}$$

(a) RGB input image (b) RGB point cloud result

Fig. 4: Jesse Hall Nerfstudio point cloud reconstruction result. 3D structure accuracy is mostly preserved in these smaller-scale examples.



(a) IR input image (b) IR point cloud result

Fig. 5: Jesse Hall Nerfstudio point cloud reconstruction result using IR input images. IR image reconstruction perform similar to RGB image reconstruction except it missed flat ground surfaces.

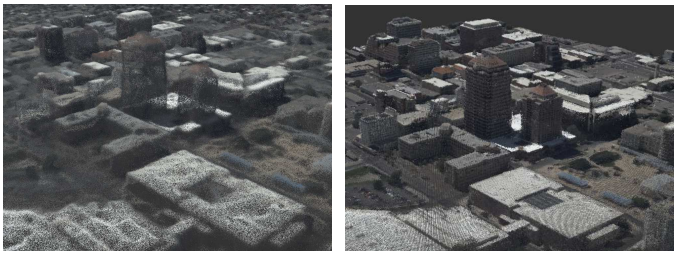
V. Experiments

A. Point Cloud and Mesh Reconstruction

We started the 3D NeRF reconstruction experiments using our drone data collections from the Jesse Hall building, which was captured at around 120 meters of altitude following a circular track with 180 images. (Fig. 4a). Fig. 4 and 5 shows RGB and IR point cloud reconstructions using the Nerfacto model. The 3D structure detail is mostly preserved in this smaller-scale example with varying density of points across the scene. The IR point cloud reconstruction also performed similar, except it failed to reconstruct the flat ground surface.

(a) RGB input image (b) RGB point cloud result

Fig. 6: Albuquerque Nerfstudio point cloud reconstruction result. The reconstruction result got significantly sparse and less detailed with this larger-scale scene.



(a) Nerfstudio point cloud result (b) VB3D2 point cloud result

Fig. 7: Zoomed in point cloud images to compare the point cloud reconstruction results. VB3D2 highlights the significant difference in accuracy and density compared to Nerfstudio result.

After that, we continued with our Albuquerque city-scale images. These images are captured at around 2 kilometers of altitude in a circular orbit with 215 images. It covers a much larger city environment than our Jesse Hall images (Fig. 6a). Unfortunately, the point cloud reconstruction quality has significantly dropped with this larger-scale environment. Fig. 7a shows the sparse result with inaccuracies by zooming into the point cloud. Fig. 7b shows the VB3D2 point cloud result, highlighting the inaccuracies in the Nerfstudio result by comparison.

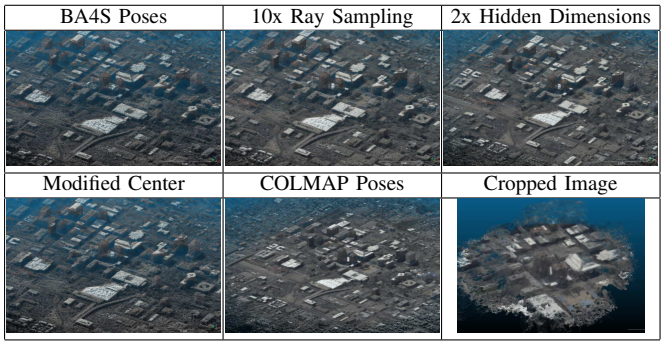


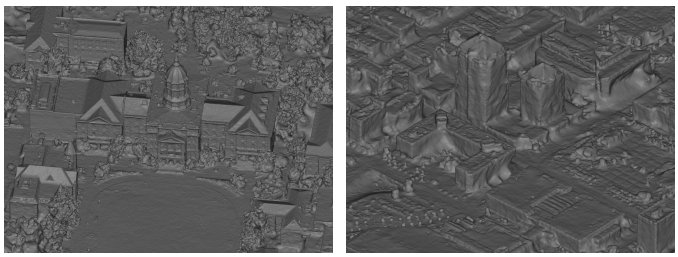
TABLE I: Nerfstudio point cloud reconstruction experiments with different parameters to improve the Albuquerque point cloud reconstruction performance. The experiments did not have a significant effect on the accuracy of the results.

We conducted further experiments with Albuquerque images to improve the point cloud reconstruction accuracy of our Nerfstudio pipeline. We sampled ten times more points along the rays and doubled the size of hidden dimensions in separate experiments. We configured the centering method that was setting the scene center in the middle of a cube bounding box, which was higher above the city. We obtained optimized camera poses both from COLMAP and our BA4S algorithm to compare the reconstruction performance. Finally, we cropped the input images from the center, stabilized the image, adjusted camera parameters accordingly, and trained the network with these smaller-size images. As a result, using higher sampling rates on the rays and increasing the hidden dimensions of the network produced denser results. However, they still contain a significant amount of inaccuracies, thus we would not consider using it as an alternative to VB3D2 method.

