Magic NeRF Lens: Interactive Fusion of Neural Radiance Fields for Virtual Facility Inspection

Ke Li, Susanne Schmidt, Tim Rolff, Reinhard Bacher, Wim Leemans, Frank Steinicke

Abstrakt—Large industrial facilities such as particle accelerators and nuclear power plants are critical infrastructures for scientific research and industrial processes. These facilities are complex systems that not only require regular maintenance and upgrades but are often inaccessible to humans due to various safety hazards. Therefore, a virtual reality (VR) system that can quickly replicate real-world remote environments to provide users with a high level of spatial and situational awareness is crucial for facility maintenance planning. However, the exact 3D shapes of these facilities are often too complex to be accurately modeled with geometric primitives through the traditional rasterization pipeline.

In this work, we develop Magic NeRF Lens, an interactive framework to support facility inspection in immersive VR using neural radiance fields (NeRF) and volumetric rendering. We introduce a novel data fusion approach that combines the complementary strengths of volumetric rendering and geometric rasterization, allowing a NeRF model to be merged with other conventional 3D data, such as a computer-aided design model. We develop two novel 3D magic lens effects to optimize NeRF rendering by exploiting the properties of human vision and context-aware visualization. We demonstrate the high usability of our framework and methods through a technical benchmark, a visual search user study, and expert reviews. In addition, the source code of our VR NeRF framework is made publicly available for future research and development [1].

Index Terms—Virtual reality, Neural Radiance Field, Data Fusion, Volumetric Data Visualization, Human-computer Interaction, Extended Reality Toolkit

I. Introduction

In recent decades, an increasing number of new large-scale facilities have been built around the world. These facilities, including particle accelerators, nuclear power plants, and lithography machines, are critical infrastructures for scientific research and industrial processes. As they are often complex systems built after decades of planning and implemented at enormous economic cost, scientists and engineers today are actively seeking methods to effectively maintain existing facilities in order to maximize their operational lifetimes and economically upgrade them to new operational standards.
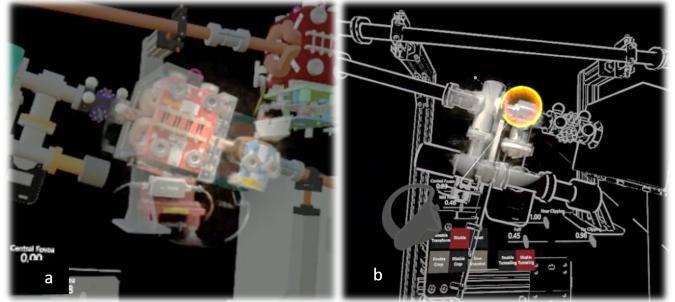
Ke Li, Reinhard Bacher, and Wim Leemans are with the accelerator division at the Deutsche Elektronen Synchrotron DESY, Notkestraße 85, 22607 Hamburg, Germany. Email: ke.li1@desy.de

Ke Li, Susanne Schmidt, Tim Rolff, and Frank Steinicke are with the Human Computer Interaction group at Hamburg University, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany.

Tim Rolff is with the Computer Vision group at Hamburg University, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany.

Please find in the supplementary material the demo video illustrating our contributions.

[1] https://github.com/uhhhci/immersive-ngp

Obrázek 1: Screenshots from the Magic NeRF Lens framework: (a) illustrates the data fusion of a high-resolution NeRF rendering and corresponding CAD model through mixed reality tunneling [2], and (b) illustrates the 3D NeRF drawing interaction using the CAD model as context.

Unfortunately, many of these facilities are hazardous environments that are inaccessible to humans most of the time. For example, at the Large Hadron Collider (LHC) at the European Organization for Nuclear Research (CERN), the particle accelerator could generate high-energy gamma radiation at doses lethal to humans during operation, making on-site operation of the accelerator tunnel limited and expensive [1]. As a result, a virtual inspection system where accurate 3D representations of the complex conditions of the facility can be viewed and updated frequently is critical for maintenance planning to reduce maintenance windows.

