Enhancing NeRFs for High-Quality Indoor Video Generation: A Study on Parameterization and Recording Methods

Tafnes Silva Barbosa*
Laboratório de Aprendizado de Máquina Aplicado à Indústria
São José dos Campos, São Paulo, BR
tafnessilvabarbosa@gmail.com

Luis Henrique da S. Resende*
Federal University of Technology – Paraná
Santa Helena, Paraná, BR
luisresende@alunos.utfpr.edu.br

Iuri Almeida Pereira
Federal University of Technology – Paraná
Santa Helena, Paraná, BR
iuripereira.2022@alunos.utfpr.edu.br

Igor Scaliante Wiese
Federal University of Technology – Paraná
Campo Mourão, Paraná, BR
igor@utfpr.edu.br

Thiago França Naves
Federal University of Technology – Paraná
Santa Helena, Paraná, BR
naves@utfpr.edu.br

Telma Woerle de Lima Soares
Federal University of Goiás
Goiânia, Goiás, BR
telma_woerle@ufg.br

Anderson da Silva Soares
Federal University of Goiás
Goiânia, Goiás, BR
andersonsoares@ufg.br

Abstract

Context: The generation of 3D representations with Neural Radiance Fields (NeRFs) has revolutionized areas like virtual reality and space visualization, but video capture for these models lacks a systematic approach. **Problem:** Video capture for NeRFs is still based on trial and error, with few well-defined parameters, leading to inefficiencies, rework, and increased training times. This process is particularly challenging in indoor environments, where variations such as lighting and camera angles significantly impact the final quality of the reconstructions. **Solution:** This paper proposes a systematic approach to optimize video capture, evaluating parameters such as lighting, camera zoom, camera path, and height while presenting metrics to reduce visual artifacts. **Information Systems Theory:** The research is based on the Task-Technology Fit (TTF) Theory, which explores how technology should be adjusted to the specific needs of tasks, aiming to optimize video capture to enhance the quality of the generated models. **Method:** The research follows an experimental approach, using the Nerfstudio tool to test various parameters in a dataset of 48 videos. The analysis is quantitative, evaluating reconstruction quality metrics. **Summary of Results:** Lighting significantly impacts the quality of reconstructions, while changes in capture angles adversely affect the results. **Contributions and Impact in the IS field:** The research contributes a methodology to optimize video capture in NeRFs, driving technological advancements.

CCS Concepts

• **Information systems** → *Computing platforms*; • **Computing methodologies** → *Reconstruction*; Neural networks.

*Both authors contributed equally to this research.

Keywords

Neural Radiance Fields (NeRF), Technology-Task Fit (TTF), Parametrization, Recording Methods

1 Introduction

In the early stages of Neural Radiance Fields (NeRFs), the initial work faced the challenge of surpassing deep convolutional network training to predict sampled volumetric representations [23]. These advances have led NeRFs to evolve significantly since then, resulting in near-realistic images. The most impressive aspect contributing to NeRF popularization is that data captured by reasonably low-quality devices, such as smartphones, is sufficient to reproduce objects from indoor and outdoor scenes [21] [3].

NeRF technology has revolutionized the creation of 3-D object representations from photos and videos, a task previously done using photogrammetry, which reconstructs 3-D structures by matching visual features across images [11]. Since its introduction, NeRF has quickly advanced, with models being refined over the past two years to enhance quality and performance.

Despite improvements in generative models, quality, and efficiency , most research has focused on optimizing volumetric representations [17], real-time rendering [15] [25], and increasing image resolution [3]. However, exploring parameterization in video and image capture, including factors like image quality, lighting, camera orientation, and recording time, remains underexplored. Video capture for NeRF applications faces challenges like low-quality images, blur, and recording instabilities, which can significantly degrade the quality of recorded videos and the resulting reconstructed 3-D scenes.

This paper addresses this research gap based on Task-Technology Fit (TTF) Theory [20] by introducing routines for producing high-quality videos and enhancing NeRF-generated models. We propose specific settings for lighting, camera zoom, capture duration, path,

orientation, and height. Parameterization techniques and standardized recording methods are introduced to maximize video quality, adjusting technology to meet consistency and artifact reduction needs. Since many NeRF users still rely on trial and error to create indoor scenes, this research is relevant in reducing rework, training time, and resource consumption, thereby advancing the field.

Among the main results, we found that, regarding the reconstruction quality, illumination was the parameter with more difference comparing the setups tested. We also found that big changes in shooting angles during filming negatively impacted the final result.

The main contributions of the paper are (i) a list of parameters tested using Nerfstudio, (ii) a dataset comprising 48 videos with their respective Nerfstudio data, ≈ 250 GB, (iii) a Jupyter Notebook with the code to compute the evaluated metrics and replication package [2], and (iv) a set of guidelines describing how to guide users to create better indoor scenes.

2 Theoretical Framework

2.1 Technology-Task Fit (TTF)

The Task-Technology Fit (TTF) model, developed by Goodhue [9], was created to understand how technology can effectively support tasks. The core concept of TTF is that the successful adoption of a technology depends on how well its characteristics align with the task's requirements.In a later work, Goodhue and Thompson [10] present a more general view, where the factors guiding the understanding of fit are solely the characteristics of the task and the technology itself [1]. In this context, the success of technology adoption relies on aligning these characteristics without necessarily involving the user as the main determinant.

TTF motivates this work by investigating how the technology can be adjusted to achieve the desired results, examining how changes in configuration and recording parameters affect the quality of 3D reconstruction with NeRFs.

2.2 Neural Radiance Fields (NeRF)

Neural Radiance Fields (NeRF) emerged in 2020 as an innovative approach to computer vision for synthesizing new visualizations of complex scenes. NeRF represents the scene as a continuous volumetric function that generates color and density at any point in space. This function receives a 5D input, consisting of a 3-D spatial location (x, y, z) and a 2-D viewing direction (θ, ϕ) [23].

The central idea behind NeRF is to use a fully connected deep neural network to model this 5D function. The network is trained to predict volumetric density and view-dependent emitted radiance for any spatial location, enabling the generation of realistic visualizations from different viewpoints while maintaining precise details of lighting and geometry. Neural networks , in turn, are computational models inspired by the workings of the human brain, designed to recognize patterns and learn from data. They consist of layers of units called neurons, which are interconnected. Each neuron receives information, processes it, and passes the result to neurons in the next layer. The network learns by adjusting these connections through multiple iterations to make the generated outputs more accurate over time.

2.3 Nerfstudio and Nerfacto

Nerfstudio is an open-source framework designed to create, train, and evaluate various NeRF models[30]. Developed with a modular architecture, it enables flexible experimentation with different NeRF configurations. This modularity was key to the development of the Nerfacto model, which integrates concepts from several research articles [4] [32] [24] [21]. Combining different approaches into a single model significantly contributes to its effectiveness and performance, positioning it as one of the leading NeRF models.

3 Related Work

The first work on NeRF (NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis) had to overcome various challenges, given that previous works had already delivered impressive results. To achieve this, they aimed to directly optimize the parameters of a continuous 5D representation of the scene, minimizing the rendering error of a set of captured images [22].

