Simplifying NeRF: Creating an Intuitive Web-Based 3D Scene Interface

Eduard Aurelius von Briesen Brandenburg Neumark

May 3, 2024
Version: 0.1

Ludwig-Maximilians-Universität München,
Department of Mathematics, Informatics and Statistics,
Institute of Informatics

Munich Film School,
Chair for AI

Master's Thesis

Simplifying NeRF: Creating an Intuitive Web-Based 3D Scene Interface

for the degree of

Master of Science (M.Sc.)

in

Human-Computer-Interaction

by

Eduard Aurelius von Briesen Brandenburg Neumark

E.Briesen@campus.lmu.de

Supervisors	Prof. Dr. Sylvia Rothe and Christoph Johannes Weber
Submission Date	14. May 2024
Working Time	14. November 2023 - 14. May 2024

Abstract

The advent of Neural Radiance Fields (NeRF) has transformed the field of 3D scene modelling and rendering, offering photorealistic digital environments at relatively low effort and cost. Despite the technology's potential, the complexity of existing NeRF frameworks presents significant challenges to users without extensive technical expertise. This thesis presents the development of a web-based interface designed to simplify the creation of NeRFs, making them accessible to a broader audience, specifically targeting professionals from the film industry. The interface has been designed to facilitate the use of NeRFs by those without prior technical expertise, while still allowing users to leverage the full capabilities of the technology. The research employs a user-centred design approach, based on expert interviews to gauge user requirements and preferences. A subsequent user study evaluated the interface through qualitative and quantitative methods, in order to validate its effectiveness and accessibility. The contributions of this work not only enhance the user experience but also expand the creative possibilities for users across various use cases, thereby promoting broader adoption and innovation in NeRF-based content creation.

Abstract (German)

Neural Radiance Fields (NeRF) haben die Bereiche der 3D-Modellierung maßgeblich verändert. Sie ermöglichen die Erstellung fotorealistischer digitaler Szenen bei relativ geringem Aufwand und Kosten. Trotz des Potenzials der Technologie stellen die Komplexität der bestehenden NeRF-Frameworks sowie die erforderlichen technischen Kenntnisse für deren Nutzung eine Herausforderung für Benutzer ohne entsprechende Vorkenntnisse dar. In dieser Arbeit wird die Entwicklung einer webbasierten Oberfläche vorgestellt, die die Erstellung von NeRFs vereinfachen soll, um sie einem breiteren Publikum zugänglich zu machen, insbesondere Fachleuten aus der Filmindustrie. Die Schnittstelle wurde so konzipiert, dass sie auch Personen ohne technische Vorkenntnisse die Verwendung von NeRFs erleichtert und es den Nutzenden dennoch ermöglicht, die Kapazitäten der Technologie voll auszuschöpfen. Die Forschungsarbeit folgt einen nutzerzentrierten Designansatz, basierend auf Experten Interviews um die Anforderungen und Präferenzen der Nutzer zu ermitteln. In einer anschließenden Nutzerstudie wurde die Oberfläche mit Hilfe qualitativer und quantitativer Methoden bewertet, um ihre Wirksamkeit und Zugänglichkeit zu validieren. Die Beiträge dieser Arbeit verbessern nicht nur das Benutzererlebnis, sondern erweitern auch die kreativen Möglichkeiten für Benutzer in verschiedenen Anwendungsfällen und fördern so eine Verbreitung und Innovation bei der Erstellung von NeRF-basierten Inhalten.

Contents

1. Introduction **2**

1.1. Motivation 2

1.2. Research Objectives and Questions 2

1.3. Structure of the Thesis 3

2. Background **5**

2.1. View Synthesis 5

2.2. Neural Approaches to View Synthesis 6

2.3. Neural Radiance Fields for View Synthesis 7

3. Related Work **9**

3.1. Instant NGP 9

3.2. Nerfstudio 10

3.3. Luma AI 11

3.4. Volinga Suite 13

4. Methodology **15**

5. User Research **16**

5.1. Participant Selection Criteria 16

5.2. Interview Methodology 16

5.3. Key Findings 17

6. Application Design **20**

6.1. Design Process 20

6.2. User Interface Design 22

7. Technical Implementation **27**

7.1. System Architecture 27

7.2. Frontend Development 27

7.3. Backend Development 29

7.4. Challenges and Solutions 30

7.5. Future Directions 31

8. User Study and Evaluation **33**
8.1. Participant Selection Criteria . . . 33
8.2. Tasks Based Usability Test . . . 34
8.3. User Experience Questionnaire . . . 34
8.4. Follow-up Interview . . . 35
8.5. Data Analysis . . . 36

9. Results **37**
9.1. User Experience Questionnaire . . . 37
9.2. Findings from Qualitative User Testing . . . 39
9.3. Integration and Findings . . . 47

10. Discussion **50**
10.1. Interpretation of Results . . . 50
10.2. Implications for the Film and VFX Industry . . . 52
10.3. Limitations . . . 54
10.4. Integration of User Feedback . . . 55

11. Conclusion **57**
11.1. Key Findings . . . 57
11.2. Contributions to the Field . . . 57
11.3. Future Work . . . 58

A. Appendix **59**
A.1. Interview Questions . . . 59
A.2. User Study Questionnaire . . . 60
A.3. User Testing Results . . . 61
A.4. Prototype Documentation . . . 61

Bibliography **62**

List of Figures **66**

List of Tables **68**

1 Introduction

1.1 Motivation

View synthesis has long promised to revolutionize visual content creation in the film industry, but it wasn't until recent advancements in machine learning that this technology started to fulfill its potential. The advent of Neural Radiance Fields (NeRF) has been pivotal, offering unprecedented realism and detail in 3D scene modeling and rendering. This has significantly impacted various applications, from enhancing virtual reality experiences to the creation of photorealistic visual effects in movies.

Two prominent NeRF frameworks with user interfaces, namely Instant NGP [25] and Nerfstudio [38], have emerged as leaders in enabling users to explore and manipulate 3D scenes. These frameworks offer features such as real-time scene rendering, adjustable training parameters, and the creation of camera trajectories for video rendering.

Despite these advancements, the technical complexity of these frameworks frequently acts as a barrier to broader accessibility, indicating a need for improvements in user experience. These interfaces usually require a high degree of technical knowledge, as they are intended to supplement, rather than substitute, command-line interfaces. For activities like video data preprocessing, model training, and output rendering, users are required to navigate through terminal-based processes.

This complexity not only limits the potential user base to individuals with technical expertise but also hinders the creative and innovative application of NeRF technology across broader fields. Consequently, there exists a critical need to enhance the user experience and develop solutions that simplify the interaction with NeRF frameworks, making them more approachable and usable for a diverse range of users beyond the realm of technical specialists.

1.2 Research Objectives and Questions

This research aims to advance the capabilities and accessibility of NeRF frameworks, making them accessible to a broader audience and fostering innovation in 3D scene modeling and rendering. The study is guided by the following questions:

1. **Capabilities of Current NeRF Frameworks:** What are the existing interaction capabilities of NeRF frameworks, and how do they support various user groups in creating and manipulating 3D scenes?
2. **Enhancing Accessibility with a Web-Based Editor:** How can the development of a user-friendly web-based interface for NeRF improve its accessibility and simplify the creation and manipulation processes?
3. **Challenges in Developing Web-Based NeRF Tools:** What are the primary technical challenges and limitations associated with building a NeRF interface and how can these be overcome?

Scope of the Study This research focuses on the development and evaluation of a web-based interface for NeRF, aiming to improve its accessibility and usability. The study will concentrate on interface design and user interaction, without delving into the underlying algorithms of NeRF technology itself. It is delimited by its emphasis on interface design over algorithmic advancements in NeRF processing.

Significance of the Study By addressing the usability challenges of current NeRF frameworks, this research aims to make 3D scene modeling more accessible, fostering innovation and broadening the application of this technology across various fields. The development of a web-based interface could significantly lower the entry barrier to NeRF, enabling artists, designers, and educators to leverage this technology without requiring deep technical expertise.

1.3 Structure of the Thesis

This thesis is organized into several chapters, each focusing on a specific aspect of the research. The structure is outlined as follows:

2. **Background:** Provides a comprehensive overview of view synthesis, including traditional and neural approaches, and specifically introduces Neural Radiance Fields for view synthesis.
3. **Related Work:** Reviews existing solutions and technologies in the field, such as Instant NGP, NeRFStudio, Luma AI, and the Volinga Suite, discussing their features and limitations.

4. **Methodology:** Describes the methodological approach used to research and develop the web-based interface for NeRF.
5. **User Research:** Details the process and findings from user research, which informs the design and functionality of the developed interface.
6. **Application Design:** Covers the design process of the application, from initial sketches to the final design, emphasizing user interface and experience.
7. **Technical Implementation:** Explains the technical aspects of implementing the application, including system architecture and the integration of front-end and back-end components.
8. **User Study and Evaluation:** Discusses the setup, execution, and results of user studies conducted to evaluate the usability and effectiveness of the interface.
9. **Results:** Presents the findings from the user studies, analyzing the data collected through both quantitative and qualitative methods.
10. **Discussion:** Reflects on the results, discussing implications for the film and VFX industry, and addressing the limitations encountered.
11. **Conclusion:** Summarizes the key findings, contributions to the field, and suggests directions for future work.

2 Background

2.1 View Synthesis

View synthesis is a process used in computer graphics and computer vision that involves creating new, synthetic images of a scene from viewpoints that were not originally captured by a camera. This technique leverages existing images and often incorporates geometric information about the scene, enabling the generation of perspective-correct views from desired locations. The goal of view synthesis is to produce realistic and accurate representations of a three-dimensional scene from novel viewpoints, enhancing applications such as virtual reality (VR), augmented reality (AR), 3D television, and film production. The following sections provide an overview of traditional view synthesis techniques.

Image-Based Rendering Techniques Image-based rendering (IBR) techniques focus on synthesizing new views of a scene using pre-captured images, relying minimally on geometric models. The core premise of IBR is to directly utilize the radiance information captured in these images, manipulating it to generate new viewpoints without the need for detailed 3D reconstruction. This approach is particularly advantageous in scenarios where fast rendering is necessary or when accurate geometric data is unavailable. IBR methods are characterized by their ability to deliver photorealistic results, as they capture lighting, shadows, and reflections true to the original scene. These methods, such as those using light fields or lumigraphs, excel in applications where visual realism is critical, efficiently simulating the complexity of real-world lighting and textures [4, 6, 9, 14, 19].

Volumetric Methods Volumetric rendering techniques construct a three-dimensional volume of the scene, often represented as a grid of voxels. Each voxel contains data such as color and opacity, which contribute to the final image through a process akin to 3D texturing. Unlike surface-based modeling, which requires explicit surfaces, volumetric methods fill the entire data space, allowing for the handling of complex phenomena like fog, clouds, and fire, which do not have clear boundaries. This approach is advantageous when the scene involves intricate details and volumetric phenomena that traditional polygon-based rendering might struggle to capture accurately. Volumetric methods often employ techniques like voxel coloring to integrate multiple images into a cohesive 3D model [7, 34].

expand and include figures

2.2 Neural Approaches to View Synthesis

Neural approaches to view synthesis have significantly advanced the capabilities of generating realistic views from sparse data. These methods integrate deep learning techniques to enhance the synthesis process, offering substantial improvements over traditional geometric and image-based methods.

Deep Learning for View Synthesis Deep learning models, particularly Convolutional Neural Networks (CNNs), are employed to predict depth and color information from sparse sets of images. These models facilitate the synthesis of intermediate views by interpolating between captured viewpoints, efficiently handling scenarios with incomplete data where traditional methods would struggle [17, 22, 29].

Learning Volumetric Representations Recent advances in neural view synthesis have explored volumetric representations to effectively encode three-dimensional scene information. This approach leverages deep learning to construct volumetric grids or embeddings that capture both color and spatial data, allowing for dynamic and realistic rendering of new views. Such volumetric techniques represent a significant departure from traditional modeling by providing a framework where 3D features are embedded directly into the network, enabling sophisticated scene understanding and rendering without explicit geometric reconstruction [20, 35].

Multiplane Images and Scene Representation Multiplane Images (MPIs) represent a neural approach to view synthesis where scenes are decomposed into layers at different depths. Neural networks learn to create these layered representations from a limited number of views. The learned MPIs can then be efficiently re-rendered from new perspectives using traditional graphics techniques. This method provides a robust solution for generating photorealistic and geometrically coherent views by accommodating variations in scene complexity and handling occlusions effectively [23, 28, 31, 39].

End-to-End Learning End-to-end learning with deep neural networks has significantly simplified the view synthesis process by enabling a single network to handle multiple synthesis tasks concurrently. This approach reduces the complexity and potential failure points associated with multi-stage synthesis methods. By training on large datasets of posed imagery, these networks learn complex mappings from input images to synthesized views, efficiently handling challenging scenarios [5, 12].

2.3 Neural Radiance Fields for View Synthesis

Neural Radiance Fields (NeRF) offer a novel approach to view synthesis that significantly advances previous techniques. Utilizing a fully connected deep neural network, NeRF optimizes a continuous volumetric scene function from a sparse set of input views [24]. This method diverges from traditional discrete representations like voxel grids or mesh geometries, employing a more fluid and detailed scene representation.

Implementation NeRF synthesizes views by querying a 5D coordinate spacecomprising spatial locations (x, y, z) and viewing directions (θ, ϕ) along camera rays (see Fig. 2.1 a). Classic volume rendering techniques are then applied to convert the output densities and colors into images. At the heart of this process is a Multilayer Perceptron (MLP), a type of neural network characterized by layers of interconnected nodes which process input data sequentially from input to output. The MLP in NeRF takes sampled 3D points and corresponding viewing directions as input and outputs the color and volume density (see Fig. 2.1 b). This output is then composited into a 2D image using volume rendering techniques (see Fig. 2.1 c). Because this rendering function is differentiable, NeRF uses gradient descent to minimize the difference between synthesized images and observed ground truth, optimizing the neural representation of the scene (see Fig. 2.1 d).

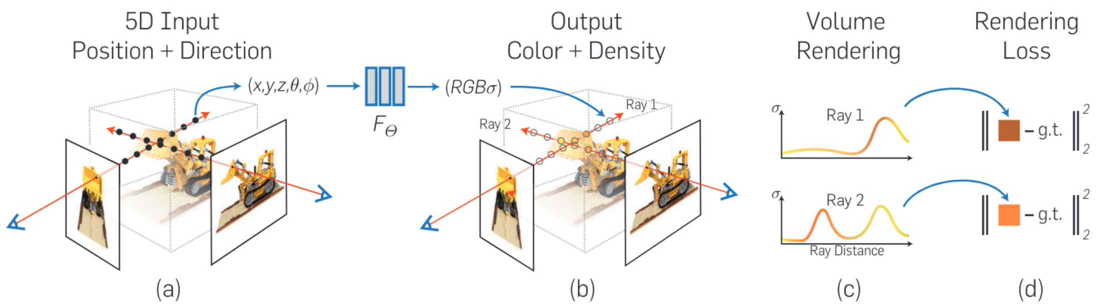


Fig. 2.1.: Overview of the NeRF scene representation and rendering procedure, illustrating the stages of sampling, neural processing, and image composition [24].