In fact, how to accurately replicate such complex real-world environments and how users could virtually interact with their digital replica is at the heart of computer graphics research. On the one hand, the exact 3D shapes of detailed components are too complex to be accurately modeled, updated, and rasterized using traditional geometric representations such as computer-aided design (CAD) models. On the other hand, off-the-shelf RGBD sensors [3] or photogrammetry methods [4] have limited capabilities to solve the challenging inverse 3D reconstruction problems, while professional 3D reconstruction pipelines, such as those used in heritage preservation processes, require expensive hardware and involve time-consuming multimodal data fusion and refinement [4], [5].

In recent years, neural radiance fields (NeRFs) [6] have emerged as a novel photorealistic 3D representation that can be generated by a neural network learning only from 2D images and their camera poses. In particular, the development of Instant Neural Graphics Primitives (instant-ngp) [7] is generating great excitement in the community, as it drastically accelerates NeRF training and rendering speed through an efficient multi-

resolution hashing data structure. However, unlike traditional rasterization pipelines, a NeRF generates 2D projections of 3D space through volumetric ray casting, whose performance scales largely with the rendering resolution. This hinders its application in real-time, stereoscopic, high resolution, high frame rate rendering in immersive virtual reality (VR). As a result, research on VR-NeRF interactive systems and applications is currently sparse. In addition, the question of how to interact with such neural 3D volumetric representations and how to integrate them into the conventional rendering pipeline is still under active research [8].

In this paper, we present Magic NeRF Lens, a virtual facility inspection framework that could exploit the complementary features of neural graphics primitives such as NeRF and conventional geometric primitives such as a CAD model through data fusion. In particular, our framework enables facility inspection in immersive VR, which is the emerging technology that is widely believed to become how users will ultimately interact with 3D spatial information in the future, and has already demonstrated advantages in terms of improving users' spatial presence [9] and spatial skills in such facility inspection and management tasks [10], [11].

A common requirement for virtual inspection in immersive VR is to render 3D scenes at one-to-one real-world scale to provide users with correct spatial and situational awareness of the remote environments [12]–[15]. Since rendering these NeRF models in today's wide field-of-view (FoV) VR head-mounted displays (HMDs) requires enormous computational resources, and real-time performance is only feasible by significantly degrading rendering resolution, we introduce two novel contextual volume data visualization and interaction techniques to balance this trade-off between rendering performance and visual quality. As Figure 1 (a) illustrates, the first data fusion design employs a "mixed reality (MR) tunneling" effect [2]. Here, the high-resolution NeRF rendering is displayed in the center of the user's field of view and fuses with the CAD model displayed at the periphery. As shown in figure 1 (b), the second design allows the user to define the target rendering areas by directly manipulating a binary volume bitfield via 3D drawing, using the CAD model as a contextual guide.

We validate the effectiveness of our methods and framework through technical benchmarking and a formal study in which users performed visual search tasks for facility inspection. We conducted confirmatory expert reviews with five particle accelerator control and management specialists at the German Electron Synchrotron (DESY), where some of the world's most advanced particle accelerators are built. Finally, we are releasing the source code for our VR-NeRF system, which enables features such as object manipulation, volumetric scene editing, and depth occlusion effects in the popular Unity game engine [16], [17]. As the first open source VR NeRF system available, we believe it will have a high impact on computer graphics and human-computer interaction (HCI) research.

In summary, the contribution of this work includes:

1) A VR-NeRF framework for virtual industrial facility inspection.
2) Two novel contextual data fusion designs to support interaction and rendering of industrial facility NeRF models at real-world size in immersive VR.
3) Comprehensive evaluation of the framework and data fusion designs through technical benchmarking, a user study, and expert reviews.
4) An open-source VR-NeRF system with features such as NeRF object manipulation, volumetric editing, and depth occlusion, all integrated into the Unity game engine.