Number of Input Images	Precision	Recall
172	0.998	0.997
129	0.998	0.997
86	0.997	0.997
43	0.846	0.816

TABLE II: Precision and Recall values for 3D point cloud reconstruction experiments using fewer number of images, compared with the original model using 215 images. 0.7 meters is chosen as a threshold distance. The current quality of the reconstruction is mostly maintained until reducing the input images down to 43

Then, we progressively decreased the number of input images used in training to observe the corresponding effects on the reconstruction quality. To do this, we divided the original set of 215 images into subsets, each containing five images. From each subset, we incrementally removed a specific number of the first images in subsequent experiments. The reconstruction quality did not significantly diminish until only 43 images were used, which achieved 85% and 82% precision and recall scores. Our pipeline failed to reconstruct a model resembling a city environment when 21 images were used. Table II presents the precision and recall scores, comparing the results obtained from a reduced number of input images to the original reconstruction, which utilized 215 images. This experiment indicates that the number of images may not be the bottleneck for this model's reconstruction performance.



(a) Jesse Hall 3D mesh result (b) Albuquerque 3D mesh result

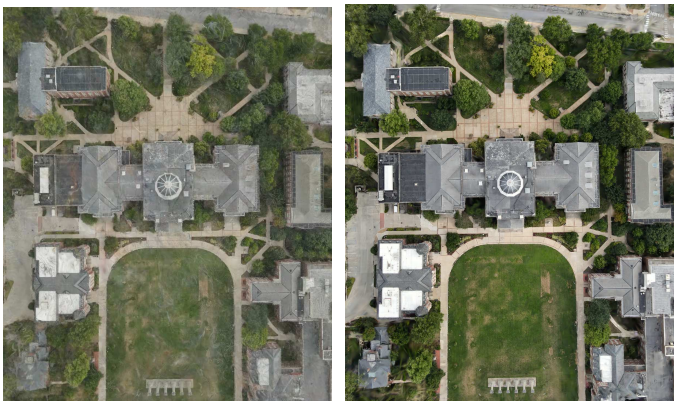
Fig. 8: Jesse Hall and Albuquerque mesh reconstruction results using Neuralangelo. The quality significantly diminishes as the scale of the scene increases.

We conducted 3D mesh reconstruction experiments with the Neuralangelo model outside of the Nerfstudio pipeline. In these experiments, Neuralangelo successfully reconstructed a detailed 3D mesh of Jesse Hall. However, when applied to the larger-scale Albuquerque images, the quality noticeably declined. Although the Albuquerque reconstruction quality by Neuralangelo was more acceptable than our Nerfstudio point cloud reconstruction, it lacked many details evident in the Jesse Hall result and offered a much simpler representation of the city-scale environment. Additionally, Neuralangelo comes with a significant performance cost. It requires 20 days of

training on an NVIDIA A100 GPU for 180 images at a resolution of 8000x6000.

These experiments suggest that the quality of the underlying 3D structure significantly deteriorates in NeRFs as the scale of the scene increases, resulting in a performance that is inferior compared to traditional methods. The results from Neuralangelo, a state-of-the-art NeRF architecture specifically optimized for surface extraction, show the difficulty and room for improvement in city-scale 3D point cloud and mesh reconstruction performance.

B. Orthographic Novel View Rendering



(a) Nerfstudio novel view rendering result (b) Gaussian splatting novel view rendering result

Fig. 9: Comparison of novel view synthesis performance between our Nerfstudio pipeline and Gaussian splatting method. Gaussian splatting greatly increases the novel view synthesis quality.

NeRFs are mostly used for their great synthetic novel view generation performance. Recently, the Gaussian splatting method further improved the novel view synthesis performance using differentiable rendering techniques without any neural networks. We tried these recent advancements on our aerial datasets for nadir view generation. Our data collections are captured with around 45 degrees of camera pitch angle. So, the input images provided to these algorithms do not include any images with a nadir view. Fig. 9 showcases the difference between the results from our Nerfstudio pipeline and the Gaussian splatting method. These images are rendered using a perspective camera with a nadir view. The Gaussian splatting method provides a much more crisp image with fewer artifacts. The training time is around an hour for both of these methods on an NVIDIA A100 GPU. However, the Gaussian splatting method requires a much larger VRAM capacity.