Advantages of NeRF Compared to the methodologies discussed above, NeRF offers several advantages. It surpasses the capabilities of mesh and voxel-based methods in rendering high-resolution details and handling intricate geometries and material properties. The continuous volumetric representation is not only capable of producing more photorealistic images but also remains highly efficient in memory usage. This efficiency facilitates handling complex real-world scenes without the prohibitive storage and computation costs associated with traditional 3D representations.

Extensions and Applications of NeRF The landscape of NeRF has been enriched by several innovative extensions that broaden the utility and interactivity of NeRF models. NeRF frameworks with user interfaces have emerged, enabling users to explore and manipulate 3D scenes [25, 38]. These frameworks offer features such as real-time scene rendering, adjustable training parameters, and the creation of camera trajectories for video rendering. Recent advancements have introduced text-based manipulation, allowing users to edit and transform scenes through natural language descriptions [1, 2, 15, 16, 45]. Another area of development is in color and appearance editing, where new methods enable adjustments of scene colors, consistent across varying views [46]. Additionally, some tools have integrated mesh editing capabilities, enhancing the geometric manipulation within NeRF-generated scenes [48]. These enhancements not only extend the functional range of NeRF but also significantly improve their application in creative industries.

In conclusion, NeRF's innovative use of neural networks for scene representation sets a new standard for photorealistic view synthesis, delivering high-quality results that are both computationally efficient and visually impressive.

Related Work 3

3.1 Instant NGP

Instant Neural Graphics Primitives (Instant NGP) [25], developed by NVIDIA, utilizes a multiresolution hash encoding that simplifies models while maintaining high performance and quality. Its GUI plays a crucial role in enhancing accessibility and functionality, making it an important advancement in neural graphics technology.

Simplified User Interaction The user interface of Instant NGP facilitates training and visualization of NeRFs (see Fig. 3.1). User can interactively explore 3D scenes in real-time and adjust parameters to achieve desired results.

Fig. 3.1.: Instant NGP's GUI rendering a NeRF scene, showing the 3D scene, camera path editor, and training parameters. [25].

VR Mode The VR mode of Instant NGP enhances user interaction by enabling immersive exploration of 3D environments in real-time. This feature is especially beneficial for professionals like architects and game developers, who gain from experiencing their virtual spaces as if they were real.

Camera Path Editor Instant NGP's camera path editor allows users to intuitively create and adjust camera trajectories, enhancing the creation of animations. This tool is essential for professionals in visualization and animation, providing precise control over camera movements for detailed and smooth presentations.

Limitations Despite advancements, Instant NGP's technical complexity and reliance on command-line interfaces for key operations remain significant barriers. These aspects limit its accessibility to those with specific technical skills and deter broader creative uses. The user experience still requires technical knowledge, underscoring the need for more intuitive interfaces that simplify interaction and expand its user base beyond technical specialists.

3.2 Nerfstudio

Nerfstudio [38] represents a significant advancement in making Neural Radiance Fields accessible to non-technical users. Its design focuses on modularity, ease of use, and integration capabilities, which are crucial for practical applications and academic research.

Modularity Nerfstudio is built on a modular framework that allows users to easily customize and extend their NeRF implementations. This modularity supports various input data formats, making it versatile for different real-world scenarios and setting it apart from Instant NGP. A wide range of existing methods are already well integrated into Nerfstudio, including Instant-GPT [25], their own Nerfacto [@26] method that combines various existing techniques, and several of the previously mentioned extensions [15, 16].

Real-Time Web Viewer One of the standout features of Nerfstudio is its real-time web viewer, which enables visualization of NeRF training and outputs directly through a web browser (see Fig. 3.2). This eliminates the need for high-end local GPU setups, such as in the case of Instant NGP, broadening the tool's accessibility [@27].

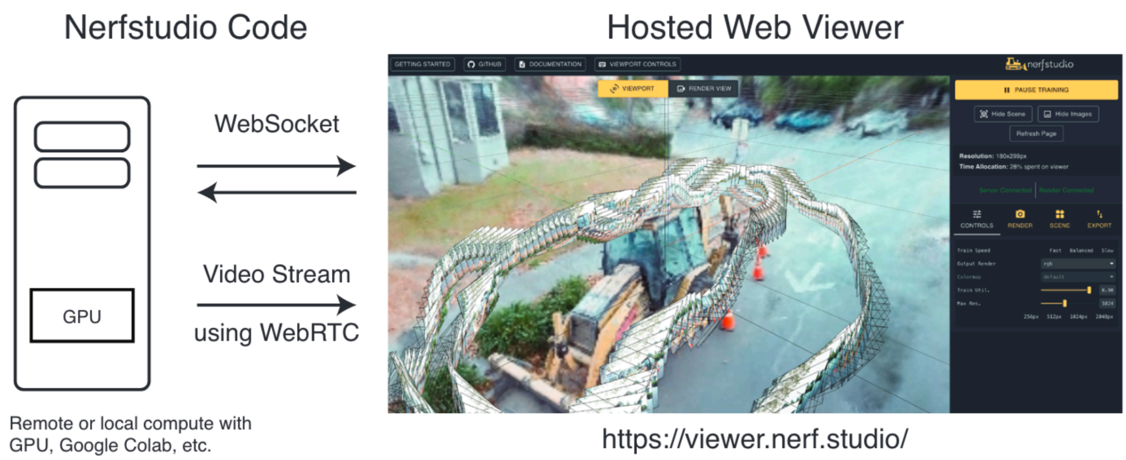


Fig. 3.2.: Nerfstudio's web viewer connected to a remote server running the training [38].

Flexibility of Data Handling Nerfstudio simplifies importing and exporting data, supporting a wide range of formats to accommodate various use cases. Users can easily import images and videos, including data from mobile capture apps such as Polycam [@30] and Record3D [@33]. Additionally, the framework supports exporting results in various formats such as videos, point clouds, and meshes. This flexibility allows users to integrate NeRF outputs into diverse creative and technical applications.

Community and Open-Source Contribution Being open-source, Nerfstudio encourages community-driven development and continuous improvement, facilitating updates that keep pace with the latest research and technological advances. This openness also allows users to adapt the tool to their specific needs.

Limitations Nerfstudio is a welcome advancement, improving on much of the features of Instant NGP. However, it still requires a certain level of technical knowledge to operate effectively, limiting its accessibility to non-technical users. The tool's primary interactions are still command-line based, posing a barrier to users who may prefer more intuitive graphical interfaces.

3.3 Luma AI

Luma AI [@21] is making Neural Radiance Fields accessible to non-technical users in a commercial space. This platform leverages augmented reality (AR) to guide users through the capture process, greatly simplifying the creation of NeRFs from everyday smartphones.

Guided Capture Process Luma AI utilizes AR to assist users in capturing images from optimal angles and distances, ensuring that the collected data is suitable for NeRF generation. This guided process reduces the complexities involved in capturing the necessary footage for effective NeRF modeling.

Cloud-Based NeRF Generation Once the footage is captured, it is automatically processed in Luma AIs cloud-based system to generate a NeRF, requiring no user input for configuration. This automation not only simplifies the user experience but also makes powerful 3D reconstruction technology readily accessible to a broad audience.

Viewing and Editing Created scenes can be viewed directly within the app or through a web browser (see Fig. 3.3). While the editing capabilities are limited, users can make basic adjustments, reshoot parts of the scene, and interact with the generated NeRF in an intuitive manner. The features here are similar to those offered in Nerfstudio's viewer, but with a more modern user-interface.

Fig. 3.3.: Luma AI Web Viewer in the camera path editor.

Export Capabilities Luma AI offers various export formats for the generated scenes, allowing users to utilize these outputs in different applications or platforms, further enhancing the utility of the captured NeRFs. They also provide a social platform for sharing and viewing NeRFs, fostering a community around the technology.

Limitations Despite its innovative approach, Luma AI's main limitation lies in the lack of user control over the NeRF training process. The automated system, while user-friendly, does not allow for adjustments in the training parameters or the refinement of the final model. This lack of control can result in less than optimal NeRF outputs for users who may require more precise or customized 3D representations. Additionally, the proprietary nature of LumaAI may be a concern for users who prefer the flexibility and transparency offered by open-source solutions, as it limits the ability to understand and modify the underlying processes.

3.4 Volinga Suite

The Volinga Suite [@44] aims to integrate Neural Radiance Fields into professional workflows. It simplifies its adoption by leveraging familiar platforms, thereby broadening the accessibility of NeRF to a wider range of users.

Unreal Engine Integration Volinga's integration as a plugin for Unreal Engine [@42] is a core feature that allows users to adopt NeRF seamlessly into existing pipelines (see Fig. 3.4). This integration is valuable for professionals already familiar with Unreal Engine, as it enables them to utilize advanced NeRF functionalities without additional training or significant adjustments to their current workflows.

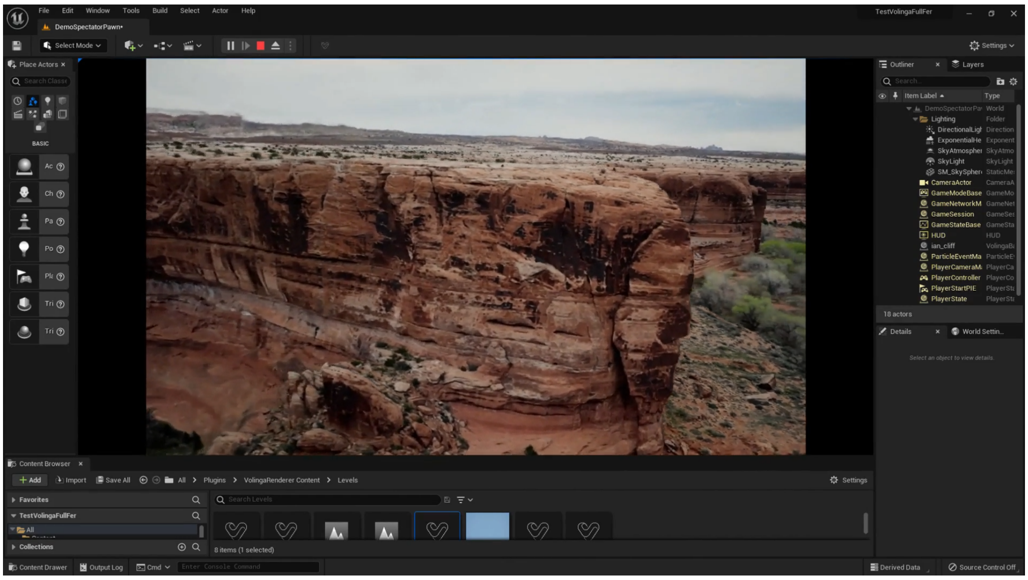


Fig. 3.4.: Volinga Suite's integrated viewer in Unreal Engine [@44].

Enhanced Control Over Training Volinga distinguishes itself from LumaAI by offering enhanced control over the NeRF training process. Users have the ability to locally adjust numerous parameters, which enables precise tuning of the model's performance to meet specific project requirements. This level of control is beneficial for projects where the quality of the NeRF output is critical.

Local and Remote Training Capabilities The Volinga Creator component supports both local and remote training of NeRF models. While remote training offers convenience and ease of access, local training provides advanced users with extensive configuration options and the ability to leverage powerful hardware, thereby maximizing the potential of NeRF technology under various usage scenarios.

Limitations Volinga Suite's reliance on Unreal Engine for its integrated viewer may limit users who are unfamiliar with or do not wish to use this specific platform. The web based viewer of Nerfstudio and Luma AI is more accessible in this regard. Although Volinga is actively contributing to the open-source projects Nerfstudio, it being proprietary may also deter users or organisations that prefer open-source solutions.

4 Methodology

This research was organized into three sequential phases: initial user research, prototype development, and user testing. Each phase was designed to inform and refine the subsequent stages, ensuring a systematic approach to developing a user-friendly NeRF interface. This iterative process aimed to align closely with user needs and feedback, creating a design that is both intuitive and functional.

Initial User Research The foundational stage of this research involved conducting a series of in-depth interviews to gather insights into the user experience of NeRF technology. The primary objective was to understand the varied challenges, needs, and preferences of users, ranging from novices to experts in NeRF model creation, particularly those with ties to the film industry. This exploratory phase was crucial for identifying the key features and improvements necessary for a more accessible and efficient NeRF interface. (see Chapter 5)

Design and Development Process The transition from initial user research findings to a functional prototype was a multi-step process focused on capturing user needs and translating them into a tangible design. This phase involved the creation of a user flow diagram, site map, wireframes, and a working prototype, each building upon the insights gained from the previous stage. (see Section 6.1)

User Study and Evaluation To evaluate the usability and overall utility of the developed prototype, a comprehensive user study was conducted. The primary aim of this study was to collect feedback on the prototype's user experience, identify any challenges participants encountered, and gauge their levels of satisfaction with the interface. Employing a mixed-methods approach allowed for a blend of quantitative and qualitative data collection and analysis, offering a multifaceted view of the prototype's performance in real-world tasks. (see Chapter 8)

5 User Research

5.1 Participant Selection Criteria

Participants were carefully selected based on their prior experience with NeRF technology and their connection to the film industry, resulting in a group of four experts. This selection ensured a diversity of perspectives, encompassing a broad spectrum of technical proficiency and practical applications of NeRF. By including individuals who have utilized NeRF in various capacities, the study aimed to uncover both the shared challenges faced by all users and the unique requirements of distinct user groups within the film industry.

5.2 Interview Methodology

The interviews were designed as semi-structured conversations, following a core set of prepared questions (see Appendix A.1) while also allowing for spontaneous discussions and additional queries. Conducted one-on-one (three online, one in person), these interviews facilitated a personalized dialogue with each participant, offering insights into individual experiences and perspectives. Although the interviews were prepared in English, all conversations were held in German to ensure comfort and clarity for participants, potentially leading to more candid and informative discussions.

The structured flow of questions began with learning about the participants' backgrounds and experiences with NeRF technology, gradually moving towards more detailed questions about their specific needs, challenges, and desired improvements in NeRF interfaces. Participants were also invited to propose features or functionalities they believed would enhance the usability and effectiveness of a NeRF interface for their professional or academic projects.

To ensure comprehensive analysis, interviews were recorded and transcribed with participants' consent, allowing for a detailed review and coding of the responses. This process enabled the identification of recurring themes, challenges, and preferences across the participant group, providing a solid foundation for the subsequent phases of prototype development and user testing. The insights gained from this initial research phase were instrumental in shaping the direction and focus of the interface design, ensuring it would effectively address the real-world needs of NeRF users.