II. Related Work

A. Large-scale Industrial Facility Maintenance

Large industrial facilities such as the LHC at CERN [18] are well known for their scientific missions. However, they also present some of the most demanding HCI and computer graphics challenges in facility control and visualization. One of the most significant challenges is planning and completing a large amount of maintenance and inspection work during very limited on-site access windows. For example, the 2.3-kilometer-long particle accelerator at the European X-Ray Free Electron Laser (EuXFEL) is a complex system with 10 million control system parameters and tens of thousands of components that require frequent inspection and maintenance. However, the accelerator must operate continuously for more than 5,000 hours per year, during which time on-site human access is not possible and any unexpected interruptions to operation result in high energy and setup costs [19]. To document the facility status for offline inspection and planning, Gong et al. propose a VR system by mixing CAD models with updated LiDAR 3D point cloud to visualize the real factory layout [14]. Bae et.al. propose an augmented reality (AR) system for contextual asset management by mixing 2D images with historical 3D data [20]. There is also extensive research into the use of telepresence and teleoperated mobile robots in such environments. However, much research is needed before robots can operate autonomously without human intervention for tasks that require fine motor skills [1], [12], [21]. Regardless of virtual planning or robot teleoperation in VR, a photorealistic 3D model of the real facility conditions can greatly improve the various workflows of managing these facilities.

B. NeRF

NeRF is widely regarded as a groundbreaking advance in 3D computer graphics and has received tremendous attention since its initial development [6]. By training a small neural network with only 2D images and their camera poses, a NeRF can compress scene details into a small network model instead of storing all the color values of the entire voxel grid on disk. Compared to photogrammetry or conventional RGBD sensors, creating a NeRF does not require bundle adjustment and dense reconstruction, but could produce higher visual quality than photogrammetric point clouds, which tend to be erroneous with limited image feature overlap or non-uniform textures [3]. Compared to active 3D scanners with sub-millimeter accuracy such as those with structured illumination [22], NeRF can work with "optically uncooperative"surfaces such as metallic or absorbent materials that are common in industrial facilities. While it is also common to combine both image-based and active 3D scanning to create a high-quality 3D digital twin

[4], it is noteworthy that the post-processing associated with these methods can take weeks to months, during which time the conditions of the facility may have already changed due to maintenance activities. As a result, implicit 3D models such as NeRF are a compelling alternative to traditional meshes or point clouds for rapid documentation of facility conditions.

C. VR NeRF

The rapid progress in NeRF training [6], [7], [23], rendering [7], [24], [25], and editing [26]–[28] is also causing great excitement in the VR community. With the availability of real-time NeRF rendering techniques such as instant-ngp [7], the question of how to complement NeRF rendering with practical user interfaces to support different application domains also becomes relevant for HCI research. However, the availability of open source VR NeRF systems remains sparse. In fact, most open source systems today only support scene visualization on 2D desktop [25], [28], [29] or were only available as proof-of-concept implementations using high-level programming languages such as Python, which has poor interoperability with popular graphics rendering engines. To fill this gap, we released immersive-ngp [16], the precursor to this work, presenting the first software for viewing NeRF scenes in a VR HMD integrated with the Unity game engine. Although instant-ngp subsequently introduced the ability to render a NeRF scene on a VR-HMD, the scalability of the instant-ngp VR system is limited, as integrating additional 3D models into the original framework would require users to rebuild the entire interaction and rendering pipeline from scratch [7]. This work provides further extensions to immersive-ngp so that users can easily manipulate, edit, and integrate NeRF into their own interactive 3D applications.

In addition to the lack of open source VR NeRF systems, another challenge for VR NeRF is the enormous amount of network queries required for stereoscopic, high resolution, high frame rate VR rendering [30]. One promising direction is to use foveated rendering to reduce render resolution in the peripheral region of the human visual field. However, the existing foveated NeRF method requires training and recombining the rendering results of separate networks. Moreover, it did not use the efficient multi-resolution hash coding data structure, which can achieve most of the rendering and performance speedup [25]. This work provides an alternative method to conventional foveated rendering by displaying a simpler 3D representation at the peripheral region of the human eye rendered through rapid geometric rasterization. Our method offers a more scalable solution for performance optimization and provides an example for the current debate on how NeRF could be integrated into the conventional graphics framework.

D. Magic Lens

The magic lens effect was first developed by Bier et al. [31], allowing users to change the visual appearance of a user-defined area of the user interface by overlaying a transparent lens over the render target. Interactive magic lens techniques are widely used in modern visualization systems [32], where context-aware rendering of large information spaces is needed to save computational costs [33], [34]. In 3D computer graphics, several 3D magic lens effects have been developed to allocate computational resources to more resolution-important features for the visualization of volumetric medical scans [33], [34] or context-aware AR applications [35], [36]. For immersive VR HMD rendering, the MR tunneling effect is also an interactive lensing technique that can exploit the characteristics of human vision to merge data from two sensors with different resolutions, frame rates, and latency [2]. This work extends the conventional 3D magic lens techniques to NeRF rendering by reducing the number of rays and sampling queries to the neural network.