A major challenge in using NeRF is handling blur, which often occurs when capturing multiple images from various angles to achieve high-quality results [23] [21]. Even when images are correctly captured and calibrated, blurriness, such as out-of-focus blurring in parts of the image due to big depth variations in the scene while using a large aperture, can lead to artifacts. These artifacts ultimately reduce the quality of the NeRF-generated images.

Previous research has addressed image blurring in NeRF models. NeRF-W [21] handles changes in lighting and moving objects, while Mip-NeRF [3] improves NeRF performance with inputs at different scales. Deblur-NeRF [19] specifically tackles training NeRF from blurry images by using neural networks to reconstruct radiance fields, enhancing the quality of generated views despite defocus or motion blur.

In addition to addressing blur, research has explored enhancing NeRF output by combining segmentation techniques to improve visualization accuracy and consistency [18]. NeRF On-the-go uses uncertainty predictions to filter unnecessary scene elements, producing cleaner view synthesis [26]. Lightning NeRF improves computational efficiency and reconstruction quality, particularly in autonomous driving scenarios, enhancing training and rendering speed [6].

On the other hand, approaches like NeRF-VPT introduce visual prompt tuning, which uses prior RGB information to optimize the rendering of new views [7]. The parameterization of NeRFs from 3D point clouds has also been explored, proposing methods such as using KNN algorithms to improve the accuracy of 3D object representation [35].

Several studies have focused on improving NeRF models to enhance 3D representation quality. [16] worked on view synthesis from a single image by combining global and local features for better reconstruction. [8] introduced Re: NeRF to reduce storage for voxelized models while maintaining performance. [13] used event data to reconstruct images in conditions with high variation and noise, showing NeRF's effectiveness in difficult contexts. [14] developed CG-NeRF to generate multi-view images from multimodal conditions, resolving pose inconsistencies while ensuring consistent quality. These studies highlight NeRF's continuous evolution and growing applications.

Table 1: Video shooting parameters.

Parameter	Options		
Environment Scene	Indoor		
Cellphone Orientation	Vertical	Horizontal	
Duration	30s	60s	120s
Illumination	Artificial	Natural	
Shooting Angle	Normal Floor-Normal	Normal-Ceil Floor-Ceil	

As we can observe, NeRF has been an important research topic in recent years due to the range of applications that can be explored with this technique. Despite these advancements and academic efforts, we found a research gap in understanding how the input parameters for NeRF videos might directly impact the quality of the generated scenes, the amount of manual rework, and the computational cost of reprocessing different versions of captured videos.

4 Experimental Setup

4.1 Aim and Research Questions

Neural radiance fields (NeRF) are being used to reconstruct tridimensional objects and scenes directly from videos or sets of images. Since the original paper was released [23], many improvements in the technique have been made [3–5]. This paper aims to draw guidelines for making indoor scene videos used by NeRFs by comparing the video recording parameters.

We evaluate different camera positions, illumination, video duration, and shooting angle configurations. The main research question investigated in this work is: to what extent do the parameters of a video influence the quality of the final scene reconstructed by a NeRF algorithm? It is worth mentioning that, in this paper, we only investigated indoor scenes, leaving the outdoor scenes as future work.

Since many different models and applications use various types of NeRFs, this work used Nerfstudio's Nefacto model to reconstruct the scenes and to evaluate the metrics for each one of them.

4.2 Dataset Construction

Using an iPhone 13 with image stabilization and 0.5x zoom to capture most of the scene, the dataset consists of 48 recordings considering the list of parameters in Table 1. These parameters were analyzed to determine their impact on reconstruction quality. The last row of Table 1 can be visualized with Figure 1, where the cube represents an imaginary object, with the camera trajectory shown in red. For the Normal-Ceiling parameter, the camera follows two circular paths: one near the object's middle and the other near the ceiling.

The replication package [2] provides practical examples of the capture formats. Since NeRFs work well for centered objects, the videos were shot walking around an imaginary object for the scenes, which showed good early results. The reconstruction of the floor and ceiling was not good when recording with the camera at the center of the scene and turning around the camera axis.

Nerfstudio works with plenty of possible inputs, such as videos, sets of images, etc. When using videos, it extracts their frames and selects nearly 300 frames by default. Figure 2 shows some examples of frames extracted from one of the 48 videos recorded in the indoor environment that we used in this work. But what should be done when the video has blurred frames? It is known that blurred images negatively affect the results of tridimensional reconstruction[19]. So, for every video, to make a better selection of frames, the frames were extracted early and, for each one of them, it was computed the laplacian [27] as seen in Eq. (1), a metric used to measure blur in images since the higher the details of an image, the higher its laplacian. The laplacians were computed from the grayscale conversion of the frames.

$$L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{1}$$

The initial goal for the number of selected frames was also 300. So, for every video, the extracted frames were separated into approximately 300 windows of images. The image with the highest laplacian within a window is selected to compose the final dataset for that video. Thus, when the windows cover a short period, there is not much structural difference between frames, making the frame with the biggest laplacian the least blurred one.

4.3 Evaluation Metrics

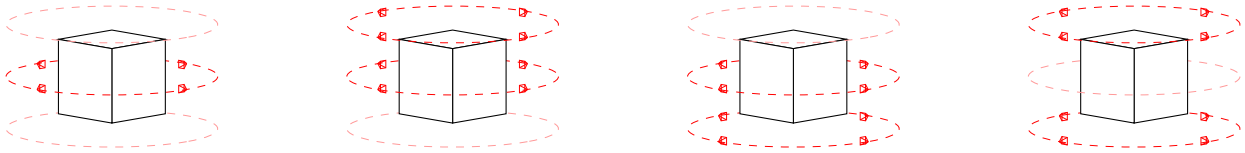
Since the datasets were created using the laplacian of the images to select them, the values of this metric were computed for every selected image to include them in the comparison. They can show how blurry or low-detailed an image dataset is.

Another metric comes from the COLMAP[28, 29] use. COLMAP is a software that uses Structure-from-Motion (SfM) to detect and match features between images and also uses Multi-View Stereo (MVS) to generate 3D cloud points which may be used as initial 3D models for Nerfstudio's models. The input data must be preprocessed before being used for the model itself. Nerfstudio uses COLMAP to find the camera poses of the dataset images. However, the COLMAP results are not always the same, i.e., they are a bit random. Sometimes, COLMAP finds most of the poses in the first camera model, but when it does not, it may find them in subsequent models. This work used the camera model with the largest percentage of poses found. Considering this, the datasets were preprocessed ten times initially to compute the average number of found poses from the best camera model suggested by COLMAP to find correlations with the video content.

During the model training, RAM and GPU memories were computed for COLMAP and Nerfacto, along with the time to complete these steps.[1]

Besides these metrics, three more metrics included in Nerfstudio's evaluation, which are also used in other works[12], were used in this work: PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural

[1]The experiments ran on a personal computer with processor AMD Ryzen 7 with 16GB RAM, GeForce RTX3070ti GPU with 8GB VRAM, Graphics' Driver Nvidia SMI 550.54.15, Cuda Toolkit 11.8 and Linux distribution Ubuntu 22.04 LTS. In addition, it used Nerfstudio 1.1.4.



(a) Normal. (b) Normal-Ceiling. (c) Floor-Normal. (d) Floor-Ceiling.