5.3 Key Findings

NeRF in the Film Industry NeRF technology is being explored for various applications in the film industry, such as visual effects, virtual production, and pre-production location scouting. Despite its potential to simplify the creation of 3D scenes, current limitations in model quality, lack of editable models, and insufficient detail hinder its professional use. However, its capability for quick 3D scene captures offers significant benefits for pre-visualization and planning in the pre-production phase, although concerns about model scale accuracy for export remain. [**P2**, **P4**]

> *So in the planning phase I think [NeRF] was pretty well received, the set visit, the planning of the actual shoot, but the quality wasn't that convincing yet.*[1]
>
> — **Participant 4**

Optimizing Parameters and Workflow Creating NeRFs typically involves preprocessing input data, training models, and exporting outputs. Technical users emphasize the importance of parameter optimization in improving NeRF quality, with iterative training and results analysis being crucial parts of their workflow. Tools like TensorBoard [@40] are utilized for quantifying variations in training outcomes. [**P1**, **P3**]

> *Problems? Optimization, i.e. data sets. In training, I would say a big thing is optimization and parameterization, especially in Nerfstudio.*[2]
>
> — **Participant 1**

User Interface and Accessibility A consensus among users highlights the need for an intuitive, all-encompassing user interface that minimizes reliance on console commands. Features that allow users to visually navigate and control the NeRF creation pipeline, including real-time progress feedback and the ability to pause and adjust processes at any stage, are highly valued. [**P1**, **P2**, **P3**]

[1] Also in der Planungsphase kam [NeRF] glaube ich ziemlich gut an, Setbegehung, Planung vorne vom eigentlichen Dreh, aber die Qualität war halt noch nicht so überzeugend.

[2] Probleme? Optimisierung, also Datensätze. Beim Training würde ich sagen, eine große Sache ist die Optimierung und die Parameterisierung, vor allem in Nerfstudio.

The most important thing for me would be that I don't have to do anything in the console. In other words, that I can simply start the program and then do everything in the UI, upload it and then it will be executed somehow.[3]

— **Participant 2**

Comprehensive Error Handling and Visualization Effective error feedback and clear, informative visualization tools are critical for user satisfaction. Users express frustration with vague error messages and cumbersome command-line interactions for troubleshooting and adjustments. [**P2**, **P3**]

So if there is an error message, then it would be cool if they would somehow tell you more precisely what the error is, and not just any log.[4]

— **Participant 2**

File Management and Project Structure Efficient file and project management, with clear distinctions between different stages (preprocessing, training, rendering) and support for various input formats, is essential. Users discuss challenges with current tools regarding data organization, suggesting improvements for handling input data and managing projects. [**P1**, **P3**]

Integration and Export Options Strong integration capabilities with popular 3D and VFX software and flexible export options are desired. Users discuss the importance of being able to easily import NeRF-generated scenes into tools like Unreal Engine [@42] or Blender [@3] for further processing and use in production-quality projects. [**P1**, **P2**, **P4**]

Multi-Mode Operation The necessity for multi-mode operation in NeRF tool interfaces emerges as a significant insight, underlining the importance of accommodating a broad spectrum of users, from novices to experts. A simplified mode is envisioned to cater to beginners, offering an intuitive and streamlined workflow, whereas an advanced mode is tailored for experienced users requiring detailed control over the NeRF creation process. [**P1**, **P2**, **P3**, **P4**]

[3] Das Wichtigste wäre für mich, dass ich halt nichts in der Konsole machen muss. Also, dass ich einfach das Programm starte und dann alles in der UI machen kann, hochladen und dann auch irgendwie durchgeführt werde.

[4] Also wenn eine Fehlermeldung ist, dann wäre es halt cool, wenn die einem das irgendwie genauer sagen würden, was der Fehler ist, und dann nicht einfach so irgendein Log.

> *It just has to be understandable enough that film students aren't afraid of it. But still, not limit how much you adjust your parameters.*[5]
>
> — **Participant 1**

Summary

These findings highlight the demand for a NeRF tool interface that is user-friendly, versatile, and capable of supporting a wide range of workflows and user expertise levels. The ideal tool would combine intuitive project management and visualization features with powerful customization options, robust error handling and feedback mechanisms, and effective performance management capabilities.

[5]Es muss halt verständlich genug sein, dass Filmstudenten davon keine Angst haben. Aber trotzdem, [das Maß] wie sehr du deine Parameter ansetzen kannst [nicht] zu limitieren

Application Design

6

6.1 Design Process

The transition from initial user research findings to a functional prototype was a multi-step process focused on capturing user needs and translating them into a tangible design. This section outlines the journey from abstract requirements to the creation of the prototype, emphasizing the methodologies employed at each step.

From User Research to User Flow Diagram

Following the completion of initial user research, the first step involved modeling the key findings into a user flow diagram (Figure 6.1). This diagram served as a visual representation of the user's journey through the prototype, highlighting the main actions, decisions, and interactions users would have with the system.

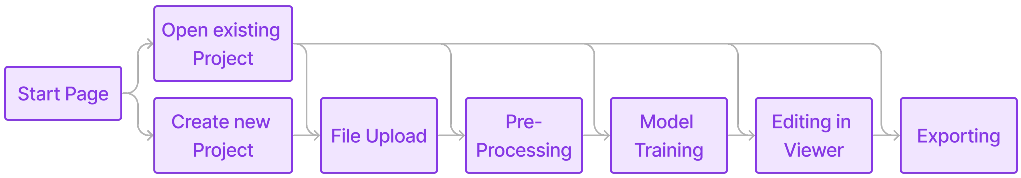


Fig. 6.1.: User flow diagram illustrating the key interactions for creating a NeRF model

The purpose of the user flow diagram was not only to map out the envisioned user experience but also to identify any potential bottlenecks or usability issues early in the design process.

Developing the Site Map

Building on the foundation laid by the user flow diagram, the next step was to expand this outline into a detailed site map. The site map provided a more comprehensive view of the prototype's structure, detailing the relationships between different pages and features. This excerpt from the site map (Figure 6.2) illustrates the level of detail and complexity involved in mapping out the prototype's user interactions.

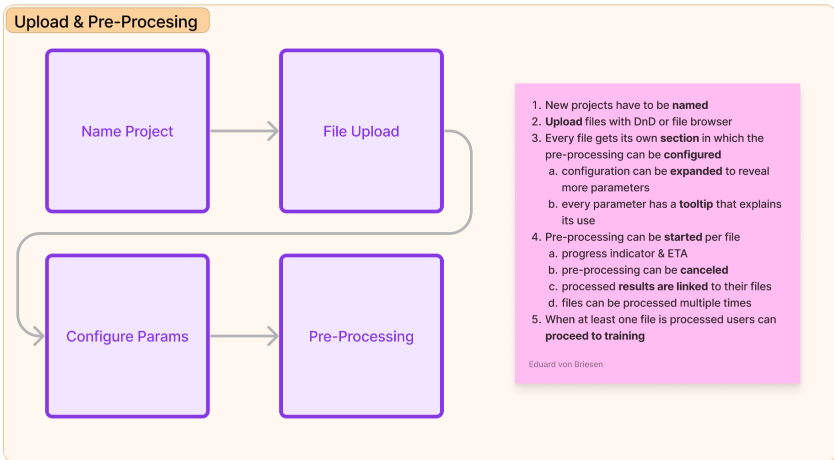


Fig. 6.2.: Excerpt of a view from the flow diagram with detailed interactions

This expanded view was instrumental in ensuring that the user flow remained intuitive across the broader system, facilitating easy navigation and a cohesive user experience.

Wireframing

With a solid understanding of the user flow and site structure, the focus shifted to wireframing. Initial wireframes were created to model the overall layout of the interface, providing a skeletal framework for the visual design. These wireframes were kept intentionally simple to prioritize structural and functional decisions over aesthetic considerations (see Fig. 6.3). At this stage, emphasis was placed on the placement of key elements, usability, and adherence to the user flow and site map.

Refinement Through Development

The transition from wireframes to a working prototype involved iterative refinement during the development phase. As the prototype took shape, initial designs were continuously evaluated and adjusted based on practical considerations and technical constraints. This phase allowed for a deeper exploration of interactions, animations, and the overall look and feel of the interface. It was during this time that the wireframes evolved into a more

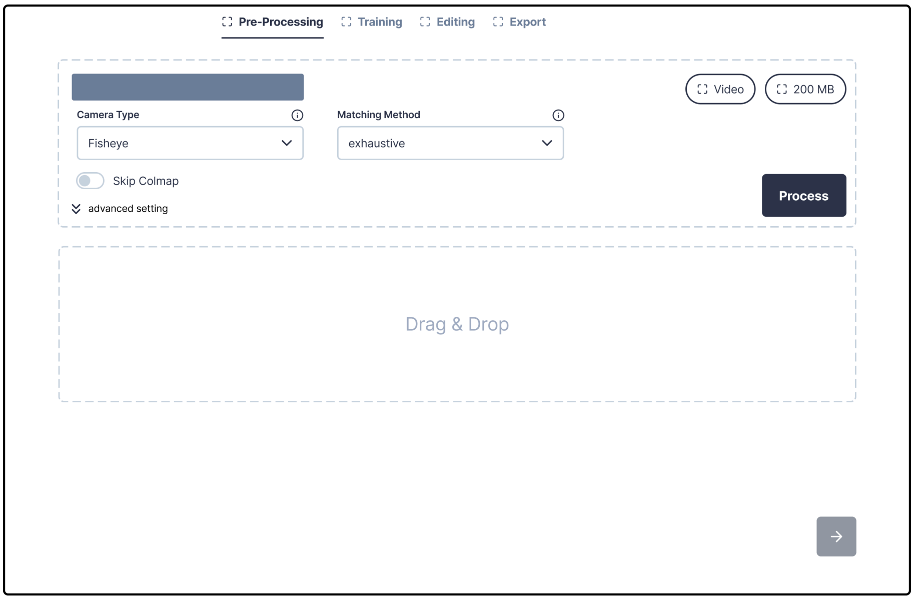


Fig. 6.3.: Wireframe of the pre-processing section, showing the layout and key interaction elements

detailed and user-friendly interface, with adjustments made as necessary to enhance usability and ensure a seamless user experience.

6.2 User Interface Design

The user interface was designed to be as simple as possible while still providing all necessary functionality. The design of the prototype can be broken down into two main parts: a dashboard that gives an overview of all projects and a project section that provides users with the tools to create and edit NeRF models.

Dashboard

The dashboard is the first view that users see when opening the application. It shows all previously created projects and allows users to create new ones. Projects are represented as cards, showing the project name, a preview of the provided input images (if present), and tags that indicate the current status of the project (see Fig. 6.4 1). An additional card is present through which users can create a new project, by providing a name (see Fig. 6.4 2). Projects can be opened by clicking a button on the respective card, when creating a new project, users are redirected to the project section.

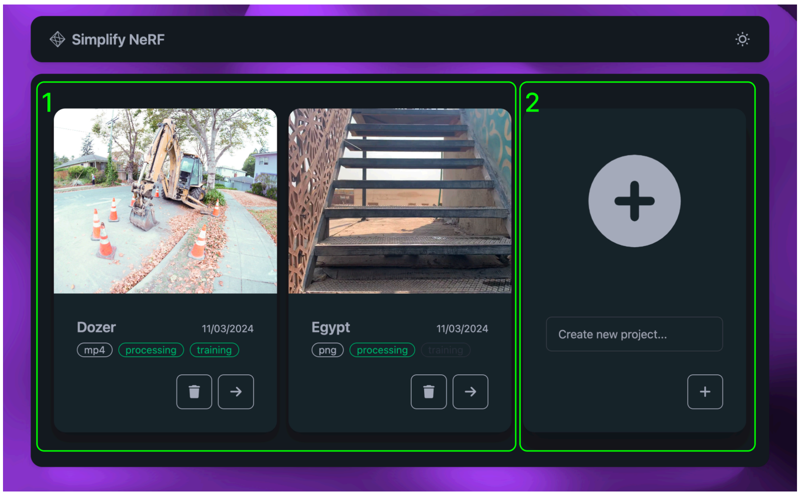


Fig. 6.4.: Dashboard overview, showing existing projects (1) and the option to create a new project (2)

Project Section

The project section is the core of the application, through which users can create and edit NeRF models. The section is divided into three parts: the input section, the training section, and the rendering section. Across all sections, users can track their progress through a progress bar at the top of the screen, that also enables easy navigation between the different sections (see Fig. 6.5 1).

Input Section

The input section combines the first few interactions, as mapped out in the user flow diagram. First, users are prompted to upload their input data, which can be done by dragging and dropping files into the browser window or by clicking a button to open a file dialog (see Fig. 6.5 2). Files can be either a set of images or a video, and there are some guardrails in place to ensure that the input data is valid. Once the input data is uploaded, it has to be processed before it can be used for training.

The pre-processing can be configured by the user, this includes parameters such as the lens type or matching method (see Fig. 6.5 3). Parameter input fields vary based on the type of input data and are only shown when relevant. Every parameter has a tooltip that explains its purpose, and advanced settings can be accessed by hovering over the question mark next to the parameter name. For advanced users, the settings can be further customized by activating the advanced settings using the toggle in the top right (see Fig. 6.6). Once the user is satisfied with the settings, they can start the pre-processing. Feedback on the progress of the pre-processing is given through a console that shows the output of the

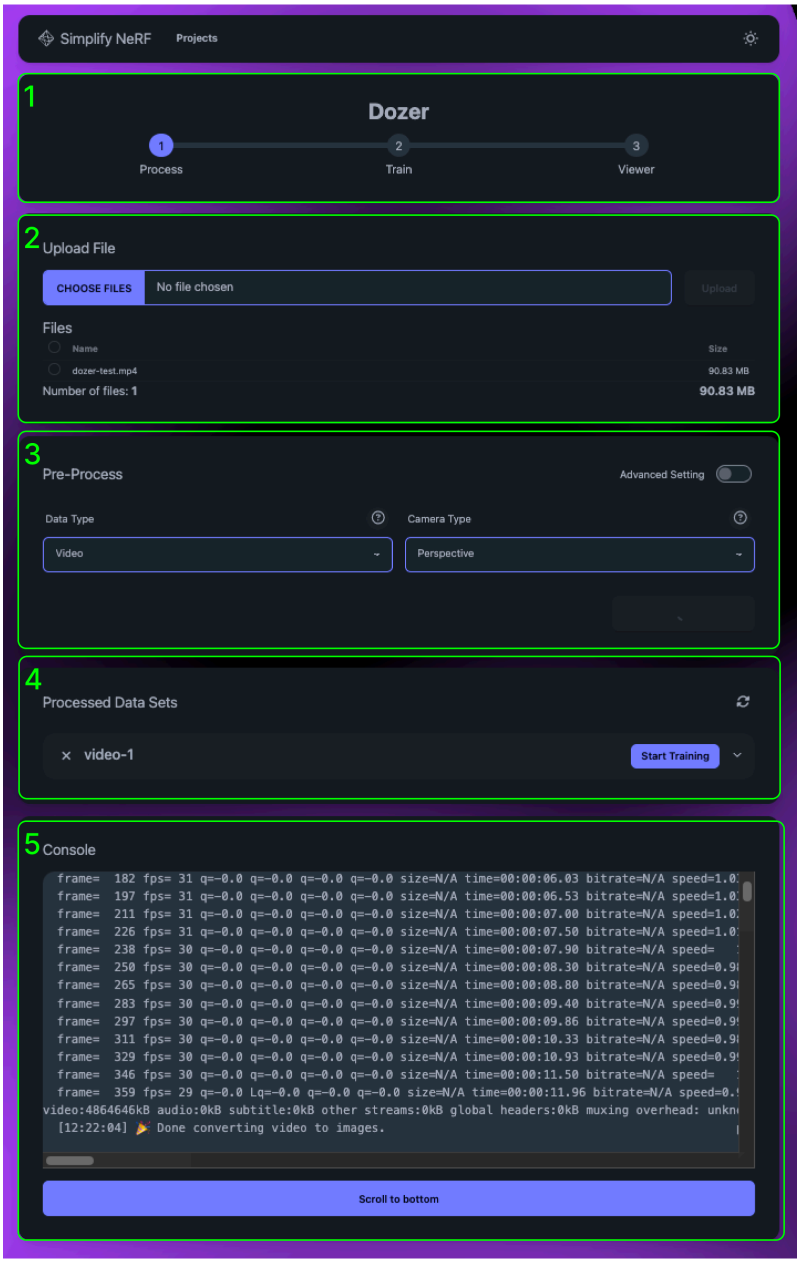


Fig. 6.5.: Pre-processing section, showing the progress bar (1), input data upload (2), parameter options (3), pre-processed data list (4), and console output (5)

process running on the server (see Fig. 6.5 5). When the pre-processing is finished, the user can move on to the training section.