III. THE MAGIC NERF LENS EFFECTS

A. VR NeRF Rendering Formulation

As suggested by Mildenhall et al. [6], NeRF represents the 3D world with a volumetric scene function F_Θ with learnable parameters Θ by mapping the 3D position $\boldsymbol{p}(x, y, z)$ and the viewing direction $d(\theta, \phi)$ to a color emission vector $\boldsymbol{c}(r, g, b)$ and a volume density float σ:

$$F_\Theta : (x, y, z, \theta, \phi) \mapsto (r, g, b, \sigma). \quad (1)$$

To generate a 2D projection from this 3D scene representation, NeRF rendering relies on conventional volumetric ray-casting rendering. A camera ray $\boldsymbol{r}_{(x,y)}$ from $\boldsymbol{o}(x, y)$ with viewing direction d is cast into 3D space for each pixel (x, y): $\boldsymbol{r}_{(x,y)}(t) = o(x, y) + t \cdot d$, where t is within the interval of the user-defined near boundary t_n and far boundary t_f of the ray: $t \in [t_n, t_f]$.

The color value $\boldsymbol{C}(\boldsymbol{r})$ per pixel can be calculated by sampling and accumulating the transmittance along the ray $T(t)$, the volume density function $\sigma(t)$, and the color value $\boldsymbol{c}(t)$:
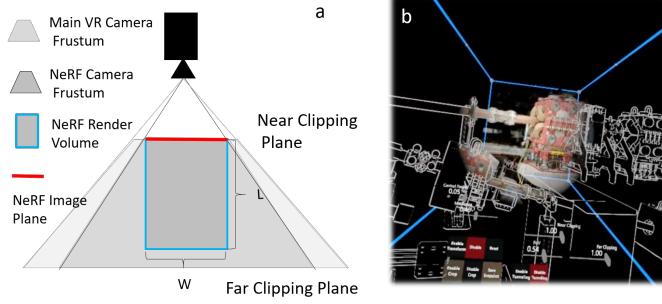
$$C(\boldsymbol{r}) = \int_{t_\mathrm{n}}^{t_\mathrm{f}} T(t) \cdot \sigma(\boldsymbol{r}(t)) \cdot c(\boldsymbol{r}(t), d) \ dt. \quad (2)$$

As a result, the NeRF rendering performance $P_{(h,w)}$ is bounded by the target render resolution with R_h height, R_w width, the number per ray $N_{r(t)}$, and the average speed of each network query (usually measured in FLOPs) $\overline{F}$ [30]:

$$P_{(h,w)} = R_h \times R_w \times N_{\boldsymbol{r}(t)} \times \overline{F} \quad (3)$$

Although the multi-resolution hash encoding data structure of instant-ngp [7] heavily parallelizes the network query process and achieves rendering speedup by significantly reducing $\overline{F}$, another strategy to further optimize rendering performance is to reduce the large number of feed-forward network queries $N_{\boldsymbol{r}(t)}$. One way to achieve this is to reduce the overall render resolution $R_{(h,w)}$, which will inevitably lead to a degradation of the rendering visual quality.

However, for a VR HMD, the per-eye NeRF rendering performance P_{hmd} is determined by the horizontal field of view FoV_h, the vertical field of view FoV_v of the HMD, and the pixel density per degree (PPD), which is the intrinsic parameter of the display hardware:

Obrázek 2: (a): Schematic sketch of the design of our interactive lens effect where the FoV of the NeRF camera is reduced and the actual NeRF render frustum is defined as a box rather than a pyramid to reduce NeRF render load. (b): Screenshot of the magic lens effect, where the blue box visualizes the NeRF crop box that is dynamically following the user's head movement.

$$P_{hmd} = FoV_h \times FoV_v \times PPD^2 \times C \times N_{\boldsymbol{r}(t)} \times \overline{F}. \quad (4)$$

Note that the constant C is a multiplication factor, given the supersampling requirements in practice for aliasing reduction and anti-aliasing [37].