Figure 1: Video shooting angles. Taken camera trajectories (in strong red) and not taken ones (in weak red). Scene imaginary object (in black).

Figure 2: Images of frames extracted from the videos of the constructed dataset.

Similarity Index), and LPIPS (Learned Perceptual Image Patch Similarity), where each is calculated in a smaller set of images of the initial dataset. PSNR is a traditional signal fidelity measurement in Eq. (2).

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right), \tag{2}$$

where MAX_I is the maximum possible pixel value of the image and MSE is the mean squared error among all the image pixels[33]. Higher values mean better results for PSNR. SSIM is a metric computed on the luminance, contrast, and structure features of the actual and the predicted images as seen in Eq. (3)[31].

$$SSIM(i_a, i_p) = \frac{(2\mu_{i_a}\mu_{i_p} + c_1)(2\sigma_{i_a i_p} + c_2)}{(\mu_{i_a}^2 + \mu_{i_p}^2 + c_1)(\sigma_{i_a}^2 + \sigma_{i_p}^2 + c_2)}, \tag{3}$$

where i_a and i_p are the actual and predicted images, respectively, c_1 and c_2 are small stabilization constants, and $\mu.$, $\sigma.$, and $\sigma_{i_a i_p}$ are mean, variance and covariance between the images. SSIM ranges from 0 to 1, and higher values mean better results. LPIPS means Learned Perceptual Image Patch Similarity, a method that uses neural networks to compute deep features across different architectures and tasks, getting closer to a human perceptual similarity[34]. LPIPS ranges from 0 to 1, but smaller values mean better results.

4.4 Replication package

The research dataset, Jupyter Notebook, with metrics computation and data analysis, and scripts used in this paper are available in the replication package [2].

5 Results

5.1 COLMAP and Nerfacto Setup

COLMAP step was run inside Nerfstudio using default parameters for the version used with a few modifications:

(1) Since Nerfstudio uses the first camera model suggested by COLMAP, the camera intrinsics refinement should not be done if it is not the best one. Because of that, they were not refined until the best camera model was found.
(2) The matching method was exhaustive since it finds more matches between the frames, comparing them with each other.

As previously mentioned, COLMAP did not deliver the same results deterministically. To reduce the likelihood of errors, it was run three times until it had found all the poses for at least one camera model. An unsuccessful COLMAP step was considered when it did not find all the poses on any of the three tries.

The Nerfacto setup took the same approach as COLMAP: mostly using the default configuration, apart from a few parameters irrelevant to the reconstruction quality. The checkpoints of the network were saved every ten thousand steps up to one hundred thousand steps to analyze if there is a trade-off between time and quality.

5.2 Pilot Study

5.2.1 COLMAP frequency of finding all poses. The first result is the average number of poses found by COLMAP, shown in the bar plot in Figure 3. A threshold of 100% was chosen to graphically divide the videos, which had all poses found on the early ten repetitions

of COLMAP. This was made to show that only a few videos did not have all poses found, but even for these ones, the percentage was high.

In its abscissa, there are black and red tick labels. As mentioned in Sec. 5.1, after the early repetition stage, COLMAP was applied thrice to get a successful number of poses. So, the red ones mean it was impossible to find all the poses when trying to apply COLMAP three times after the early ten repetitions, but this does not mean that COLMAP found a few poses. To show these three tries, in blue, Figure 3 show the number of COLMAP repetitions for this step. In Figure 3, the abscissa tick labels, the names of the video files in our dataset, were shrunk to save space. The legend for these shrunk names can be found in Table 2.

As expected, it is possible to see an increase in the number of repetitions as the bar heights get smaller. But even with three repetitions, COLMAP found all the poses only for one video that passed by it thrice. This shows that a pilot study can tell if COLMAP will be used more when providing information to companies to decide the cost of processing based on the type of video. One can tell that using COLMAP ten times will not compensate the choice of using it three times. This can be solved by using the sequential matcher during the pilot study, which reduces computation time.

It is important to register that choosing the COLMAP's best camera model resulted in a good percentage of found poses for all the videos, which shows that this is a good practice that can save computing time and resources for companies.

5.3 Reconstruction Results

After the COLMAP statistics stage, the experiment was run, and the evaluation metrics mentioned in Sec. 4.3 were computed. They are shown in Table 2.

During the pilot study, not all the tries done in COLMAP were successful, where success is defined as 100% of poses found. The frequency of success may be found in the second column of Table 2. It was included to show the slight difference between itself and the average number of found poses. Most of them have values close to each other, but for a few, they are different since COLMAP found many poses but did not find all of them. Because of that, the results were ordered based on the average number of found poses, looking for patterns in this COLMAP behavior as a possible consequence of the input videos themselves.

5.3.1 Nerfacto's Metrics. These metrics compare images predicted by Nerfacto with images from the real data. Since the experiments were run saving checkpoints every 10k iterations, it is possible to show the evolution of these metrics graphically. They can be seen in Figure 4, colored box-plot-based graphics. Each column represents a video, and each row represents 10k of Nerfacto's iterations. There are color legends for them on the left of the plots.

There are columns where the values do not change considerably, increasing the number of iterations. This shows it is possible to use fewer iterations to achieve similar quality in the reconstruction, which makes the process much faster and saves time and resources.

The values for SSIM and LPIPS are very good for most of the videos. However, it is very interesting to analyze the parameters of Table 2 separately. Table 3 shows the percentage values for the videos analyzed separately for each parameter, considering Nerfacto's metrics and the average percentage of found poses in the early ten repetitions stage.

Table 3 shows that for the average number of found poses for COLMAP, there was not a significant difference for the results of *cellphone position* and *times* parameters. For the *angle* parameter, the worst was the *floor-ceiling*, and for the *light*, the best was the *natural*. There are possible explanations for these results. The cellphone position may only influence the field of view of the reconstruction: filming vertically allows recording more of the environment with few frames, making it more efficient for the processing step. Since approximately 300 frames were extracted from each video, the duration did not influence it, but it could be because recording a scene with small videos may lead to making them faster, which can make the videos more motion blurred. Natural illumination was better, possibly because of the intensity of the light, resulting in less noise when compared with artificial lighting. For the shooting angle, the worst was *floor-ceiling*, which may have been caused by not transitioning the cellphone/camera between different angles as smoothly. This last one may also be caused by a decrease in the overlapping frames since the turns are further away from each other compared to the other angles, which may influence the descriptors matching by COLMAP.

For the metrics SSIM and LPIPS, Table 3 shows that only three parameters stand out: *celphone position – vertical*, *light – natural* and *angle – normal*. Again, the shooting angle *floor-ceiling* had the worst performance, indicating that abrupt transitions and lack of frame overlapping may not favor the quality of the reconstruction. It is also possible to induce that the more thoroughly the scene is explored in the video, the better Nerfacto can reconstruct it. All the other shooting angles performed well in the LPIPS metrics lower than 0.15, but only normal ones performed well for LPIPS lower than 0.125. This may show that with more views, the scene gets more complex, making it harder for Nerfacto to reconstruct it, trading off with a better generalization.