In case the data was already pre-processed, a list is visible that shows all available pre-processed data, and the user can select one to use for training (see Fig. 6.5 4). Users can also inspect the configuration with which the data was pre-processed, and delete it if necessary.

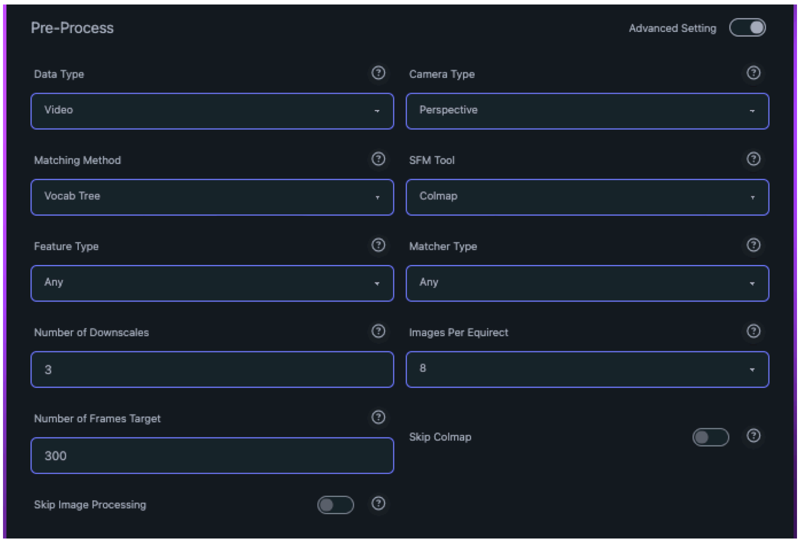


Fig. 6.6.: Advanced settings for pre-processing, showing additional parameters and options

Training Section

The training section is structured similarly to the input section, with a form that allows users to configure the training process (see Fig. 6.7 1), and a console that shows the output of the training process running on the server. When the user is satisfied with the configuration, they can start the training process.

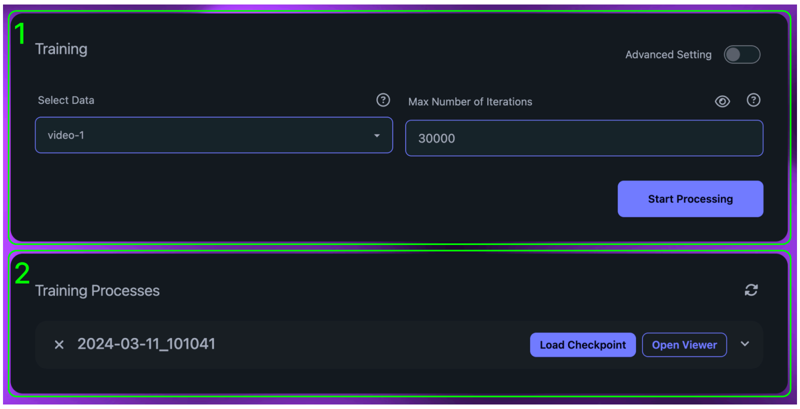


Fig. 6.7.: Training section, showing the training configuration form (1), and previous training runs (2)

Previous training runs are listed, and users can inspect the configuration with which the training was run, and delete it if necessary (see Fig. 6.5 2). The viewer can be opened to inspect the results of the training run, or an existing checkpoint can be selected to continue training from that point.

Viewer Section

The viewer section is the final step in the process and allows users to inspect their NeRF model, while it is still training or after the training has finished (see Fig. 6.8 1). At its core is the Nerfstudio Viewer that is integrated into the application. It provides users with all the functionality available in the standalone version, with a few integrations that simplify the render process. Instead of providing commands that have to be executed in the terminal, the rendering is started by clicking a button.

The renders are listed below the viewer, and once they finish processing, they can be downloaded to the user's machine (see Fig. 6.8 2). Due to the narrow layout of the page, the viewer can easily be opened in a new tab to provide a better viewing experience.

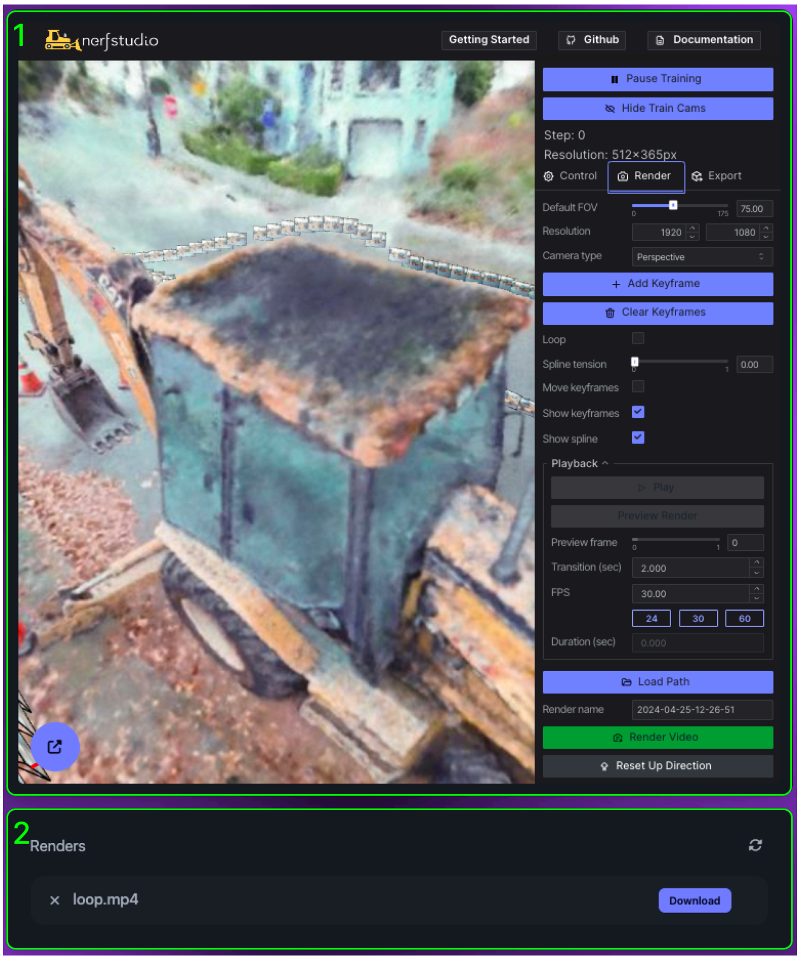


Fig. 6.8.: Viewer section, showing the Nerfstudio viewer (1) and list of rendered outputs (2)

7 Technical Implementation

7.1 System Architecture

The architecture follows a standard client-server pattern, with a service functioning as a wrapper for the nerfstudio CLI, and the client as a web application. The service is responsible for handling incoming requests from the client and translating them into commands that the nerfstudio CLI could understand. Requests from the client are sent to the service using HTTP, and, in the case of an asynchronous operation, the server updates the client using WebSockets (see Fig. 7.1).

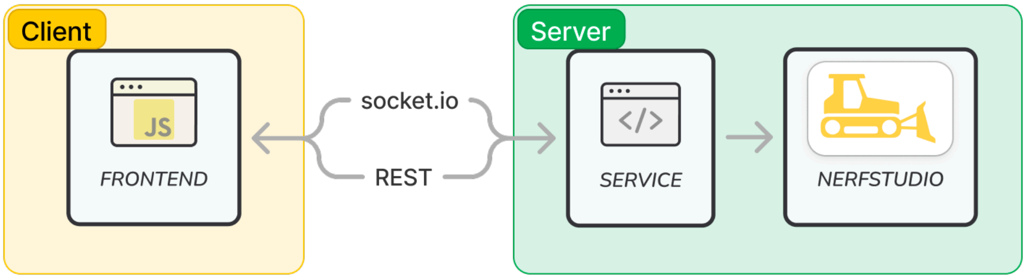


Fig. 7.1.: System Architecture Overview. The client communicates with the server, which in turn interacts with the nerfstudio CLI.

For the prototype, all components were composed into a single Docker [@10] container and deployed as a single unit. This approach leveraged the pre-configured container provided by nerfstudio and allowed for a quick and easy deployment of the prototype.

7.2 Frontend Development

The frontend is built with React [@32], chosen for its popularity and strong support in web application development. Vite [@43] serves as the build tool, offering a fast and efficient development experience. For styling, Tailwind CSS [@37], a utility-first CSS framework, provides a set of predefined classes to style components efficiently. Additionally, the daisyUI [@8] component library supplies pre-styled components, facilitating rapid UI construction.

Extensibility

Extensibility was a key consideration during the development of the frontend, as the underlying nerfstudio CLI is in itself extendable. All parameters for processing and training a NeRF model are configurable using JSON or strongly typed TypeScript objects (see Listing 7.1).

```
const stepsPerSave: NumberInput = {
  name: "stepsPerSave",
  label: "Steps Per Save",
  tooltip: "Number of steps between each save of the model.",
  inputType: "number",
  defaultValue: 1000,
};
```

Listing 7.1: Minimal parameter configuration for 'Steps Per Save' input.

Currently, the supported input types are number, select, and boolean, but it is easy to add new types by extending the configuration object. It is also possible to define dependent parameters that are only shown when a certain condition is met. Images to illustrate the effect of a parameter can be added as well, to provide additional context to the user.

Filters and presets are also configurable using arrays of the names of the parameters that should be included in the filter or preset. The full configuration can be found in the `frontend/src/config` folder in the codebase.

Nerfstudio Viewer Integration

The nerfstudio viewer is built using Viser [@27], a library for building 3D visualizations using python. This posed some limitations in integrating it into the frontend, as it is not easily possible to directly embed a python application into a web application. To work around this, the viewer was hosted by nerfstudio and embedded into the frontend using an iframe.

Some modifications could be made to the viewer to make it more suitable for embedding. This included some simple styling changes to make the viewer fit better into the frontend. More complex changes were also made to improve the user experience. The viewer contained several interactions where it was necessary for the user to copy console commands, to be used in the CLI. These interactions were replaced with buttons that send a request to the server to execute the command instead. These modifications were done at build-time of the container by applying a patch to the nerfstudio source code of the base image.

The main shortcoming of this approach is that the frontend is unaware of the state of the viewer, and cannot update the UI based on actions triggered in the viewer. It has to rely on polling the server to get the current state of rendering processes to give feedback to the user.

7.3 Backend Development

The backend was built using tRPC [@41], a framework for building type-safe APIs in TypeScript. This type-safety was useful in building the API, as nerfstudio endpoints require a specific set of parameters of various types, that could be easily defined using TypeScript and reused in the frontend.

A major concern for the backend was the handling of asynchronous operations. Processes can often take several minutes to complete, and the user should be able to see the progress of these operations. tRPC implement subscriptions, which allows the client to subscribe to an event and receive updates when that event occurs. This is used for any long-running operations, such as pre-processing or training a NeRF model (see Listing 7.2).

```
export const nerfstudioRouter = router({
  process: publicProcedure
    .input(
      z.object({
        project: z.string(),
        dataType: z.enum(["images", "video"]),
        ...
      }),
    )
    .subscription(({ input }) => {
      return observable<{message: string}>((emit) => {
        const args = ...
        const process = spawn("ns-process-data", args, {
          cwd: path.join(WORKSPACE, input.project),
        })
```

```
        process.stdout.on("data", (data: any) => {
          emit.next({message: data.toString()});
        });
      });
    })
});
```

Listing 7.2: Example tRPC endpoint for Pre-Processing returning a subscription.

Some additional endpoints were implemented using Express [@11]. This includes the file upload and render endpoints, due to limitations in tRPC discussed in Section 7.4.

All project related data is stored in a workspace directory, which is mounted as a volume in the docker container. This allows for the data to persist between container restarts, and for the user to access the data outside of the container.

7.4 Challenges and Solutions

Limitations of tRPC

Although tRPC is a powerful tool for building APIs, with tight integration into the frontend, it is not without its limitations. It lacks support for `multipart/form-data` file uploads, which are necessary for uploading images and videos. This was solved by implementing a separate endpoint using Express, which is used to upload files to the server.

The way tRPC handles subscriptions is also not ideal for long-running processes. If the client disconnects, the subscription is lost, and the client will lose the context for incoming events. A better solution could use a standard WebSocket connection, with the events containing the state necessary for the client to properly update the UI.

tRPC does not have a client for Python, which is needed when processes are started from the nerfstudio viewer. This required the implementation of another separate endpoint using Express, which is called by the viewer to render the NeRF model. All these limitations could be solved by using a more general-purpose framework, such as Express.

Working with the nerfstudio CLI

Building the prototype against the nerfstudio CLI offered a useful layer of abstraction. This enabled rapid development, without having to worry about the underlying implementation details of a huge codebase. However, this abstraction comes at the cost of flexibility and overall complexity of the system.

The CLI is built using tyro [@47], a tool for building command line interfaces in Python using configuration objects. Efforts were made to translate the configuration objects into TypeScript objects, that could be used throughout the project. This would allow for a more seamless integration of the CLI. Sadly, this was not possible, as the configuration objects are not easily serializable, and would require a lot of additional work to implement. Thus the implemented solution has to rely on manually constructing the commands based on documentation, which is error-prone and not very maintainable.