From equation 4 we could see that reducing the FoV of NeRF rendering in VR could also result in a significant speedup without sacrificing visual quality. Another way to optimize $P_{(h,w)}$ is to reduce the overall sample size $N_{\boldsymbol{r}(t)}$. Abundant previous work has shown that this can be achieved by empty space skipping and early ray termination techniques [38], which were already part of the original instant-ngp implementation [7]. In the remainder of this section, we demonstrate the design of two 3D magic lens effects to efficiently utilize the performance optimizers, including FoV restrictor, empty space skipping, and early ray termination, for rendering real-world scale industrial facilities in a VR HMD.

B. Magic NeRF Lens with FoV Restrictor

By applying a FoV restrictor to the NeRF model, it creates a "VR tunneling" effect [39], which is a typical motion sickness reduction technique in VR gaming to restrict the optical flow and sensory conflicts of the peripheral region of the human eye [40]. However, a FoV restrictor could significantly reduce the user's sense of presence and immersion. To compensate for the NeRF rendering reduction, we implement the "MR tunneling"effect [2] by displaying a CAD model or other types of conventional 3D representations of the industrial facilities in the peripheral regions of the user's field of view. Previous research shows that MR tunneling could save computational costs by merging two visual modalities with different resolution, frame rate, and motion-to-photon latency while achieving more desirable overall perceptual effects [2].

Figure 2(a) shows the schematic relationship between the main VR camera, the NeRF camera, and the active NeRF rendering volume. Figure 2 (b) shows the volumetric crop box of the NeRF model, which is dynamically aligned with the user's viewing direction and the near clipping plane of the VR camera, so that the volume crop box acts as an interactive lens that automatically selects the target rendering volume according to the size of the image plane W and the distance of the far clipping plane L. The parameters W and L are both user-defined values to reduce the total number of sampling points and could be adjusted during application runtime to avoid large frame rate jitter. For the static MR tunneling effect, the high-resolution NeRF rendering is displayed in the central region of the HMD. However, an eye tracker could be integrated to achieve foveated MR tunneling and reduce the need for frequent head movements while inspecting the facilities [2].

C. Magic NeRF Lens with Context-aware 3D Drawing

For some facility inspection tasks, users typically do not need the entire NeRF model for their tasks. For example, components such as walls, simple electrical boxes, floors, and ceilings are static elements that normally do not need to be inspected and modified, and could be visualized simply by their CAD models. Based on the original idea of magic lens in user-defined context-aware rendering, we design the magic NeRF lens effect with 3D drawing interaction, where users can dynamically reveal and erase an area of the NeRF model. As shown in Figure 1 (b), the wireframe representation of a CAD model outlines the edges and general context of the environment, and users could adjust the radius of a 3D sphere attached to the VR controller and point the 3D sphere at a spatial location where the NeRF render volume should be revealed on demand.

The transparency representation of NeRF could be considered as a 3D density grid $\sigma(h,w,l)$, where h, w, l are the data height, width, and length. With the empty space skipping technique, the network query at the location where the volume density value is zero, indicating that this space is essentially empty, could be automatically ignored for that sample location. Therefore, our proposed 3D NeRF drawing effect could further improve the overall rendering speed in VR by computing only the areas specifically requested by the user.

IV. SYSTEM IMPLEMENTATION

To implement the proposed magic NeRF lens effects, we first develop a VR NeRF system. Since our technique requires combining NeRF rendering with other types of 3D representation in the conventional rasterization pipeline, we developed immersive-ngp [16], [17], the implementation of the VR-NeRF framework in the Unity game engine.