The last result is shown in Figure 5. The videos were ordered based on the laplacians' mean values to show reconstructions with better PSNR for higher laplacian mean values. To mathematically state that a linear regression was made considering the Laplacians mean values and the maximum values of the PSNR of each video. The result is in Figure 6. Even with high residuals and r-square too low (0.24, probably because of two outliers), a relationship between the laplacian and the PSNR can be noticed. This indicates that the Laplacian operator can be used to show if the reconstruction will be good or not somewhat confidently before the NeRF training is finished, saving time and computation.

To summarize our findings, the parameters our work showed to be the most important when recording videos for NeRF training were lighting and shooting angle conditions. To obtain a better quality of the reconstructed scene by Nerfacto, the filmed scene has to be properly illuminated and the frames correctly selected from the video in a way that COLMAP sees a good number of different views for all the shooting angles because the more the algorithm can see of the scene the better the reconstruction will be. Additionally, the transition between the shooting angles must be smooth, or frames must not be selected from transitions to avoid motion blur.

Table 2: Evaluation Metrics by Video.

Scene	Right COLMAP Frequency	COLMAP Avg. Found Poses	Number of Frames	COLMAP Repetitions	Time (min)		RAM (Gb)		VRAM (GiB)		Laplacian			Metrics		
					COLMAP	Nerfacto	COLMAP	Nerfacto	COLMAP	Nerfacto	Min	Mean	Max	Max. PSNR (dB)	Max. SSIM	Min. LPIPS
i_h_30_n_fc	100.0	100.00	321	1	16.18	48.98	5.30	7.91	4.15	2.15	3.75	9.39	18.54	22.87	0.87	0.20
i_h_30_n_n	100.0	100.00	408	1	18.74	48.65	5.30	8.35	4.15	2.15	4.06	8.92	21.63	23.95	0.93	0.13
i_h_30_n_nc	100.0	100.00	418	1	14.35	48.31	5.42	8.54	4.15	2.15	5.00	9.69	24.61	26.55	0.95	0.13
i_h_60_a_fn	100.0	100.00	304	1	5.88	48.55	5.37	7.56	4.15	2.15	3.74	6.14	12.15	22.36	0.92	0.15
i_h_60_a_n	100.0	100.00	309	1	7.46	48.75	5.38	7.84	4.15	2.15	4.01	6.55	10.93	24.09	0.94	0.13
i_h_60_a_nc	100.0	100.00	369	1	6.21	48.66	5.41	8.23	4.15	2.15	3.47	6.08	12.68	23.74	0.94	0.14
i_h_60_n_fn	100.0	100.00	330	1	13.20	47.53	5.56	8.02	4.15	2.15	6.38	12.67	24.79	25.78	0.94	0.12
i_h_60_n_fc	100.0	100.00	319	1	8.22	48.00	5.50	7.95	4.15	2.15	4.53	9.54	21.76	24.26	0.93	0.14
i_h_60_n_n	100.0	100.00	306	1	10.58	48.22	5.52	7.78	4.15	2.15	4.65	10.00	23.61	25.94	0.94	0.12
i_h_60_n_nc	100.0	100.00	328	1	8.79	49.55	4.21	7.21	4.15	2.15	4.35	9.50	28.37	24.13	0.95	0.13
i_v_120_a_fn	100.0	100.00	307	1	4.85	50.59	4.25	7.16	4.15	2.15	3.56	6.11	18.04	25.53	0.94	0.13
i_v_120_n_fc	100.0	100.00	304	1	5.92	50.20	4.46	6.62	4.15	2.15	4.43	8.14	15.52	26.07	0.94	0.15
i_v_120_n_n	100.0	100.00	327	1	12.62	50.39	4.46	6.76	4.15	2.15	4.54	10.96	27.40	26.37	0.95	0.14
i_v_120_n_nc	100.0	100.00	326	1	9.41	49.92	4.63	6.92	4.15	2.15	5.15	10.49	22.22	27.50	0.95	0.14
i_v_30_a_fn	100.0	100.00	337	1	5.52	50.50	4.45	6.98	4.15	2.15	3.40	5.33	9.87	25.67	0.94	0.13
i_v_30_a_fc	100.0	100.00	327	1	4.39	50.40	4.56	6.82	4.15	2.15	3.14	5.80	11.40	25.99	0.93	0.14
i_v_30_a_n	100.0	100.00	343	1	10.54	50.41	4.61	6.97	4.15	2.15	3.90	6.19	12.92	26.23	0.95	0.10
i_v_30_a_nc	100.0	100.00	342	1	6.82	50.61	4.65	7.13	4.15	2.15	3.52	5.97	14.80	26.15	0.95	0.14
i_v_30_n_fn	100.0	100.00	413	1	11.89	50.61	4.54	7.51	4.15	2.15	3.60	7.11	20.97	26.78	0.94	0.14
i_v_30_n_fc	100.0	100.00	436	1	14.72	50.44	4.64	7.76	4.15	2.15	3.30	8.10	19.65	25.83	0.94	0.14
i_v_30_n_n	100.0	100.00	416	1	23.01	50.18	4.77	7.68	4.15	2.15	5.20	11.29	29.17	29.06	0.96	0.12
i_v_30_n_nc	100.0	100.00	382	1	13.92	50.25	5.06	7.32	4.15	2.15	5.53	10.29	27.85	28.36	0.96	0.13
i_v_60_a_fc	100.0	100.00	325	1	4.54	50.51	4.76	7.06	4.15	2.15	3.83	6.02	11.44	25.48	0.95	0.13
i_v_60_a_n	100.0	100.00	353	1	6.43	50.59	4.73	7.20	4.15	2.15	2.65	4.44	7.91	25.09	0.95	0.12
i_v_60_a_nc	100.0	100.00	310	1	6.32	50.49	4.79	7.18	4.15	2.15	3.80	6.11	14.22	25.75	0.95	0.12
i_v_60_n_fn	100.0	100.00	306	1	8.73	50.53	4.88	7.48	4.15	2.15	4.94	9.90	23.63	26.97	0.94	0.14
i_v_60_n_fc	100.0	100.00	355	1	9.44	49.26	4.95	7.58	4.15	2.15	3.40	8.78	17.92	25.99	0.95	0.14
i_v_60_n_n	100.0	100.00	355	1	17.79	49.28	4.96	7.60	4.15	2.15	6.44	12.20	24.80	29.32	0.96	0.11
i_v_60_n_nc	100.0	100.00	339	1	9.18	49.00	5.04	7.34	4.15	2.15	3.63	8.81	20.25	25.62	0.95	0.13
i_v_120_a_nc	100.0	100.00	326	1	6.86	50.54	4.36	6.65	4.15	2.15	4.22	6.78	14.23	25.64	0.94	0.14
i_h_30_n_fn	100.0	100.00	306	1	12.16	48.65	5.35	7.68	4.15	2.15	5.76	11.70	25.50	25.42	0.93	0.13
i_v_120_a_n	100.0	100.00	325	1	13.36	50.43	4.35	6.94	4.15	2.15	7.63	22.51	60.11	27.17	0.89	0.10
i_v_120_n_fn	100.0	100.00	326	1	11.72	50.29	4.39	6.73	4.15	2.15	7.09	12.78	25.46	27.36	0.94	0.14
i_h_120_a_fn	100.0	100.00	319	1	5.28	49.86	5.06	7.40	4.15	2.10	2.64	5.35	9.62	21.32	0.92	0.14
i_h_120_a_n	100.0	100.00	311	1	5.12	49.64	5.12	7.45	4.15	2.15	2.68	4.87	8.57	21.14	0.93	0.14
i_h_120_a_nc	100.0	100.00	312	1	6.94	49.49	5.17	7.43	4.15	2.15	4.20	6.44	12.00	22.08	0.92	0.14
i_h_120_n_fn	100.0	100.00	305	1	12.05	47.65	5.22	7.42	4.15	2.15	7.06	13.37	27.41	26.07	0.93	0.13
i_h_120_n_fc	100.0	100.00	319	1	8.82	49.32	5.26	7.56	4.15	2.15	5.64	10.57	19.58	24.98	0.94	0.13
i_h_120_n_n	100.0	100.00	307	1	12.47	48.85	5.18	7.55	4.15	2.15	5.55	13.09	29.20	26.13	0.95	0.12
i_h_120_n_nc	100.0	100.00	308	1	8.40	49.48	5.27	7.62	4.15	2.15	4.93	10.87	22.46	24.95	0.95	0.14
i_h_30_a_fn	100.0	100.00	415	1	10.71	49.09	5.23	9.20	4.15	2.15	3.56	5.56	15.92	22.22	0.92	0.14
i_h_30_a_n	100.0	100.00	407	1	12.51	49.66	5.34	9.51	4.15	2.15	3.41	5.88	12.80	23.58	0.94	0.12
i_h_30_a_nc	100.0	100.00	437	1	6.23	50.27	4.11	7.10	4.15	2.10	2.37	4.08	7.39	21.63	0.93	0.15
i_h_30_a_fc	60.0	99.91	424	3	20.64	49.49	5.39	9.72	4.15	2.15	3.15	5.53	12.59	17.92	0.78	0.39
i_h_120_a_fc	0.0	98.18	325	3	17.58	49.49	5.14	7.52	4.15	2.15	4.16	6.44	13.85	22.27	0.93	0.15
i_v_120_a_fc	0.0	96.37	303	3	11.26	50.72	4.19	6.68	4.15	2.15	3.58	6.13	12.13	25.10	0.94	0.14
i_v_60_a_fn	80.0	96.37	355	1	4.86	50.86	4.78	7.20	4.15	2.10	3.21	4.89	11.80	25.33	0.94	0.14
i_h_60_a_fc	0.0	91.57	325	3	17.41	49.18	5.32	7.72	4.15	2.15	4.16	6.44	13.85	16.27	0.69	0.63