7.5 Future Directions

Improved CLI Integration

As outlined above, the approach of wrapping the nerfstudio CLI with a custom API has some limitations. A more integrated and robust solution would be built within the nerfstudio codebase itself. It might be able to reuse the existing configuration object, used for CLI, to build a REST API. This would allow for a more seamless integration of the CLI into the frontend, and would allow for more flexibility in the future. It would also reduce the overall complexity of the system, as it would not require a separate server to handle requests.

Improved Viewer Integration

The current approach of embedding the viewer in an iframe is not ideal. The nerfstudio viewer in version 0.3.4 and earlier, was built using React with a direct integration of Viser. With the first major release of nerfstudio, the viewer was rewritten in Python, and the React integration was removed.

Re-building this projects frontend with a framework such as Viser, would be very limiting in terms of features and flexibility. A better solution would be to integrate Viser into the frontend, akin to the original nerfstudio viewer. This integration would allow for a more seamless interaction between the frontend and the viewer, and improve the overall user experience. UI elements could be re-used to ensure a consistent look and feel, and the frontend could react to events triggered in the viewer.

8 User Study and Evaluation

To evaluate the usability and overall utility of the developed NeRF interface prototype, a comprehensive user study was conducted. The primary aim of this study was to collect feedback on the prototype's user experience, identify any usability challenges participants encountered, and assess their satisfaction with the interface. Employing a mixed-methods approach allowed for a blend of quantitative and qualitative data collection and analysis, providing a multifaceted view of the prototype's performance in real-world tasks.

Participants were given a series of tasks to complete within the prototype, followed by a User Experience Questionnaire (UEQ) and a follow-up interview to gather detailed feedback on their experiences. Of the ten participants, eight studies were conducted in person, while two were conducted remotely due to logistical constraints.

8.1 Participant Selection Criteria

Participants were selected similarly to the initial user research phase, focusing on individuals working in the film industry and possessing varying levels of experience with NeRF technology, from novices to experts. The resulting sample was diverse, with participants ranging in experience from no prior exposure to NeRF to expert-level proficiency (see Fig. 8.1).

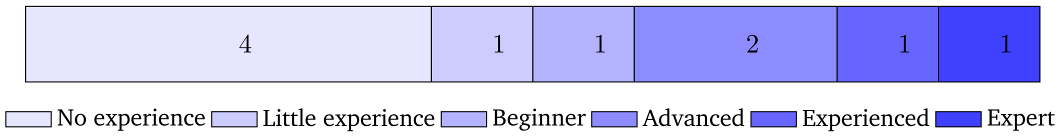


Fig. 8.1.: Level of experience with NeRF technology among participants.

Four of the participants have backgrounds in filmmaking, including directors, camera operators, and post-production specialists. The remaining six are software developers and researchers, selected for their expertise with NeRF or human-computer interaction.

8.2 Tasks Based Usability Test

The usability test was conducted in a controlled environment, where participants were asked to complete several tasks with the prototype. These tasks were designed to cover a range of functionalities and features of the prototype, representing a typical workflow for creating NeRF models. The tasks included:

1. Creating a new project.
2. Uploading a prepared video file.
3. Pre-processing the uploaded file to prepare it for training.
4. Switching to an existing project with pre-processed data.
5. Starting a NeRF training.
6. Creating a camera path in the viewer.
7. Exporting a video.

To maintain an appropriate time frame, none of the tasks required completion of a training process, and pre-processed data and pre-trained models were provided. On average, participants took 30 minutes to complete the tasks.

Participants were passively observed while working on their tasks to identify any problems or operational errors they encountered and to determine their overall performance. Additionally, the screen was recorded to capture participants' interactions with the prototype, allowing for a more detailed analysis of their behavior later on.

8.3 User Experience Questionnaire

After completing their tasks, participants were asked to fill out the User Experience Questionnaire (UEQ) [18], a standardized tool for assessing user experience. The UEQ was done immediately after the tasks to capture participants' immediate impressions while the experience was still fresh in their minds, and before any influence from the follow-up interview. Due to an oversight, the UEQ was not administered before the tasks for the studies conducted remotely.

The UEQ measures user experience across six dimensions:

- **Attractiveness** - the overall impression of the product.

- **Perspicuity** - the clarity and understandability of the product.
- **Efficiency** - the perceived effort required to use the product.
- **Dependability** - the perceived reliability and trustworthiness of the product.
- **Novelty** - the perceived originality and innovation of the product.
- **Stimulation** - the perceived level of excitement and engagement with the product.

This tool covers both classical usability goals (Efficiency, Perspicuity, Dependability) and user experience qualities (Novelty, Stimulation), with Attractiveness serving as a valence dimension not directly related to usability or user experience.

The questionnaire consists of 26 items, each represented by two terms of opposite meaning, with the order of the terms randomized for each item to avoid bias. Participants rate each item on a 7-point scale from -3 to +3, with 0 representing a neutral response. An example of the scale is as follows:

boring o o o o o o o *exciting*

Participants filled out the questionnaire digitally using a web-based survey tool [@36], which also included additional questions to gather demographic information and capture prior experience with NeRF and other 3D modeling tools.

8.4 Follow-up Interview

After completing the usability test, participants engaged in a short follow-up interview to provide more detailed feedback on their experience with the prototype. These interviews were semi-structured, following a predefined set of questions with room for participants to share their own thoughts and suggestions. Questions focused on participants' overall impressions of the prototype, usability challenges they encountered, and suggestions for improvement. The interview template is included in Appendix A.2.

8.5 Data Analysis

Both the video recordings of the usability test and the audio recordings of the follow-up interviews were analyzed to identify common themes and patterns in participant feedback. Videos were coded to highlight usability issues or challenges encountered during tasks, while interview transcripts were coded to extract detailed feedback and suggestions.

The UEQ data was analyzed using the standard procedure provided by the questionnaire's authors, which includes a spreadsheet tool that calculates all necessary values and visualizes the results.

In summary, this user study and evaluation was pivotal in validating the effectiveness of the NeRF interface prototype, uncovering valuable insights into its usability, and identifying opportunities for further refinement. The mixed-methods approach ensured a comprehensive assessment, capturing both tangible aspects of interface interaction and the subjective experiences of users, providing a solid foundation for subsequent development stages.

Results

9

This section presents the results of the user study conducted to evaluate the usability of the application. It first presents the results of the quantitative user experience questionnaire to gauge overall user experience. Then, it analyzes the results of the qualitative user testing to provide more detailed insights into the usability of the application. Finally, the results are integrated and discussed.

9.1 User Experience Questionnaire

Despite the relatively small sample size of 10 participants, the results of the User Experience Questionnaire (UEQ) provide a good overview of the overall user experience. The scores for the different scales of the UEQ are shown in Figure 9.1.

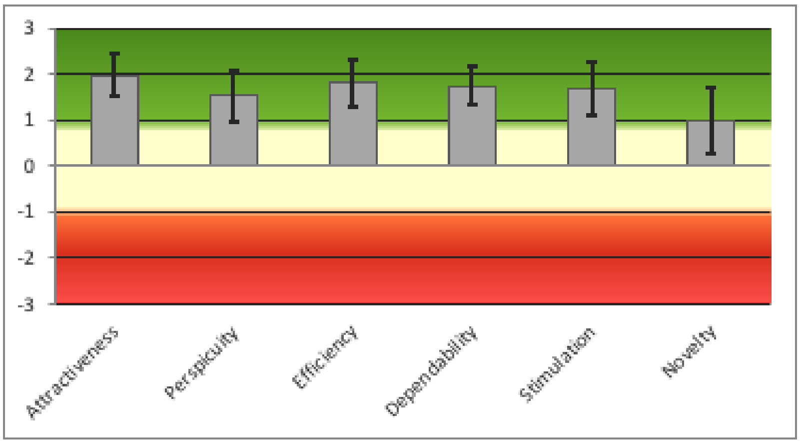


Fig. 9.1.: Results of the User Experience Questionnaire, showing the average scores for each scale and their 95 % confidence interval.

The *Novelty* scale scores the lowest with a value just below 1, while the *Attractiveness* scale has the highest score with 1.98. The other scales score between 1.5 and 1.8, indicating a generally positive user experience.

The UEQ scores can be benchmarked against the results of 452 other studies provided by the UEQ benchmark. The benchmarking results are shown in Figure 9.2, where *Attractiveness* and *Dependability* classify as *Excellent*, placing them in the top 10% of all studies.

Stimulation and *Efficiency* are classified as *Good*, while *Novelty* and *Perspicuity* only classify as *Above Average*.

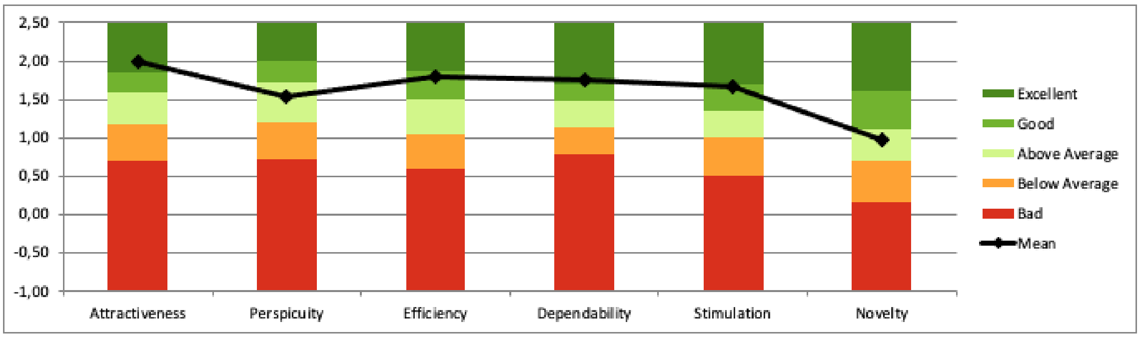


Fig. 9.2.: Benchmarking results of the User Experience Questionnaire, comparing the results of the study to a benchmark of 452 other studies.

The variance of the scores is relatively high, which could be attributed to the small sample size. The variance across the first five scales ranges from 0.49 to 0.92, only exceeding 1 for the *Novelty* scale with a value of 1.31.

Table 9.1 provides a summary of the UEQ results, including mean values, confidence intervals, and standard deviation. The standard deviation serves as a measure of agreement among participants, with lower values indicating higher agreement. Any value below 0.83 is considered *high agreement*, while values between 0.83 and 1.01 are considered *medium agreement* and values above 1.01 are considered *low agreement*. Of the six scales, three have a high agreement, two have a medium agreement and one has a low agreement.

Scale	Mean	Conf.	Conf. Int.	Std. Dev.	Agreement
Attractiveness	1.983	0.445	1.538 - 2.428	0.718	high
Perspicuity	1.525	0.573	0.952 - 2.098	0.924	medium
Efficiency	1.800	0.500	1.300 - 2.300	0.806	high
Dependability	1.750	0.432	1.318 - 2.182	0.691	high
Stimulation	1.675	0.594	1.081 - 2.269	0.958	medium
Novelty	0.975	0.710	0.265 - 1.685	1.145	low

Tab. 9.1.: Summary of the User Experience Questionnaire results

Summary

The analysis of the UEQ results provides insights into the application's user experience. High scores in *Attractiveness* and *Efficiency* scales, accompanied by high agreement among participants, suggest that these aspects of the application are both effective and appealing.

The consistency in these scores suggests that such attributes can be quickly gauged by users, even within the limited timeframe of the user study, leading to a uniformly positive perception.

The lower score and agreement on the *Novelty* scale indicate varied perceptions of the applications innovativeness. This could imply that while the application may not introduce new features or functionalities, it effectively repackages existing ones within an intuitive user interface. The moderate novelty score might reflect a use of familiar concepts and interactions, which can reduce the learning curve by leveraging well-understood mechanisms.

However, it is important to note that these findings are based on a relatively small sample size of 10 participants. While the results provide valuable insights, they should not be overvalued or considered definitive. Caution should be exercised in generalizing these findings without further validation from a larger, more diverse sample.

9.2 Findings from Qualitative User Testing

This section presents the findings from the user study conducted to evaluate the user interface of the application. The results are organized into categories reflecting overall impressions, learning curve and accessibility, identified issues, and recommendations for improvement, which now include feature requests.

Overall Impressions

Users generally find the interface user-friendly, effective, and aesthetically pleasing, enhancing the user experience across various functionalities.

User-Friendly Participants highlighted the intuitive layout and ease of navigation within the interface, appreciating how straightforward it was to perform tasks without prior training or extensive help. [**P1**, **P2**, **P4**, **P5**, **P7**, **P8**, **P10**]

> *So it's definitely a very nice tool, so it's pleasant to use.*[1]
>
> — **Participant 2**

[1] Also es ist auf jeden Fall sehr schönes Tool, also angenehm zum Nutzen.

Effective The effectiveness of the interface was noted in terms of its responsiveness and reliability. Users were satisfied with the speed at which the interface responded to commands and the consistency of its performance during tasks, which helped in building trust and reducing frustration during interactions. [**P2**, **P4**, **P7**, **P8**, **P10**]

> *Otherwise I thought it was super precise. I had the feeling that there wasn't much room for error.*[2]
>
> — **Participant 4**

Aesthetically Pleasing Many users commented on the visual appeal of the interface, mentioning the modern and clean aesthetic that made the experience more engaging. [**P7**, **P8**, **P10**]

> *In terms of design, I thought it was really cool, the way the colors are designed, even the animated background, I'm personally in favor of something like that.*[3]
>
> — **Participant 10**

These aspects collectively contribute to a positive user experience, making the application not only a tool that meets functional needs but also a pleasure to use, thereby encouraging repeated and prolonged engagement.

Learning Curve and Accessibility

Accessibility and ease of use are the main concerns of the application, driven to reduce the technical knowledge required to enable a broader user base. This section elaborates on the specific feedback received regarding the short comings and successes of the application in this regard.

Ease of Use As mentioned above, the overall user-friendliness was well received by participants across the board. Feedback from users indicates that the layout and workflow of the application facilitate quick learning and ease of use. [**P4**, **P5**, **P7**, **P8**, **P10**] The progress indicator, as shown in Figure 6.7, was particularly appreciated, as it helped users understand the current state of the application and what steps were required to complete a task. [**P5**, **P7**, **P8**, **P9**]

[2] Sonst fand ich super präzise. Also ich hatte das Gefühl, da war nicht so super viel Raum für Fehler.

[3] Designtechnisch fand ich es echt cool, also wie die Farben gestaltet sind, auch das mit dem animierten Hintergrund, bin ich persönlich auch für sowas.