Next, we imported Gaussian splatting reconstruction results into Unreal Engine using the Luma AI plugin. We used the orthographic camera in Unreal Engine to render orthographic images with a nadir view, which produced very good results. Fig. 10 and 11 shows the perspective and orthographic nadir view rendering results from Unreal Engine. Currently, the orthographic images we render using Unreal Engine are lower

(a) Perspective camera (b) Orthographic camera

Fig. 10: Jesse Hall novel nadir view rendering examples using perspective and orthographic camera.

(a) Perspective camera (b) Orthographic camera

Fig. 11: Albuquerque novel nadir view rendering examples using perspective and orthographic camera.

resolution because of the incompatibilities of Gaussian splatting rendering in Unreal Engine. So, we plan to implement our orthographic camera into one of the Gaussian splatting viewers. Additionally, we plan to test orthorectification algorithms with these images and match them with satellite images.

Our PatchNeRF extension is a first step towards building a pipeline specifically optimized for georegistration. By focusing on selected 3D patches rather than the entire scene, we achieved comparable results with training 7,500 iterations compared to 30,000 iterations using the whole scene. These experiments took 10 minutes and 46 minutes, respectively. Fig. 12 shows the results by zooming into the side of the Jesse Hall building, which is our region of interest. Fig. 12a is the result of reconstructing the whole scene, whereas Fig. 12b reconstructed only this region of the scene. The total number of rays used in training is the same in both cases. That means our region of interest received all the rays in the patch-

(a) Default approach novel nadir view rendering result achieved in 46 minutes

(b) PatchNeRF approach novel nadir view rendering result achieved in 10 minutes

Fig. 12: Comparing the full scene reconstruction (a) with targeted 3D patch reconstruction (b) by zooming into the selected region of interest. Achieved similar novel nadir view synthesis results for our region of interest with a significantly reduced performance cost.

based approach, whereas these rays were distributed across the scene in the default method. This allows us to achieve the same reconstruction quality for the region of interest with a much lower performance cost. We would like to extend this approach for Neuralangelo and Gaussian splatting methods, where the performance cost is much higher. The next step is developing a georegistration pipeline where we automatically select multiple regions of interest, render orthographic nadir views, and match with the satellite images.

Conclusion

In this paper, we shared our experiments with NeRF and Gaussian splatting methods using our aerial images that include large city-scale environments. We showcased that even though NeRFs are able to generate high-quality synthetic novel views, the underlying 3D structure is greatly impacted as the scale of the scene increases. We conclude that the city-scale point cloud and mesh reconstruction quality using NeRFs are not ideal for our research purposes, as they do not meet the necessary standards of accuracy and detail required for our specific applications and fall short compared to traditional methods.

Furthermore, we rendered orthographic nadir view images and developed a PatchNeRF approach to specifically optimize the georegistration process. We believe there is significant potential in developing an end-to-end georegistration pipeline utilizing NeRF and Gaussian splatting techniques.

Acknowledgment

Dylan Callaghan Chapell (Colorado College) and Edward Ruien Shang (Purdue University) were supported by the NSF REU program at the University of Missouri EECS Department. The research was supported by the National Science Foundation under the award CNS-2243619 (REU).

References

[1] "Transparent sky." https://transparentsky.net/. (Accessed Dec. 13, 2023).

[2] J. T. Kajiya and B. P. Von Herzen, "Ray tracing volume densities," *SIGGRAPH Comput. Graph.*, vol. 18, p. 165–174, jan 1984.

[3] 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," in *ECCV*, 2020.

[4] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5459–5469, June 2022.

[5] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, pp. 102:1–102:15, July 2022.

[6] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5501–5510, June 2022.

[7] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European Conference on Computer Vision (ECCV)*, 2022.

[8] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5855–5864, October 2021.

[9] 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, June 2022.

[10] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Zip-nerf: Anti-aliased grid-based neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19697–19705, October 2023.

[11] H. Turki, M. Zollhöfer, C. Richardt, and D. Ramanan, "Pynerf: Pyramidal neural radiance fields," in *Thirty-Seventh Conference on Neural Information Processing Systems*, 2023.

[12] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, July 2023.

[13] K. Rematas, A. Liu, P. P. Srinivasan, J. T. Barron, A. Tagliasacchi, T. Funkhouser, and V. Ferrari, "Urban radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12932–12942, June 2022.

[14] A. Malik, P. Mirdehghan, S. Nousias, K. N. Kutulakos, and D. B. Lindell, "Transient neural radiance fields for lidar view synthesis and 3d reconstruction," *NeurIPS*, 2023.