Scene name legend: 1_2_3_4_5
1 (environment): i – inside
2 (cellphone position): h – horizontal; v – vertical
3 (filming duration)
4 (illumination): a – artificial; n – natural
5 (shooting angles): n – normal; nc – normal-ceiling; fn – floor-normal; fc – floor-ceiling

Table 3: Separately analyze of the videos considering each parameter from Table 1. Bold values mean best when compared to others.

Parameter	Cellphone Orientation		Duration			Illumination		Shooting Angle			
Values	Horizontal	Vertical	30	60	120	Artificial	Natural	Floor-Ceiling	Normal	Normal-Ceiling	Floor-Normal
Colmap ≥ 100%	87,50%	**91,67%**	**93,75%**	87,50%	87,50%	66,67%	**100,00%**	66,67%	**100,00%**	**100,00%**	91,67%
Colmap ≥ 97,5%	**95,83%**	91,67%	**100,00%**	87,50%	93,75%	87,50%	**100,00%**	83,33%	**100,00%**	**100,00%**	91,67%
Colmap ≥ 95%	95,83%	**100,00%**	**100,00%**	93,75%	**100,00%**	95,83%	**100,00%**	91,67%	**100,00%**	**100,00%**	**100,00%**
Colmap ≥ 90%	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**	**100,00%**
Max SSIM>0,85	91,67%	**100,00%**	93,75%	93,75%	**100,00%**	91,67%	**100,00%**	83,33%	**100,00%**	**100,00%**	**100,00%**
Max SSIM>0,9	87,50%	**95,83%**	87,50%	**93,75%**	**93,75%**	87,50%	**95,83%**	75,00%	91,67%	**100,00%**	**100,00%**
Max SSIM>0,95	0,00%	**29,17%**	**18,75%**	**18,75%**	6,25%	8,33%	**20,83%**	0,00%	**33,33%**	25,00%	0,00%
Min LPIPS<0,15	87,50%	**95,83%**	87,50%	**93,75%**	**93,75%**	**91,67%**	**91,67%**	66,67%	**100,00%**	**100,00%**	**100,00%**
Min LPIPS<0,125	16,67%	**25,00%**	18,75%	**31,25%**	12,50%	**20,83%**	**20,83%**	0,00%	**66,67%**	8,33%	8,33%
Min LPIPS<0,1	0,00%	**4,17%**	0,00%	0,00%	**6,25%**	**6,25%**	0,00%	0,00%	**8,33%**	0,00%	0,00%

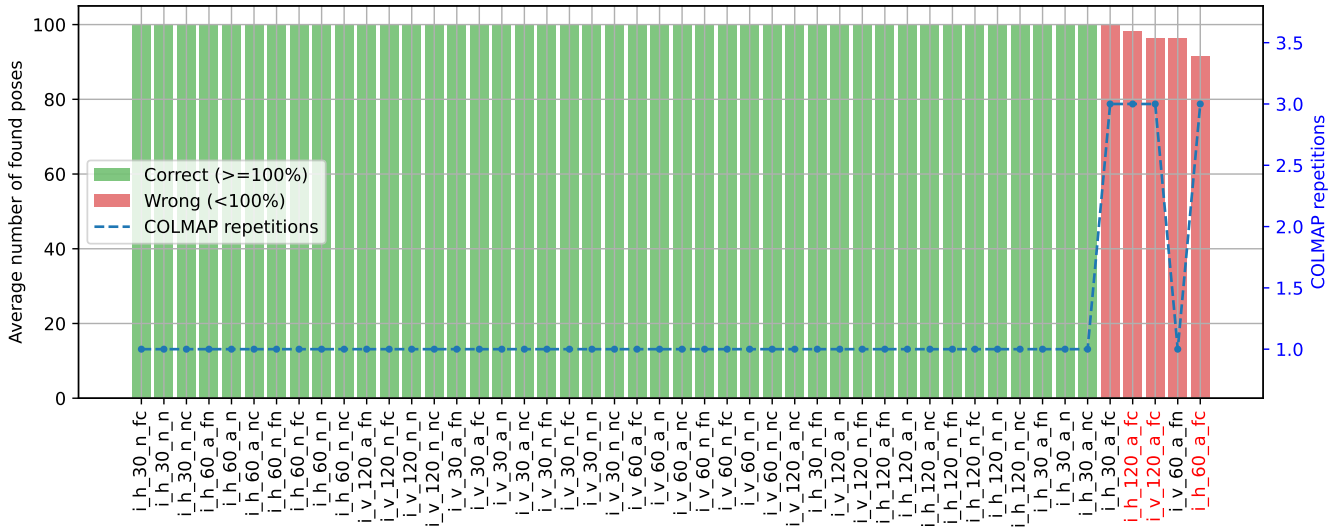


Figure 3: COLMAP number of repetitions after early repetition stage (in blue).