> *I thought it was nice that it was split into three parts, so to speak, that this overview at the top said, okay, you're currently at Step 1 and then you know, okay, there are two more to go, so you can categorize it a bit.*[4]
>
> — **Participant 8**

Overwhelming Aspects Despite the general ease of use, some users expressed concerns about overwhelming aspects of the interface. The parameter settings often require a deep understanding of NeRF technology to understand their effects, but this did not hinder the overall usability of the application. [**P5**, **P7**] The viewer component, while powerful, presents a steep learning curve due to its complexity and the dense presentation of information and controls. New users, unfamiliar with NeRF and, in particular, Nerfstudio, might find this part of the application challenging to navigate initially. [**P2**]

> *But the [Nerfstudio Viewer] is quite complicated in itself, I think. It would be good to have some kind of explanation next to it about what you have to do or what you can do.*[5]
>
> — **Participant 2**

Technical Complexity Reducing the technical complexity inherent in NeRF applications is crucial for making the application more accessible to a broader audience. Based on the feedback received from users, who were previously unfamiliar with NeRF, the application requires users to have some level of prior knowledge. The guided experience and the reduced number of options to configure were appreciated by certain participants. It enabled them to quickly get started with the application without feeling overwhelmed. Most users felt confident that after a short learning phase, they would be able to use the application effectively. [**P2**, **P3**, **P4**, **P5**, **P6**, **P7**, **P8**] However, the feedback indicates a need for an onboarding process or tutorials to help new users overcome initial hurdles and gain confidence in using advanced features. In contrast, advanced users appreciated the depth of control and customization available. [**P9**, **P10**]

---

[4] Ich fand es schön, dass es quasi dreigeteilt war, dass diese Übersicht oben, dass da stand, okay, du befindest dich gerade bei Step 1 und dann weißt du, okay, es folgen noch zwei weitere, dann kann man es ein bisschen einordnen.

[5] Der [Nerfstudio Viewer] ist aber an sich recht kompliziert, finde ich. [Es wäre gut] daneben irgendwie eine Erklärung [zu haben], was man machen muss oder was man da machen kann.

> *I think I'll need a bit more time to get used to it, but I think that will happen quite quickly.*[6]
>
> — **Participant 2**

Project Management Novice users appreciated the similarities to existing project management of an already familiar tool from the film industry. [**P5**] For experienced users, the abstraction of tedious project management tasks, like file uploads and data organization, was well received. [**P1**] This allowed them to focus on the core tasks of training and rendering, without getting bogged down by administrative overhead.

> *Yes, it's definitely very self-explanatory, so how to manage projects, how to create projects, and so on. I'm familiar with this from almost all software that has something to do with film, video editing and Photoshop.*[7]
>
> — **Participant 5**

Technical Language One particularly insightful point of concern was the unfamiliar terminology used, as there exist differences between some terms in the context of NeRF and the film industry. [**P5**] This can lead to confusion and hinder the learning process, as users struggle to understand the meaning and implications of certain terms. Adjusting the language used in the application to specifically target the intended audience can help bridge this gap and further reduce the learning curve.

> *I think sometimes it could be less technical and a bit more translator if you make it movie-specific.*[8]
>
> — **Participant 5**

In summary, the application succeeds in providing new users with access to advanced NeRF tools, while also catering to the needs of experienced users by offering advanced options for customization.

[6] Ich glaube, ich bräuchte schon noch ein bisschen um mich rein zu arbeiten, aber ich glaube, dass das recht schnell gehen wird.
[7] Ja, ist auf jeden Fall sehr selbsterklärend, wie man also das Projektmanagement, wie man Projekte anlegt usw. Das kenne ich aus ungefähr allen Softwares, die jetzt was mit Film, mit Video Editing, Photoshop zu tun haben.
[8] Ich [finde] teilweise könnte es weniger technisch und ein bisschen mehr Übersetzter sein, wenn man das filmspezifisch macht.

Identified Issues

The following issues were identified during the user testing sessions, based on feedback from participants. Most of these issues became apparent while participants were performing tasks, and then were further discussed during the post-task interviews.

Unclear Navigation Users experienced confusion about navigating through tasks and understanding their progression within the application. This was particularly evident when having to switch to a different project. The intended way to navigate to the dashboard was to click on the logo, which was not immediately clear to all users. Some users succeeded in finding the dashboard after some exploration, while others used browser navigation to return to the dashboard. [**P1**, **P4**, **P5**, **P6**, **P8**, **P10**] Some participants also mentioned uncertainties about navigating between the different sections of a project. The progress indicator can be used to navigate between the different sections, but some users took some time to discover this feature. [**P9**]

> *What I would also like is for the home button to be marked with a little box when I went back, so that I know how to get back to Home.*[9]
>
> — **Participant 4**

Inconsistent Wording An issue that many participants stumbled upon was the inconsistent wording used specifically on the button to start the training process. In the version of the application used during the user study, the button was labeled "*Start Processing*", which was confusing to most users. [**P2**, **P3**, **P6**, **P7**, **P8**]

> *Then I would have looked somewhere on the button for 'Nerf Training' and it says 'Start Processing'.*[10]
>
> — **Participant 8**

Project Creation Similar to the wording issue, the process of creating a new project was not as intuitive as intended. Many participants encountered a small annoyance when they tried to click on the big plus icon placed in the center of the card, which was not clickable. (see Fig. 6.4) Although everyone quickly discovered the intended way to create a new project, this issue was mentioned by almost all participants. [**P3**, **P4**, **P5**, **P6**, **P7**, **P8**]

[9] Was ich noch gerne hätte, wäre, dass der Homebutton, als ich zurückgegangen bin, also mit einem kleinen Häuschen markiert ist, damit ich weiß, wie ich zurück zu Home komme.

[10] [Dann] hätte ich halt irgendwo auf dem Button nach 'Nerf Training' gesucht und da steht 'Start Processing'.

> *If [...] I have to create a new project and there is a big plus at the top and a small plus at the bottom, I would also tend to press the big plus at first.*[11]
>
> — **Participant 3**

File Upload A few participants experienced issues with the file upload process. [**P1**, **P4**, **P6**] The system model is quite complex, as it involves multiple steps and the feedback provided was lacking in some cases. Before pre-processing can start, the user has to select the input data to use, upload it to the server, wait for the upload to complete before continuing. Users were not always aware of the progress, and would sometimes continue to the next step before the upload was complete.

> *During the upload, I didn't check for a moment that it was still loading.*[12]
>
> — **Participant 6**

Viewer and Screen Layout Feedback indicates a need for a more flexible UI that adjusts to different screen sizes and supports fullscreen modes. [**P10**]

Awareness of Progress In some cases, users were not aware of what stage of the creation process they were in. This was somewhat caused by the inconsistent wording (see 9.2), but also by the high degree of similarity between the pre-processing and training steps. [**P2**]

> *It happened to me that I didn't really realize that I was already in the training step and not yet in this preprocessing step.*[13]
>
> — **Participant 2**

Console Output The console output was perceived differently by users. Some users appreciated the detailed feedback and insights provided, even with no technical background. [**P5**] Others that could not interpret the output, found it useless and distracting. [**P3**, **P4**]

> *That's quite exciting. When you see the console, you never get to see anything else.*[14]
>
> — **Participant 5**

[11] Wenn [...] ich ein Projekt neu erstell[en] soll und oben ist ein großes Plus und unten ist ein kleines Plus, würde jetzt erst mal auch dazu neigen, auf das große Plus zu drücken.

[12] Bei dem Upload, da habe ich ja auch, kurz nicht gecheckt, dass es noch lädt.

[13] Mir ist es passiert, dass ich ja nicht so richtig gecheckt habe, dass ich jetzt dann schon in dem Trainingsschritt war und noch nicht in diesem Preprocessing Schritt.

[14] Das ist ja ganz spannend. Wenn man [die Konsole] sieht, kriegt man ja sonst nie zu sehen.

Recommendations for Improvement

This section outlines some of the recommendations for improving the application based on the feedback received from participants. In contrast to the identified issues, these recommendations are more general and focus on improving the overall user experience.

Enhance Interface Usability To address the need for a more personalized and less distracting user interface, the animated background should be able to be disabled. Additionally, implementing a 'back-to-top' button would streamline navigation, enabling users to quickly return to the top of the page without manual scrolling. [**P10**]

Viewer Customization and Usability Enhancements Several enhancements to the viewer are recommended to improve its functionality and usability:

- **Multiple Viewing Modes**: Incorporate the ability to view the cameras perspective, as well as the scene from different angles, providing users with a more comprehensive understanding of the scene. [**P5**]
- **Advanced Camera Controls**: Introduce adjustable controls for camera speed, motion blur, and the ability to set up speed ramps, to provide users with greater creative control. [**P5**]
- **Camera and Shot Management**: Develop a more structured approach to camera and shot management within the application to support complex productions.Allow users to label and organize different camera shots and paths, making it easier to manage multiple views or scenes within a single project. [**P5**]
- **Measurement Tools**: Add tools that enable users to take precise measurements within the viewer, useful for detailed scene planning and analysis. [**P7**]

The participants requesting these features were exclusively users with a background in film production, who would include these features in their workflow.

Expand Advanced Settings and Benchmarks Users with an academic background in NeRF requested the following features, when evaluating the application for their workflow:

- **Support for different NeRF implementations**: Allow users to select different NeRF implementations, providing flexibility and customization options based on their requirements. [**P1**]

- **Advanced-Advanced Settings**: Provide deeper customization options for experienced users who require fine control over parameters, possibly allowing them to inject custom settings or scripts. [**P9**]
- **Integration of Benchmarking Tools**: Integrate tools like TensorBoard or WandDB to allow users to perform benchmarks, providing insights into model performance. [**P10**]

Tighter Integration of Viewer One participant suggested a tighter integration of the viewer with the rest of the application. Building the viewer controls directly into the interface, instead of inserting the viewer as a separate component, to provide a more seamless experience. [**P9**]

General Feedback concerning NeRF

The interview also provided insights into the participants' general perception of NeRF technology and its potential applications. This feedback is not directly related to the application but provides valuable insights into the participants' understanding and expectations of the technology. Especially participants with a background in film production and no prior experience with NeRF, provided valuable insights into the potential use cases and limitations of the technology.

Lack of Modification Options Creatives would like to see more options for modifying the scene, such as adding objects or changing the lighting. [**P3**]

Use Cases Some participants saw no potential use cases for the application or NeRF in general in their current workflow. [**P3**] Others cited the limitation to static scenes as a major drawback, as they require dynamic scenes for their projects. [**P4**, **P5**] But the possibility for stylized and 'impossible' camera movements was inspiring application in advertisement and music videos. [**P4**] Pre-visualization was also mentioned as a potential use case, as it allows for quick and easy scene creation that can be used for planning and storyboarding. [**P4**]

> *So what pops into my head are some aestheticized commercials where you can do something with a cool idea and try out your camera technique and make shots possible that you couldn't actually do.*[15]
>
> — **Participant 4**

Summary

The qualitative user testing revealed that users generally perceive the interface of the application as user-friendly, effective, and aesthetically pleasing. The intuitive layout and straightforward navigation were highlighted, allowing users to perform tasks efficiently without prior training.

However, certain aspects of the interface were seen as overwhelming, particularly the complex parameter settings and the viewer component, which pose a steep learning curve. These elements require a deeper understanding of NeRF technology, potentially limiting accessibility for newcomers. Additionally, issues such as unclear navigation and inconsistent wording were identified, suggesting that minor improvements in interface design could significantly enhance usability and user satisfaction.

9.3 Integration and Findings

This section integrates the results from the quantitative user experience questionnaire and the qualitative user testing to provide a comprehensive overview of the application's usability and user experience.

Detailed Insights from Integrated Data

Each UEQ dimension is discussed in the context of corresponding qualitative insights, providing a comprehensive understanding of the applications performance from the users' perspective.

[15] Also das was mir rein ploppt, sind irgendwelche ästhetisierten Werbungen, wo du mit einer coolen Idee was machen kannst und kameratechnisch sich ausprobieren kannst und Shots möglich machst, die du eigentlich nicht machen könntest.

Attractiveness The high scores in the *Attractiveness* dimension of the UEQ were strongly supported by qualitative feedback, with users frequently noting the visual design and aesthetic appeal of the application. Comments highlighted the modern interface and engaging visuals, which not only made the application appealing but also more enjoyable to use. This correlation suggests that the visual design is one of the application's strongest points.

Dependability *Dependability* also scored highly on the UEQ, which is consistent with user remarks about the applications reliability and stability. Participants expressed confidence in the application, noting that it consistently performed well during tasks without errors or interruptions. This reliability is crucial for user trust and satisfaction, indicating that the applications design is effectively meeting user needs.

Efficiency The *Efficiency* scores were moderate but generally positive, which aligns with mixed feedback from users about the application's performance. While most users found the application efficient in terms of task completion, some noted occasions where the interface could be streamlined to reduce points of confusion. Suggestions for improving workflow efficiency were common, pointing to potential areas for enhancing user experience.

Stimulation The *Stimulation* dimension received moderate scores, which could reflect user feedback on the task format within the application. Several users indicated that while the tasks were clearly defined and structured, this rigidity often did not leave much room for exploration or personalization, which could dampen their enthusiasm and engagement. This suggests that the application's functionality, while efficient, might benefit from integrating more flexible, explorative elements that allow users to engage more creatively with the tasks. Such changes could potentially increase user interest and satisfaction by making the experience not only functional but also more dynamically engaging.

Novelty The *Novelty* scores were the lowest among the dimensions. This could be interpreted in several ways. One possibility is that the application utilizes familiar usability patterns, which while contributing to ease of use, may not be perceived as innovative by users. Another interpretation is that, especially for expert users, the application does not enhance the underlying NeRF technology with new or unique features, only repackaging existing capabilities in a more accessible format. This suggests that while the application is effective and user-friendly, it may not offer significant novelty or innovation to users.

Perspicuity The *Perspicuity* dimension received mixed reviews, with some users reporting that the application was easy to understand and use right from the start, while others struggled with certain features or found the learning curve steeper than expected. This suggests a discrepancy in user experience that could be addressed by examining how information and options are presented in the interface, ensuring that they are accessible to all users regardless of their prior expertise.

Summary

The combined analysis of both quantitative and qualitative results offers nuanced insights into the user experience of the application. While high scores in dimensions like *Attractiveness*, *Efficiency*, and *Dependability* indicate positive user perceptions, caution is warranted in interpreting the findings, particularly regarding *Novelty* and *Stimulation*. The relatively lower score and agreement in the *Novelty* dimension suggest a potential lack of significant innovative features or the application's reliance on familiar usability patterns. Similarly, the moderate score in *Stimulation* could be attributed to the task format, potentially limiting user exploration. These insights highlight the complexity of user perceptions and the need for further investigation to understand underlying reasons fully.

Discussion 10

10.1 Interpretation of Results

This section interprets the key findings from the study in relation to the initially posed research questions, exploring the capabilities of current NeRF frameworks, the impact of a web-based editor on accessibility, and the challenges encountered in the development of such tools.

RQ1: Capabilities of Current NeRF Frameworks

What are the existing interaction capabilities of NeRF frameworks, and how do they support various user groups in creating and manipulating 3D scenes?

The current landscape of NeRF frameworks, notably Instant NGP and Nerfstudio, offers varied interaction capabilities tailored to different user needs in the creation and manipulation of 3D scenes. Nerfstudio stands out with its modular design and real-time web viewer, allowing users to interact dynamically with NeRF models. This design supports not only professionals who require robust tools for detailed manipulation but also provides a platform for future innovation and research (see Section 3.2).