[15] K. Deng, A. Liu, J.-Y. Zhu, and D. Ramanan, "Depth-supervised nerf: Fewer views and faster training for free," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12882–12891, June 2022.

[16] B. Roessle, J. T. Barron, B. Mildenhall, P. P. Srinivasan, and M. Nießner, "Dense depth priors for neural radiance fields from sparse input views," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12892–12901, June 2022.

[17] R. Wu, B. Mildenhall, P. Henzler, K. Park, R. Gao, D. Watson, P. P. Srinivasan, D. Verbin, J. T. Barron, B. Poole, and A. Holynski, "Reconfusion: 3d reconstruction with diffusion priors," *arXiv*, 2023.

[18] W. Bian, Z. Wang, K. Li, J.-W. Bian, and V. A. Prisacariu, "Nope-nerf: Optimising neural radiance field with no pose prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4160–4169, June 2023.

[19] Z. Wang, S. Wu, W. Xie, M. Chen, and V. A. Prisacariu, "NeRF−−: Neural radiance fields without known camera parameters," *arXiv preprint arXiv:2102.07064*, 2021.

[20] Y. Fu, S. Liu, A. Kulkarni, J. Kautz, A. A. Efros, and X. Wang, "Colmap-free 3d gaussian splatting," 2023.

[21] M.-J. Rakotosaona, F. Manhardt, D. M. Arroyo, M. Niemeyer, A. Kundu, and F. Tombari, "Nerfmeshing: Distilling neural radiance fields into geometrically-accurate 3d meshes," in *International Conference on 3D Vision (3DV)*, 2023.

[22] Z. Li, T. Müller, A. Evans, R. H. Taylor, M. Unberath, M.-Y. Liu, and C.-H. Lin, "Neuralangelo: High-fidelity neural surface reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8456–8465, June 2023.

[23] Y. Xiangli, L. Xu, X. Pan, N. Zhao, A. Rao, C. Theobalt, B. Dai, and D. Lin, "Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering," in *The European Conference on Computer Vision (ECCV)*, 2022.

[24] H. Turki, D. Ramanan, and M. Satyanarayanan, "Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12922–12931, June 2022.

[25] M. Tancik, V. Casser, X. Yan, S. Pradhan, B. Mildenhall, P. P. Srinivasan, J. T. Barron, and H. Kretzschmar, "Block-nerf: Scalable large scene neural view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8248–8258, June 2022.

[26] Y. Li, L. Jiang, L. Xu, Y. Xiangli, Z. Wang, D. Lin, and B. Dai, "Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3205–3215, 2023.

[27] "Unreal engine 5.3 documentation." https://docs.unrealengine.com/5.3/en-US/. (Accessed Dec. 13, 2023).

[28] M. Tancik, E. Weber, E. Ng, R. Li, B. Yi, J. Kerr, T. Wang, A. Kristoffersen, J. Austin, K. Salahi, A. Ahuja, D. McAllister, and A. Kanazawa, "Nerfstudio: A modular framework for neural radiance field development," in *ACM SIGGRAPH 2023 Conference Proceedings*, SIGGRAPH '23, 2023.

[29] R. Martin-Brualla, N. Radwan, M. S. M. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, "Nerf in the wild: Neural radiance fields for unconstrained photo collections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7210–7219, June 2021.

[30] D. Verbin, P. Hedman, B. Mildenhall, T. Zickler, J. T. Barron, and P. P. Srinivasan, "Ref-nerf: Structured view-dependent appearance for neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5491–5500, June 2022.

[31] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[32] H. Aliakbarpour, K. Palaniappan, and G. Seetharaman, "Fast structure from motion for sequential and wide area motion imagery," *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp. 1086–1093, 2015.

[33] "Luma ai." https://lumalabs.ai/. (Accessed Dec. 13, 2023).

[34] "Cloudcompare - documentation." https://www.danielgm.net/cc/documentation.html. (Accessed Dec. 13, 2023).

[35] "Nvidia omniverse documentation." https://docs.omniverse.nvidia.com/. (Accessed Dec. 11, 2023).

[36] S. Yao, H. AliAkbarpour, G. Seetharaman, and K. Palaniappan, "3D patch-based multi-view stereo for high-resolution imagery," in *Geospatial Informatics, Motion Imagery, and Network Analytics VIII* (K. Palaniappan, P. J. Doucette, and G. Seetharaman, eds.), vol. 10645, p. 106450K, International Society for Optics and Photonics, SPIE, 2018.

[37] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics*, vol. 36, no. 4, 2017.