6 Lessons Learned

Our first experiments used most of Nerfstudio's default configurations. However, the initial results were not good since Nerfstudio uses COLMAP fixedly. For example, since COLMAP can estimate poses from videos or images by finding features, matching them, and applying a global bundle adjustment, Nerfstudio uses these estimated poses as inputs for the reconstruction models. The problem is that Nerfstudio always uses the first model of camera poses, but this is not the best one every time. When COLMAP does not find a good percentage of poses in one model, he generates additional models, which may be better than the first finding a higher amount of poses. Since training the reconstruction models uses only the frames in which a pose was successfully found, using as many poses as possible is desirable, so the final result has a good generalization between them. So, it is necessary to find the COLMAP-generated model of poses with the highest amount of them to use as input for Nerfacto.

Additionally, after finding the models of poses, Nerfstudio refines the camera intrinsics based on the first model found. But this refinement might be useless if this is not the best one. This refinement must be done using the best pose model since it will be used for Nerfacto. This is crucial since the closer to the ground truth, which is not known in this case, the poses are, the better the reconstruction will be.

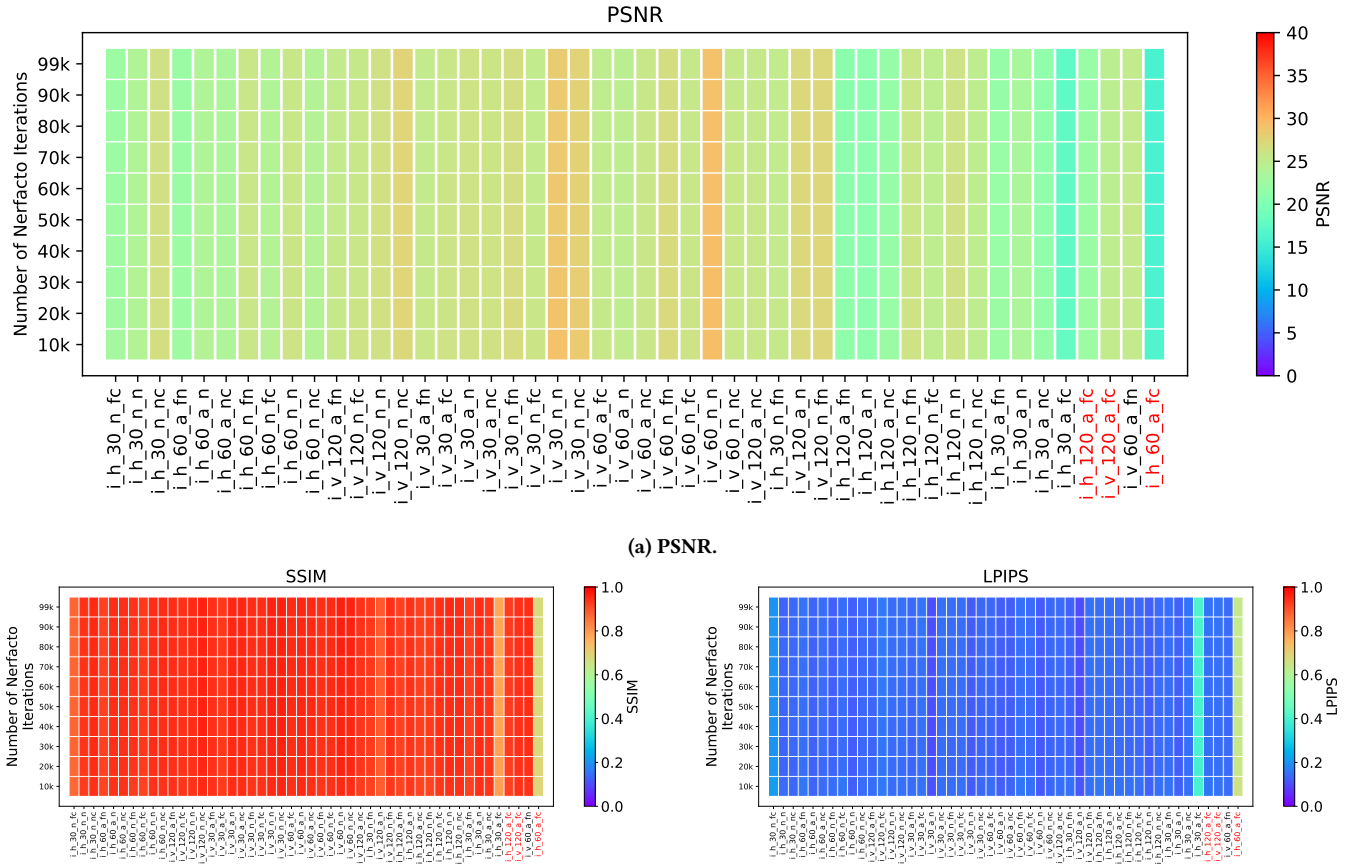
As proof of this, the Figure 7 compares the first results and the ones with the changes made concerning the percentage of poses found in the pilot study. As seen in it, the results are much better using the proposed approach.

As previously mentioned, using COLMAP only once does not ensure a good result since their results are slightly random. So, using it more than once is necessary to get the best result in the number of poses found. As seen in Figure 3, the first video with a red bar was processed by COLMAP thrice and was able to find 100% of the poses on the third try since the tick label in the figure is not red. This can increase the probability of getting good results, which is very good for companies wanting to sell a product consistently.

After gaining experience with the COLMAP and Nerfacto techniques, we may suggest good practices for 3D reconstructions based on videos or set images as input. For a reconstruction to have good coverage of the scene, it is necessary to have as many views of the environment as possible, so it is recommended to film from several different shooting angles. For the reconstructed scene to have a good generalization between the video or photo frames used, they must have a good amount of overlap between subsequent or nearby frames. This helps COLMAP to match the features found and determine the camera positions more correctly and helps Nerfacto to reconstruct the environment in spatial positions where no original frame existed. For a filming person walking at normal speed, extracting 3 frames per second from the videos ensures good overlapping between the frames used for the reconstruction. In addition to this, the movement of the camera must also be at the same walking speed. For photo datasets, the user must be aware that, depending on the separation of photos, the results may not be good enough.

We can also talk about filling the environment to be filmed: as COLMAP tries to find descriptors in the images, extensively monochrome walls must be filled with details (for example, pictures, decorations, etc.) to help COLMAP pre-process to extract as many poses as possible. Concerning illumination, the environment to be filmed or photographed must be properly lit, not too dark, and not with too much variation in light intensity.

In this work, we also used the Laplacian technique to choose frames with less motion blur, but even so, it is recommended that the movie be made as stable as possible, using stability tools and the highest camera's shutter speed so that the reconstruction is of good quality, given that the model will be based on the frame used as input. It should also be noted that the higher the resolution of the input frames used, the better the final result can be, provided that



(a) PSNR.

(b) SSIM.

(c) LPIPS.

Figure 4: Nerfacto's evaluation metrics.