However, findings from initial interviews indicate that these frameworks typically require substantial technical knowledge, which can limit their use to those with advanced skills, hindering broader adoption. In contrast, platforms like Luma AI prioritize accessibility through a more streamlined, user-friendly interface that automates many NeRF processes. This approach significantly lowers the entry barrier for non-technical users. Yet, this simplicity often comes at the cost of reduced control over the final outputs, which can be a critical drawback in professional settings where precise adjustments and customizations are crucial (see Section 5.3).

This analysis highlights a clear need for NeRF frameworks that not only simplify the user experience but also retain the advanced functionalities required by professionals, informing much of the design and development of the web-based editor in this study.

RQ2: Enhancing Accessibility with a Web-Based Editor

How can the development of a user-friendly web-based interface for NeRF improve its accessibility and simplify the creation and manipulation processes?

The development of the web-based editor for NeRF was specifically designed to enhance accessibility by minimizing the need for extensive technical knowledge. This approach has been effective in making NeRF technology more approachable, particularly noted in the positive feedback from the user study (see Section 9.2). Participants with less experience found the interface intuitive and easy to navigate, enabling them to create and manipulate 3D scenes with minimal guidance. Professional users appreciated the streamlined workflow and the depth of control offered by the tool.

However, despite the initial success in improving accessibility, the testing phase also revealed several usability issues that could impede user efficiency and satisfaction (see Section 9.2). These included challenges with interface navigation, wording, and responsiveness of the tool under various user actions. Such feedback underscores the importance of continuous user testing and iterative development to address these concerns.

This iterative approach is essential to refine the interface, ensuring it not only meets the basic needs of non-technical users but also scales to support the complex demands of professional environments. By continually enhancing the interface, the tool can better support a wide spectrum of users in efficiently working with NeRF.

RQ3: Challenges in Developing Web-Based NeRF Tools

What are the primary technical challenges and limitations associated with building a NeRF interface and how can these be overcome?

The technical development of a web-based NeRF editor introduces specific challenges related to remote operation, real-time feedback, and integration with existing frameworks:

Remote Operation NeRF processing is computationally intensive, typically requiring the processing to be offloaded to a server. The prototype leverages a client-server architecture to allow users to manage NeRF-related tasks through a web browser efficiently (see Section 7.1).

Real-Time Feedback To enhance user experience, immediate feedback on user actions is crucial. The prototype uses WebSockets to establish a real-time connection between the client and server, facilitating instant updates on running processes (see Section 5.3).

Integration with Existing Frameworks Integrating the web-based editor with Nerfstudio presents significant technical hurdles. Currently, server-side interactions are limited to command-line interface commands, which complicates maintainability and limits deeper integration. Direct integration could circumvent the need for makeshift solutions such as parsing log statements for progress tracking. On the client-side, Nerfstudio's viewer is hosted separately and embedded via an iframe in the web-based editor. This setup restricts direct communication between the interfaces, leading to a disjointed user experience. Additionally, the visual and functional disparities between the two interfaces can cause user confusion and inconsistency (see Section 7.4).

To overcome these challenges, a more integrated approach is necessary. Enhancing the server-side architecture to support direct API calls rather than relying on CLI can improve maintainability and functionality. On the client-side, integrating Nerfstudio's viewer directly into the web-based editor's framework would enable better synchronization and a unified user experience. This would not only streamline user interactions but also align the UI elements and workflow, reducing confusion and enhancing usability.

10.2 Implications for the Film and VFX Industry

The findings of this study indicate several implications for the film and VFX industry. They highlight the potential applications of NeRF technology in production workflows and the challenges that need to be addressed for wider adoption.

Pre-Visualization and Set Planning One significant outcome of the study was the potential of NeRF technology in the field of pre-visualisation and set planning. The ability to rapidly generate three-dimensional scenes from footage captured by a phone or camera can be of significant benefit to filmmakers at the early stages of production. A NeRF model can be employed to explore different camera angles and lighting conditions, thus aiding the planning of shots and general visual direction. This capability offers significant potential for time and resource savings, by providing a more accurate representation of the final scene before actual production begins. Furthermore, the generated scenes can be used for set planning purposes, thereby ensuring that the physical set allows for the desired camera angles and placements of actors, props, crew, and equipment.

Virtual Production Another potential application for NeRF technology is in virtual production, where virtual environments are used to replace physical sets or backgrounds. The ability to easily scan and generate 3D scenes makes it possible for filmmakers to create virtual environments with much less effort than traditional methods. This makes NeRF a valuable tool for low-budget productions or independent filmmakers who may not have access to expensive equipment or large teams.

Innovative Applications The capabilities of NeRF also prompted creative ideas among participants of the user study with a background in film. This included suggestions for virtual art exhibitions or highly stylised camera movements for advertisements. The enthusiasm of participants for exploring these new possibilities indicates that NeRF has the potential to drive innovation within the film and VFX industry, and that further exploration is warranted in this area.

Conerns and Limitations One significant concern raised by professionals in the film industry is the quality of NeRF models. These models often contain artifacts and noise that can be challenging to remove, particularly in complex scenes, rendering them unsuitable for high-quality production. During the user study, other limitations became apparent, including the static nature of the generated images and the quality of the output. The lack of a time dimension represents a significant limitation for the dynamic nature of film and video production. Although there have been some recent developments in the field of NeRF models that allow for the addition of motion [13], the results remain limited and do not match the fidelity of static NeRF models.

Accessibility and Adoption Although the tool developed in this study was the first touchpoint for some filmmakers with NeRF technology, they were able to use it with confidence and saw potential for its use in production. Their feedback on the potential usage of NeRFs did not focus on the accessibility of the tool, but rather on the specific features that would be required for certain use cases. While accessibility and ease of use are not the primary concerns for professionals in the film industry, they are nevertheless important factors for the wider adoption of NeRF technology. Tools targeting the film and VFX industry should aim to slot right into existing workflows without overwhelming users with new concepts or technical jargon. Industry-specific language and workflows should be considered in the design of such tools to ensure they are intuitive and easy to use for professionals. In light of these considerations, the potential for NeRF technology in the film and VFX industry is promising, and it could find a place in various stages of production, even with its current limitations.

10.3 Limitations

This study, while providing valuable insights into the usability and potential applications of a new NeRF web-based editor, is subject to several limitations that should be considered when interpreting the results.

Size and Diversity of User Group The user study was conducted with a small, diverse group of participants, which limits the generalizability of the quantitative results from the user experience questionnaire. The limited number of participants, their varied backgrounds, and different levels of NeRF expertise may skew the findings and are not necessarily representative of all potential users. Future studies could enhance validity by including a larger, more homogeneous group focusing on filmmakers to derive statistically significant insights and ensure broader applicability.

User Study Tasks The tasks assigned to participants were designed to be simple and guided, focusing primarily on basic functionalities. Participants were not required to engage with advanced features or given the freedom to explore the tool independently. This limited scope may not have fully tested the tool's capabilities, particularly in handling the complexity typical of real-world projects. Future research should incorporate a broader range of tasks, including those that engage with more advanced features and simulate real-world usage scenarios to provide a deeper understanding of the tools practical challenges and strengths.

Comparison with Existing Tools The study did not include a direct comparison to existing NeRF frameworks, such as those presented in Section 3. Such a comparison would have provided deeper insights into the strengths and limitations of existing tools, and how the prototype developed in this study compares to them. A comparative analysis would have also highlighted the unique features and functionalities of the web-based editor, and how it addresses the limitations of current NeRF frameworks.

Design Iterations The web-based editor was developed and evaluated within a single design cycle, potentially overlooking some usability issues and user feedback integration. Continuous iterative design, incorporating ongoing user feedback, is essential to refine the tool and address diverse user needs effectively. Subsequent iterations should aim to expand testing phases and include longer-term usage evaluations to thoroughly assess performance and user satisfaction over time.

10.4 Integration of User Feedback

Based on the issues identified during user testing, several adjustments were made to enhance the usability and intuitiveness of the application. They still need to be tested and refined further, but the changes are expected to address the primary concerns raised by users.

Improved Navigation To address the confusion in navigation to the dashboard, a dedicated button was added to the navigation bar. This feature has should improved the clarity of navigation to users significantly (see Fig. 10.1).

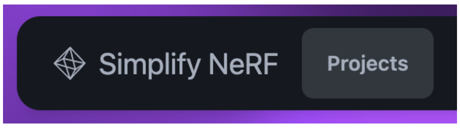


Fig. 10.1.: Dedicated Button for Dashboard Navigation

Clarified Wording The wording on the button to start the training process was changed from "*Start Processing*" to "*Start Training*", which aligns better with user expectations and reduces confusion.

Enhanced Project Creation The project creation process was moved into a modal dialog, which not only eliminates a point of confusion but also clarified the need to name projects before creation (see Fig. 10.2).

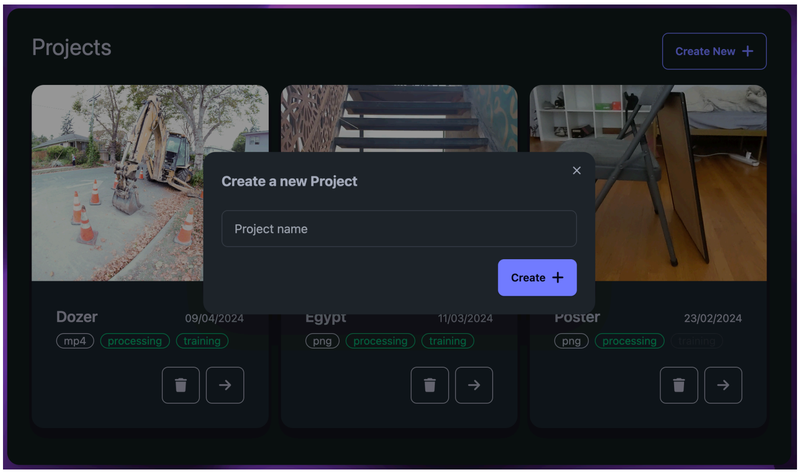
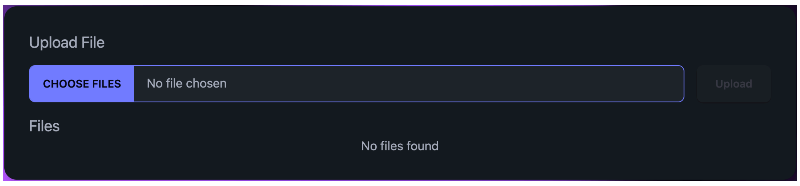
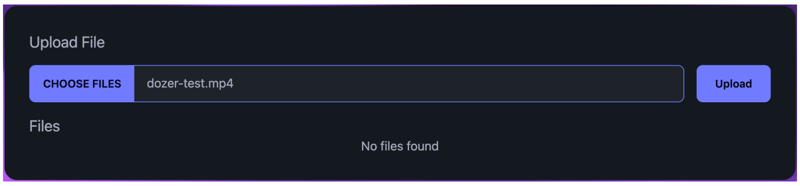


Fig. 10.2.: Modal Dialog for Project Creation

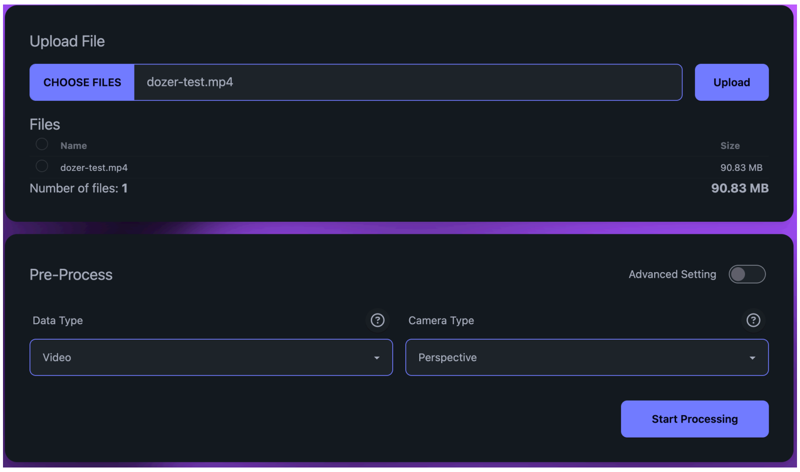
File Upload Improvements The file upload process was improved by adding some guardrails, to ensure user would not accidentally skip a step. The upload button starts out disabled, so that the only interactive element is the file-input field. Once a file is selected, the button becomes active, indicating to the user that they can proceed to upload their selected file. Only one the file is uploaded, the UI elements related to pre-processing appear, guiding the user through the next steps. This solution is likely to prevent many of the issues users encountered when uploading files during testing (see Fig. 10.3).



(a) Initial State



(b) File Selected



(c) File Uploaded

Fig. 10.3.: Improved File Upload Process

11 Conclusion

11.1 Key Findings

Neural Radiance Fields have emerged as a powerful technology for view synthesis, particularly for its ability to render photorealistic scenes with significant applicability in the film industry. A review of existing NeRF interfaces has identified significant usability issues that are impeding the adoption of this technology by filmmakers. Tools that are accessible to non-experts often lack the necessary control features, while those designed for experts present a steep learning curve due to their complexity.

To address these issues, this research developed a new NeRF interface that strikes a balance between simplicity and functional depth. The graphical user interface streamlines the NeRF model creation process by consolidating essential controls. The user study conducted to evaluate this interface demonstrated that it was well received by both novice and expert users. The simplicity and guided process were appreciated by novices, while the extensive control options and workflow enhancements were appreciated by experts. Nevertheless, the study also identified areas for potential enhancement, suggesting refinements to the interface's usability and the incorporation of additional features.

11.2 Contributions to the Field

The thesis makes a significant contribution to the field by conducting a general evaluation of NeRF tools. This highlights the practical limitations and needs associated with current NeRF technologies. A novel interface has been developed, designed to effectively balance ease of use for newcomers with the depth required by advanced users. Furthermore, the interface has been subjected to a rigorous evaluation through a user study, which has confirmed its enhanced usability and functionality.

11.3 Future Work

Although the new NeRF interface has demonstrated promising results, there is considerable scope for further development. The interface could be enhanced to support a wider range of Nerfstudio features and plugins, thereby broadening the tool's capabilities. Furthermore, it is necessary to enhance the system's robustness and scalability in order to accommodate an expanding user base and increasingly complex application scenarios. It is imperative that the framework be more deeply integrated with Nerfstudio if its full capabilities are to be fully leveraged, thereby enhancing both usability and functionality (see 7.5). Finally, the continuous refinement of the interface, guided by ongoing user feedback and extended user testing, including comparative studies, will provide deeper insights and encourage continual improvements. These future directions are of pivotal importance for maintaining the interface's relevance and effectiveness in adapting to the evolving demands of filmmakers and other creative professionals.

Appendix

A

bring both questionnaires into the same format

A.1 Interview Questions

Introduction

- Thank the interviewee for their participation.
- Briefly explain the purpose of the interview, which is to gather insights for the development of a NeRF interface.
- Assure the interviewee that their responses will be kept confidential.