A. The VR-NeRF Plugin for Unity

The early version of immersive-ngp that we have released includes a basic Unity native rendering plugin that supports data sharing between the instant-ngp application [7] and the Unity [16], [17] render engine, allowing pre-trained NeRF models from instant-ngp to be loaded and rendered directly into the Unity game engine. For rendering stereoscopic NeRF models within a VR HMD, two render textures are natively
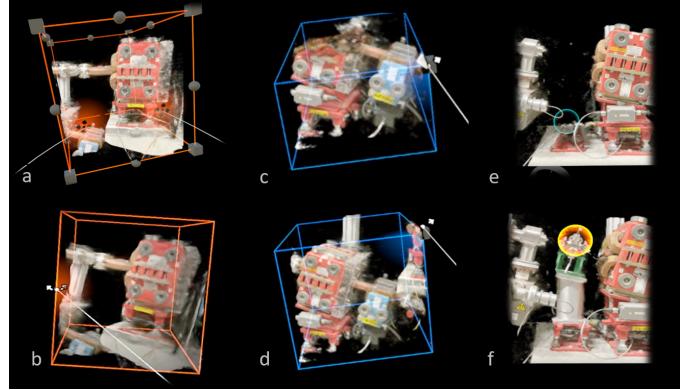
created and placed as screen space overlays in front of the user's eyes. Synchronization problems are avoided by updating both render textures simultaneously in a single frame. Unity's tracked VR head poses are converted directly to instant-ngp camera matrices in the instant-ngp coordinate system. Users can navigate the 3D NeRF space using conventional VR joystick controls. In terms of software, the immersive-ngp framework uses Unity Editor version 2019.4.29f1 with the OpenVR desktop runtime, which supports a wide range of VR runtimes such as Oculus and SteamVR. Both immersive-ngp and instant-ngp use OpenGL as their graphics application programming interface (API) and support Deep Learning Super Sampling (DLSS). In this work, we use the same software architecture and extend the preliminary implementation so that a NeRF model can be transformed, manipulated, edited, and integrated as additional 3D objects into the Unity rendering pipeline.

B. Immersive-NGP Extensions

a) Manipulating a NeRF Model as an Object: To support spatial manipulation of a NeRF model in VR, we create a model space for the NeRF model whose spatial properties such as position, rotation, and scale are defined by a volume bounding box. As Figure 3 (a-b) shows, the bounding box is represented as a transparent cube whose translation (T), rotation (R), and scale (S) matrices are combined into one transformation matrix that is applied to the view matrix of the instant-ngp camera to render the correct view of the NeRF model using the classical view model projection theory. The object manipulation interaction from the Mixed Reality Toolkit (MRTK) is attached to the cube, allowing the user to intuitively rotate, scale, and translate the box and its associated NeRF model using one- or two-handed control.

b) Crop Box Editing: To allow arbitrary spatial transformation of the axis-aligned bounding box (AABB) in VR, we attach a second 3D cube game object with an object manipulator where the spatial transformations of the Unity AABB bounding box, such as translation and rotation, are applied to the AABB crop box defined in instant-ngp's coordinate system. As shown in Figure 3 (c-d), users can extend, reduce, or rotate the AABB render volume using the same object manipulation interaction provided by MRTK.