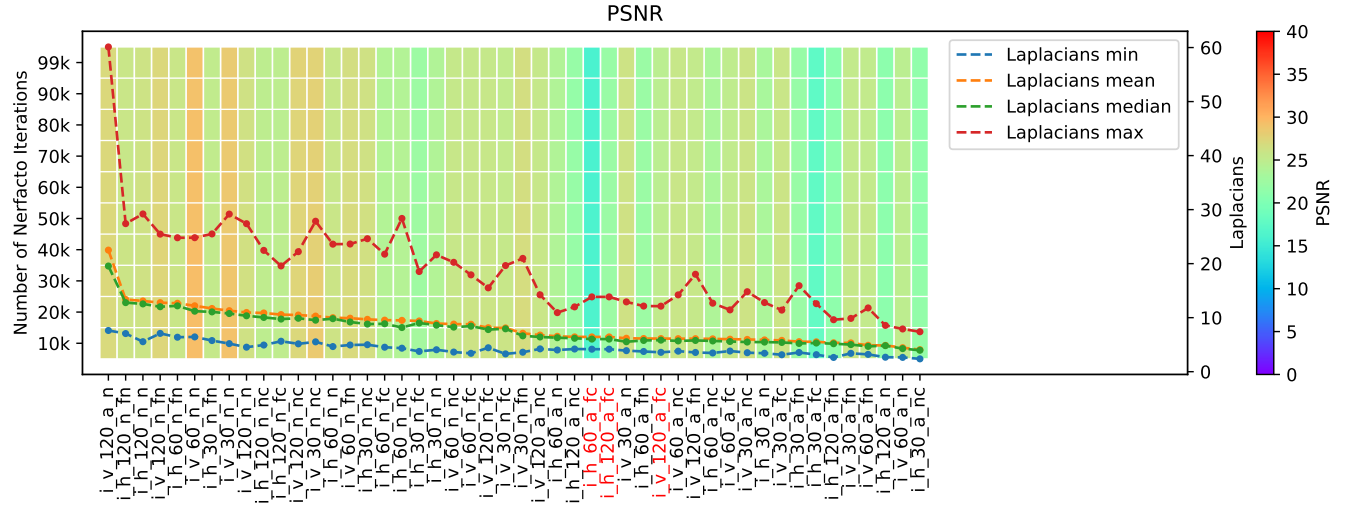


Figure 5: PSNR metrics ordered from left to right based on Laplacians' mean values. Laplacians' metrics are shown to enhance the visualization.

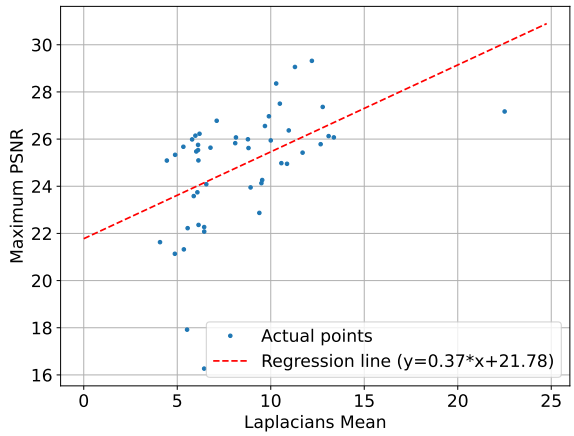


Figure 6: Linear regression made with Laplacians' mean values and maximum values of the PSNR for each video.

Nerfstudio is configured to use the full resolution of the image using the option `-downscale-factor 1` as the last one in the command `ns-train nerfacto`, where 1 means no downscale[2].

7 Limitations

Initially, eleven parameters were chosen for recording the videos, though the study began with more, including zoom options and camera perspective. However, the increase in parameters led to an exponential rise in the number of videos required, extending the processing and training time. As a result, only parameters significantly impacting the outcome were selected based on a prior pilot experiment.

The dataset consists of 48 videos from a single indoor environment to limit processing time, as adding more environments would require additional videos. This work focuses on indoor scenes, and the insights are relevant only to this context, with plans to investigate outdoor scenes in future work. More videos would improve the statistical reliability in 4.2, such as the relationship between PSNR and Laplacian values. The total processing time was approximately 60 hours (an average of 1:20 hours/min for each video). To address this limitation, a replication package with experiment details, metrics, and code is provided, encouraging further research due to the difficulty of finding large datasets for indoor and outdoor scenes.

Table 2 also shows some information about the usage of time and hardware. Reading these data, the user must have access to a GPU with at least 2.5Gb of VRAM to use this technology for the Nerfacto part, and 4.5Gb of VRAM for the COLMAP part. Without this hardware, the processing time becomes unpractically extended.

8 Conclusions

This paper fills a gap in NeRF research by optimizing video quality through systematic parameterization and standardized recording techniques. It moves beyond trial-and-error methods, offering insights and tools to improve NeRF-generated videos. Contributions include a list of tested parameters, a dataset of 48 videos with Nerfstudio data, a Jupyter Notebook for metrics, and guidelines for enhancing indoor scene creation.

Following the recommendations, practitioners can achieve higher-quality outputs with less rework, reduced training time, and fewer computational resources. Key factors affecting the quality of indoor scene reconstructions include proper illumination, appropriate luminance, and smooth camera movements. Filming in small indoor spaces is more challenging due to limited coverage, making it essential to capture most of the environment and provide a sufficient number of frames to COLMAP for accurate scene description.

In addition to that, neither all the users are able to film scenes with an equipment that can precisely record the poses of the cameras, which is another way to generate great 3D reconstructions. Using COLMAP instead is a way to ease the use of this technology for them. So, the proposed guide is necessary, since their steps influence the final results also due to COLMAP usage.

Our study used a single NeRF model, although other models are available within the Nerfstudio environment. Notably, the Task-Technology Fit model carries our approach by aligning task requirements - such as capturing high-quality visual data in indoor environments — with best video capture practices to maximize the output quality. A promising alternative is the Gaussian Splatting algorithm, including Nerfstudio's Splatfacto model; we plan to explore the potential of this technique in future work.

Acknowledgments

To the Applied Machine Learning Laboratory (LAMIA[3]) and the Center of Excellence in Artificial Intelligence (CEIA[4]) for fostering research and support, and to CNPq/MCTI/FNDCT (grant 408812/2021-4) and Fundação Araucária (grant PRD2023361000043) for the financial support.

References

[1] Beatriz Albuquerque, Antonio Cunha, Leonardo Souza, Sean Siqueira, and Rodrigo Santos. 2024. Generating and Reviewing Programming Codes with Large Language Models: A Systematic Mapping Study. In *Anais do XX Simpósio Brasileiro de Sistemas de Informação* (Juiz de Fora/MG). SBC, Porto Alegre, RS, Brasil. https://sol.sbc.org.br/index.php/sbsi/article/view/30892

[2] Tafnes Silva Barbosa. 2024. *Indoor Scene Parameterization.* https://github.com/TafnesBarbosa/Indoor-scene-parameterization/blob/main/readme.md.

[3] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. 2021. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision.* 5855–5864.

[4] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. 2022. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 5470–5479.

[5] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. 2023. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 19697–19705.

[6] Junyi Cao, Zhichao Li, Naiyan Wang, and Chao Ma. 2024. Lightning NeRF: Efficient Hybrid Scene Representation for Autonomous Driving.