Background

- Can you briefly describe your experience with NeRF or 3D modeling in general?

Needs and Challenges

- What specific tasks or objectives do you typically aim to achieve when working with NeRF models or 3D scenes?
- What are the main challenges or pain points you encounter when using current NeRF frameworks or interfaces?

Usability and Features

- In your opinion, what features or functionalities would make a NeRF interface most useful for your work or research?
- Are there any specific editing or manipulation tools you find lacking in current NeRF interfaces?
- How important is real-time visualization and interactivity in a NeRF interface for your needs?

Ease of Use

- How do you envision the ideal user interface for NeRF? What elements would make it easy to use, even for those with limited technical expertise?

- What level of technical knowledge or familiarity with 3D modeling should the ideal NeRF interface require from its users?

Integration and Compatibility

- Are there any other software tools or workflows you typically use alongside NeRF, and how important is it for a NeRF interface to integrate with these tools?
- Do you have any preferences regarding the file formats or data compatibility that the NeRF interface should support?

Feedback and Suggestions

- Are there any additional thoughts, suggestions, or requirements you would like to share regarding the development of a NeRF interface?
- Is there anything else you believe is essential for us to understand about your needs and expectations?

Closing

- Thank the interviewee for their time and valuable input.
- Provide contact information for any follow-up questions or clarifications.

A.2 User Study Questionnaire

Usability Experience

- Can you share your overall impressions of using the Simplify NeRF application?
- Were there any specific features or functionalities of the application that stood out to you positively? If so, why?
- On the other hand, were there any aspects of the application that you found challenging or frustrating? Please elaborate.

Task-Specific Feedback

- Were there any particular steps or actions within the tasks that you found confusing or unclear? If yes, could you describe them?
- Were there any features or functionalities you expected to find in the application that were missing? If so, what were they?

Suggestions for Improvement

- Based on your experience using the Simplify NeRF application, do you have any suggestions for improving its usability or functionality?
- Are there any specific changes or enhancements you would like to see in future versions of the application?
- How do you think the application could better meet your needs or expectations as a user?

Closing

- Is there anything else you would like to add or share about your experience with the Simplify NeRF application?
- Thank the participant for their time and valuable feedback. Offer contact information for any follow-up questions or clarifications.

A.3 User Testing Results

A.4 Prototype Documentation

Bibliography

[1] Chong Bao, Yinda Zhang, Bangbang Yang, et al. “SINE: Semantic-driven Image-based NeRF Editing with Prior-guided Editing Field”. In: *The IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*. 2023 (cit. on p. 8).

[2] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. “Text2LIVE: Text-Driven Layered Image and Video Editing”. In: *European Conference on Computer Vision*. Springer, 2022, pp. 707–723 (cit. on p. 8).

[4] Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, and Michael Cohen. “Unstructured lumigraph rendering”. In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. SIGGRAPH ’01. New York, NY, USA: Association for Computing Machinery, Aug. 1, 2001, pp. 425–432 (cit. on p. 5).

[5] Qifeng Chen and Vladlen Koltun. “Photographic Image Synthesis with Cascaded Refinement Networks”. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. ISSN: 2380-7504. Oct. 2017, pp. 1520–1529 (cit. on p. 7).

[6] Shenchang Eric Chen and Lance Williams. “View interpolation for image synthesis”. In: *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. SIGGRAPH ’93. New York, NY, USA: Association for Computing Machinery, Sept. 1, 1993, pp. 279–288 (cit. on p. 5).

[7] Brian Curless and Marc Levoy. “A volumetric method for building complex models from range images”. In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. SIGGRAPH ’96. New York, NY, USA: Association for Computing Machinery, Aug. 1, 1996, pp. 303–312 (cit. on p. 6).

[9] Paul E. Debevec, Camillo J. Taylor, and Jitendra Malik. “Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach”. In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. SIGGRAPH ’96. New York, NY, USA: Association for Computing Machinery, Aug. 1, 1996, pp. 11–20 (cit. on p. 5).

[12] John Flynn, Ivan Neulander, James Philbin, and Noah Snavely. “Deep Stereo: Learning to Predict New Views from the World’s Imagery”. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). ISSN: 1063-6919. June 2016, pp. 5515–5524 (cit. on p. 7).

[13] Sara Fridovich-Keil, Giacomo Meanti, Frederik Warburg, Benjamin Recht, and Angjoo Kanazawa. *K-Planes: Explicit Radiance Fields in Space, Time, and Appearance*. Mar. 24, 2023. arXiv: `2301.10241 [cs]` (cit. on p. 53).

[14] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. "The lumigraph". In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. SIGGRAPH '96. New York, NY, USA: Association for Computing Machinery, Aug. 1, 1996, pp. 43–54 (cit. on p. 5).

[15] Ayaan Haque, Matthew Tancik, Alexei Efros, Aleksander Holynski, and Angjoo Kanazawa. "Instruct-NeRF2NeRF: Editing 3D Scenes with Instructions". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023 (cit. on pp. 8, 10).

[16] Jan-Niklas Dihlmann, Andreas Engelhardt, and Hendrik P. A. Lensch. "SIGNeRF: Scene Integrated Generation for Neural Radiance Fields". In: *arXiv preprint arXiv:2401.01647* (2024) (cit. on pp. 8, 10).

[17] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. "Learning-based view synthesis for light field cameras". In: *ACM Transactions on Graphics* 35.6 (Nov. 11, 2016), pp. 1–10 (cit. on p. 6).

[18] Bettina Laugwitz, Theo Held, and Martin Schrepp. "Construction and Evaluation of a User Experience Questionnaire". In: USAB 2008. Vol. 5298. Nov. 20, 2008, pp. 63–76 (cit. on p. 34).

[19] Marc Levoy and Pat Hanrahan. "Light field rendering". In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. SIGGRAPH '96. New York, NY, USA: Association for Computing Machinery, Aug. 1, 1996, pp. 31–42 (cit. on p. 5).

[20] Stephen Lombardi, Tomas Simon, Jason Saragih, et al. "Neural volumes: learning dynamic renderable volumes from images". In: *ACM Transactions on Graphics* 38.4 (Aug. 31, 2019), pp. 1–14 (cit. on p. 6).

[22] Maxim Tatarchenko, Maxim Tatarchenko, Alexey Dosovitskiy, et al. "Single-view to Multi-view: Reconstructing Unseen Views with a Convolutional Network". In: *arXiv: Computer Vision and Pattern Recognition* (2015) (cit. on p. 6).

[23] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, et al. "Local light field fusion: practical view synthesis with prescriptive sampling guidelines". In: *ACM Transactions on Graphics* 38.4 (Aug. 31, 2019), pp. 1–14 (cit. on p. 6).

[24] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, et al. "NeRF: representing scenes as neural radiance fields for view synthesis". In: *Communications of the ACM* 65.1 (Dec. 17, 2021), pp. 99–106 (cit. on p. 7).

[25] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. "Instant Neural Graphics Primitives with a Multiresolution Hash Encoding". In: *ACM Trans. Graph.* 41.4 (July 2022), 102:1–102:15 (cit. on pp. 2, 8–10).

[28] Eric Penner and Li Zhang. "Soft 3D reconstruction for view synthesis". In: *ACM Transactions on Graphics* 36.6 (Dec. 31, 2017), pp. 1–11 (cit. on p. 6).

[29] Peter Hedman, Peter Hedman, Julien Philip, et al. "Deep blending for free-viewpoint image-based rendering". In: *ACM Transactions on Graphics* (2019) (cit. on p. 6).

[31] Pratul P. Srinivasan, Pratul P. Srinivasan, Richard Tucker, et al. "Pushing the Boundaries of View Extrapolation With Multiplane Images". In: (2019) (cit. on p. 6).

[34] Steven M. Seitz and Charles R. Dyer. “Photorealistic Scene Reconstruction by Voxel Coloring”. In: *International Journal of Computer Vision* 35.2 (Nov. 1, 1999), pp. 151–173 (cit. on p. 6).

[35] Vincent Sitzmann, Justus Thies, Felix Heide, et al. “DeepVoxels: Learning Persistent 3D Feature Embeddings”. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). ISSN: 2575-7075. June 2019, pp. 2432–2441 (cit. on p. 6).

[38] Matthew Tancik, Ethan Weber, Evonne Ng, et al. “Nerfstudio: A Modular Framework for Neural Radiance Field Development”. In: *ACM SIGGRAPH 2023 Conference Proceedings*. SIGGRAPH ’23. 2023 (cit. on pp. 2, 8, 10, 11).

[39] Tao Zhou, Tinghui Zhou, Richard Tucker, et al. “Stereo Magnification: Learning View Synthesis using Multiplane Images”. In: *arXiv: Computer Vision and Pattern Recognition* (2018) (cit. on p. 6).

[45] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. “CLIP-NeRF: Text-and-Image Driven Manipulation of Neural Radiance Fields”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 3835–3844 (cit. on p. 8).

[46] Qiling Wu, Jianchao Tan, and Kun Xu. *PaletteNeRF: Palette-based Color Editing for NeRFs*. Dec. 25, 2022. arXiv: `2212.12871[cs]` (cit. on p. 8).

[48] Yu-Jie Yuan, Yang-Tian Sun, Yu-Kun Lai, et al. “NeRF-Editing: Geometry Editing of Neural Radiance Fields”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 18353–18364 (cit. on p. 8).

Webpages

[@3] *Blender - Free and Open 3D Creation Software*. URL: `https://www.blender.org/` (visited on Apr. 25, 2024) (cit. on p. 18).

[@8] *daisyUI Tailwind CSS Components (version 4 update is here)*. URL: `https://daisyui.com/` (visited on Mar. 25, 2024) (cit. on p. 28).

[@10] *Docker: Accelerated Container Application Development*. May 10, 2022. URL: `https://www.docker.com/` (visited on Apr. 25, 2024) (cit. on p. 27).

[@11] *Express - Node.js web application framework*. URL: `https://expressjs.com/` (visited on May 1, 2024) (cit. on p. 30).

[@21] *Luma AI*. URL: `https://lumalabs.ai/` (visited on Apr. 25, 2024) (cit. on p. 11).

[@26] *Nerfacto*. URL: `http://docs.nerf.studio/nerfology/methods/nerfacto.html` (visited on Apr. 25, 2024) (cit. on p. 10).

[@27] *nerfstudio-project/viser*. original-date: 2022-11-16T00:13:51Z. May 1, 2024. URL: `https://github.com/nerfstudio-project/viser` (visited on May 1, 2024) (cit. on pp. 10, 28).

[@30] *Polycam - LiDAR & 3D Scanner for iPhone & Android*. URL: `https://poly.cam/` (visited on Apr. 23, 2024) (cit. on p. 11).

[@32]*React*. URL: https://react.dev/ (visited on Mar. 25, 2024) (cit. on p. 27).

[@33]*Record3D 3D Videos and Point Cloud (RGBD) Streaming for iOS*. URL: https://record3d.app/ (visited on Apr. 23, 2024) (cit. on p. 11).

[@36]*SoSci Survey the Professional Solution for Your Online Survey*. URL: https://www.soscisurvey.de/ (visited on Apr. 26, 2024) (cit. on p. 35).

[@37]*Tailwind CSS - Rapidly build modern websites without ever leaving your HTML*. Nov. 15, 2020. URL: https://tailwindcss.com/ (visited on Mar. 25, 2024) (cit. on p. 27).

[@40]*tensorflow/tensorboard*. original-date: 2017-05-15T20:08:07Z. Apr. 24, 2024. URL: https://github.com/tensorflow/tensorboard (visited on Apr. 25, 2024) (cit. on p. 17).

[@41]*tRPC - Move Fast and Break Nothing. End-to-end typesafe APIs made easy*. URL: https://trpc.io/ (visited on Mar. 25, 2024) (cit. on p. 29).

[@42]*Unreal Engine*. URL: https://www.unrealengine.com/en-US/home (visited on Apr. 25, 2024) (cit. on pp. 13, 18).

[@43]*Vite*. URL: https://vitejs.dev (visited on Mar. 25, 2024) (cit. on p. 27).

[@44]*Volinga*. URL: https://volinga.ai/ (visited on Apr. 29, 2024) (cit. on p. 13).

[@47]Brent Yi. *tyro*. URL: https://brentyi.github.io/tyro/ (visited on May 1, 2024) (cit. on p. 31).

List of Figures

2.1. Overview of the NeRF scene representation and rendering procedure, illustrating the stages of sampling, neural processing, and image composition [24]. 7

3.1. Instant NGP's GUI rendering a NeRF scene, showing the 3D scene, camera path editor, and training parameters. [25]. 9
3.2. Nerfstudio's web viewer connected to a remote server running the training [38]. 11
3.3. Luma AI Web Viewer in the camera path editor. 12
3.4. Volinga Suite's integrated viewer in Unreal Engine [@44]. 13

6.1. User flow diagram illustrating the key interactions for creating a NeRF model . 20
6.2. Excerpt of a view from the flow diagram with detailed interactions 21
6.3. Wireframe of the pre-processing section, showing the layout and key interaction elements 22
6.4. Dashboard overview, showing existing projects (1) and the option to create a new project (2) 23
6.5. Pre-processing section, showing the progress bar (1), input data upload (2), parameter options (3), pre-processed data list (4), and console output (5) . . . 24
6.6. Advanced settings for pre-processing, showing additional parameters and options 25
6.7. Training section, showing the training configuration form (1), and previous training runs (2) 25
6.8. Viewer section, showing the Nerfstudio viewer (1) and list of rendered outputs (2) 26

7.1. System Architecture Overview. The client communicates with the server, which in turn interacts with the nerfstudio CLI. 27

8.1. Level of experience with NeRF technology among participants. 33

9.1. Results of the User Experience Questionnaire, showing the average scores for each scale and their 95 % confidence interval. 37
9.2. Benchmarking results of the User Experience Questionnaire, comparing the results of the study to a benchmark of 452 other studies. 38

10.1. Dedicated Button for Dashboard Navigation 55

10.2. Modal Dialog for Project Creation 55
10.3. Improved File Upload Process 56

List of Tables

9.1. Summary of the User Experience Questionnaire results 38

Declaration of Honor

I do solemnly declare that I prepared this thesis independently and that the thoughts taken directly or indirectly from other sources are indicated accordingly. The work has not been submitted to any other examination authority and also not yet been published.

Munich, May 3, 2024

Eduard Aurelius von Briesen
Brandenburg Neumark

Declaration of AI Usage

I declare that I have exclusively used ChatGPT 4, OpenAI's Whisper and MAXQDA's AI features in my thesis for the purposes listed below:

- Transcription of Interviews.
- Summarizations of Research Papers.
- Correction of grammar, proofreading and editing.
- Brainstorming of thesis structure and content.

Prompts are available in the AI Usage Reflection. I confirm that I have checked my thesis and its text for unintended plagiarism and that I have verified all facts and references used from the ChatGPT output.

Munich, May 3, 2024

Eduard Aurelius von Briesen
Brandenburg Neumark