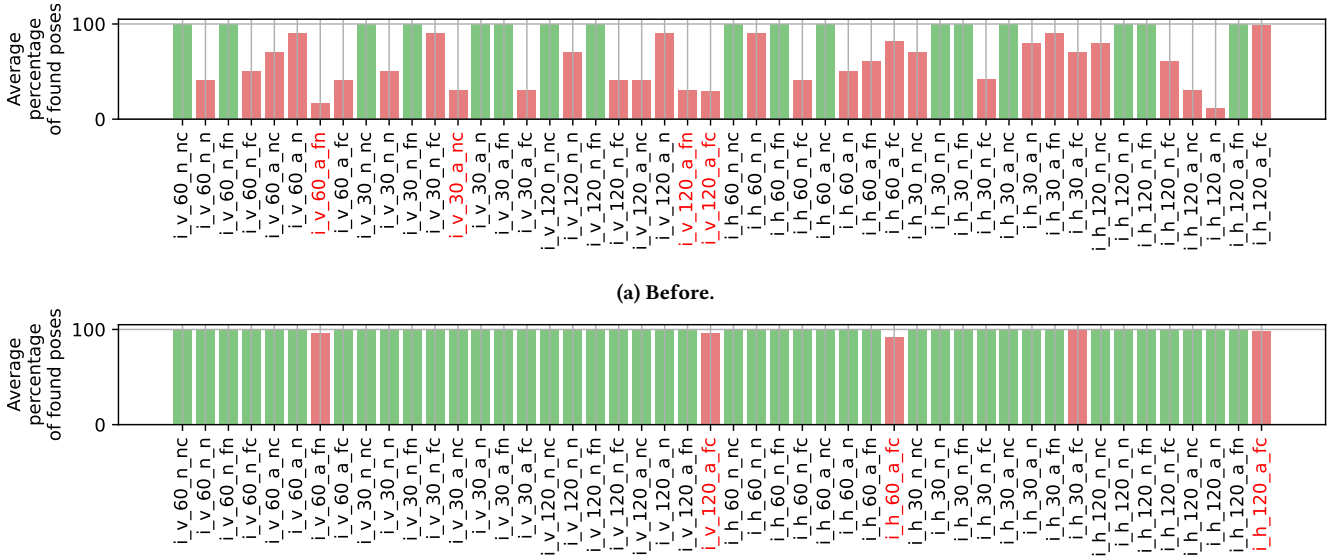
[7] Linsheng Chen, Guangrun Wang, Luchun Yuan, Keze Wang, Ken Deng, and Philip H.S. Torr. 2024. NeRF-VPT: Learning Novel View Representations with Neural Radiance Fields via View Prompt Tuning.

[8] Chenxi Lola Deng and Enzo Tartaglione. 2023. Compressing Explicit Voxel Grid Representations: fast NeRFs become also small. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV).* IEEE. https://doi.org/10.1109/wacv56688.2023.00129

[2] This considerably increases GPU consumption

[3] https://www.lamia-edu.com

[4] https://ceia.ufg.br



(a) Before.

(b) After.

Figure 7: Comparison between pilot studies made before and after changing the way of using COLMAP by Nerfstudio. Green bars stand for 100% of poses found. Red bars stand for less than 100%.

[9] Dale Goodhue. 1995. Understanding user evaluations of information systems. *Management Science* 41 (1995), 1827–1844. https://api.semanticscholar.org/CorpusID:153740964

[10] Dale Goodhue and Ronald L. Thompson. 1995. Task-Technology Fit and Individual Performance. *MIS Q.* 19 (1995), 213–236. https://api.semanticscholar.org/CorpusID:39660303

[11] Richard Hartley and Andrew Zisserman. 2003. *Multiple view geometry in computer vision.* Cambridge university press.

[12] Siming He, Zach Osman, and Pratik Chaudhari. 2024. From NeRFs to Gaussian Splats, and Back. *arXiv preprint arXiv:2405.09717* (2024).

[13] Inwoo Hwang, Junho Kim, and Young Min Kim. 2023. Ev-NeRF: Event Based Neural Radiance Field. arXiv:2206.12455 [cs.CV] https://arxiv.org/abs/2206.12455

[14] Kyungmin Jo, Gyumin Shim, Sanghun Jung, Soyoung Yang, and Jaegul Choo. 2021. CG-NeRF: Conditional Generative Neural Radiance Fields. arXiv:2112.03517 [cs.CV] https://arxiv.org/abs/2112.03517

[15] Petr Kellnhofer, Abhimitra Meka, Michael Stengel, Christian Theobalt, Marcus Magnor, Hans-Peter Seidel, and Tobias Ritschel. 2021. Neural Lumigraph Rendering. *IEEE Transactions on Visualization and Computer Graphics* (2021).

[16] Kai-En Lin, Lin Yen-Chen, Wei-Sheng Lai, Tsung-Yi Lin, Yi-Chang Shih, and Ravi Ramamoorthi. 2022. Vision Transformer for NeRF-Based View Synthesis from a Single Input Image. arXiv:2207.05736 [cs.CV] https://arxiv.org/abs/2207.05736

[17] Lingjie Liu, Michael Zollhöfer, and Andreas Geiger. 2020. Neural sparse voxel fields. In *Advances in Neural Information Processing Systems*, Vol. 33. 15651–15663.

[18] Yichen Liu, Benran Hu, Chi-Keung Tang, and Yu-Wing Tai. 2024. SANeRF-HQ: Segment Anything for NeRF in High Quality.

[19] Li Ma, Xiaoyu Li, Jing Liao, Qi Zhang, Xuan Wang, Jue Wang, and Pedro V Sander. 2022. Deblur-nerf: Neural radiance fields from blurry images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12861–12870.

[20] Davit Marikyan and Savvas Papagiannidis. 2023. Task-Technology Fit: A Review. In *TheoryHub Book*. TheoryHub.

[21] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. 2021. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. arXiv:2008.02268 [cs.CV] https://arxiv.org/abs/2008.02268

[22] Ben Mildenhall, Pratul P. Srinivasan, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis.

[23] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.

[24] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *CoRR* abs/2201.05989 (2022). arXiv:2201.05989 https://arxiv.org/abs/2201.05989

[25] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. 2021. KiloNeRF: Speeding up neural radiance fields with thousands of tiny MLPs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14335–14345.

[26] Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc Poliefeys, and Songyou Peng. 2024. NeRF On-the-go: Exploiting Uncertainty for Distractor-free NeRFs in the Wild.

[27] Azriel Rosenfeld. 1976. *Digital picture processing.* Academic press.

[28] Johannes Lutz Schönberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[29] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*.

[30] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. 2023. Nerfstudio: A Modular Framework for Neural Radiance Field Development. In *ACM SIGGRAPH 2023 Conference Proceedings (SIGGRAPH '23)*.

[31] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.

[32] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. 2021. NeRF-: Neural Radiance Fields Without Known Camera Parameters. *CoRR* abs/2102.07064 (2021). arXiv:2102.07064 https://arxiv.org/abs/2102.07064

[33] Guangtao Zhai and Xiongkuo Min. 2020. Perceptual image quality assessment: a survey. *Science China Information Sciences* 63 (2020), 1–52.

[34] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

[35] Dominik Zimmy, Joanna Waczynska, Tomasz Trzcinski, and Przemysław Spurek. 2024. Points2NeRF: Generating Neural Radiance Fields from 3D point cloud.