c) Volumetric Editing: We adopt instant-ngp's implementation of a volumetric editing feature in our framework. Similar to instant-ngp's preliminary implementation, a 3D sphere object is attached to the controller at a certain distance to indicate the intended drawing region. As Figure 3 (e) shows, users can interactively select the region where voxels within the sphere should be made transparent by moving the sphere to the target region in 3D space while pressing a button on the VR controller to confirm the drawing action. The same interaction and action could be performed to reveal the volume at the region set to transparent, as shown in Figure 3 (f). In an instant-ngp NeRF model, the volume density grid σ stores the transparency of the scene learned by the network. A binary bitfield bit_σ can be generated based on a transparency threshold and can be used to quickly mask the transparent region from being sampled during raytracing [7].
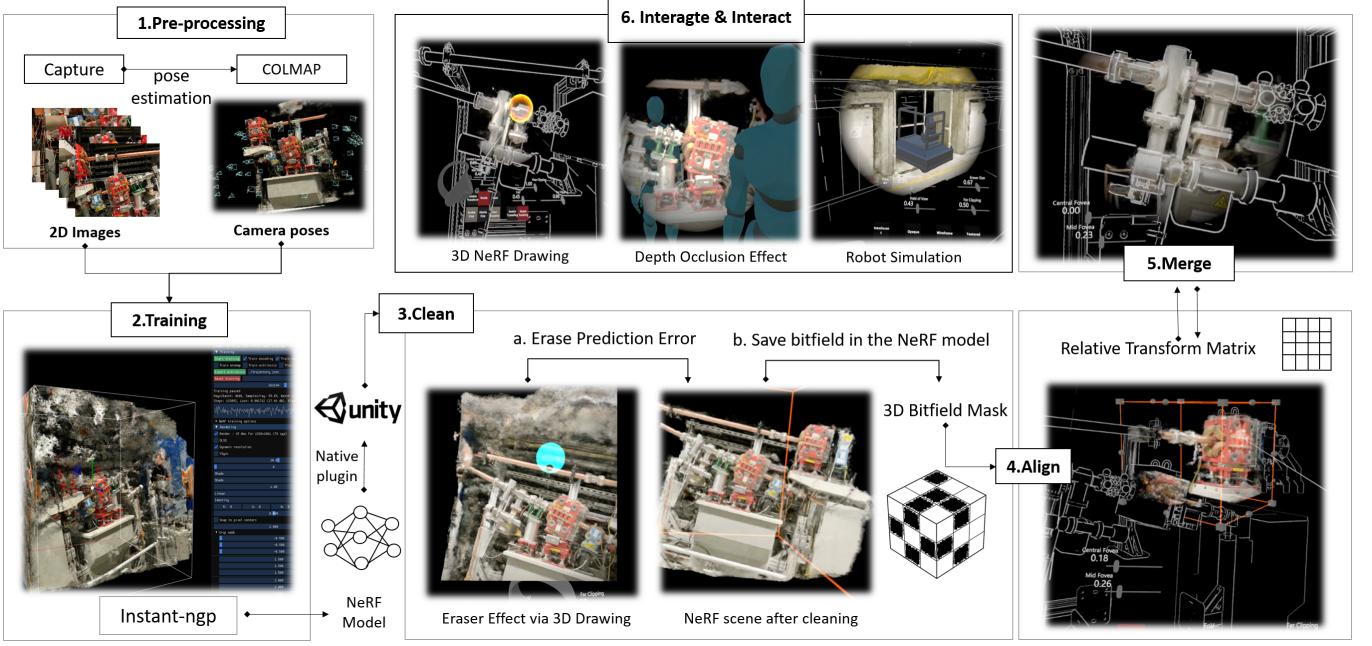

Obrázek 3: Illustration of our system extension to immersive-ngp. (a-b): NeRF model manipulation, where users can rotate, scale, and translate the model in VR. (c-d): NeRF model crop box manipulation, where users can rotate and scale the render volume crop box in VR. (e-f): Volume editing, where users could dynamically delete and remove an area of the render volume via 3D VR drawing.

d) Depth-based Rendering: Depth-based rendering is important for creating applications that need to incorporate various visual effects such as occlusion, physics simulation, and scene relighting [41] for image-based rendering. We integrate depth-based rendering into the Unity game engine by implementing an additional CUDA surface object in instant-ngp, where the depth map from the NeRF camera can be copied directly to a Unity texture on the GPU. When enabled, the per-eye depth map of the NeRF camera could be rendered at every frame, and users could select different depth scales according to the needs of their application. As shown in Figure 4, a depth occlusion effect could be achieved to integrate virtual objects into the NeRF model by performing a pixel-by-pixel depth test between the Unity camera depth map and the NeRF camera depth map.

In summary, the immersive-ngp extensions introduced in this work provide many core components for building interactive VR NeRF applications. The preliminary implementations of the work have already been presented at two VR demo exhibitions [16], [17], where users could test the end-to-end pipeline of scanning, manipulation, and editing. A full demo video of the current system can be found in the supplementary material.

C. Data Fusion Pipeline for Multi-modal Rendering

In this section, we focus on developing a data fusion pipeline to merge a NeRF model with its corresponding CAD model, although the proposed pipeline is also applicable to other types of conventional 3D data. We choose CAD models because CAD models of large-scale physics facilities are often created in the early stages of facility design and have complementary features to their corresponding NeRF models. In contrast to industrial product modeling, CAD models of large-scale industrial facilities typically lack realistic textures because the detailed environments are usually too complex

Obrázek 4: Overview of the data fusion pipeline to merge a NeRF model with its corresponding CAD model, illustrating sub-processes from image pre-processing, NeRF model training, scene cleaning, scene alignment, scene merging, and examples of final integration and interaction using different features of our framework.
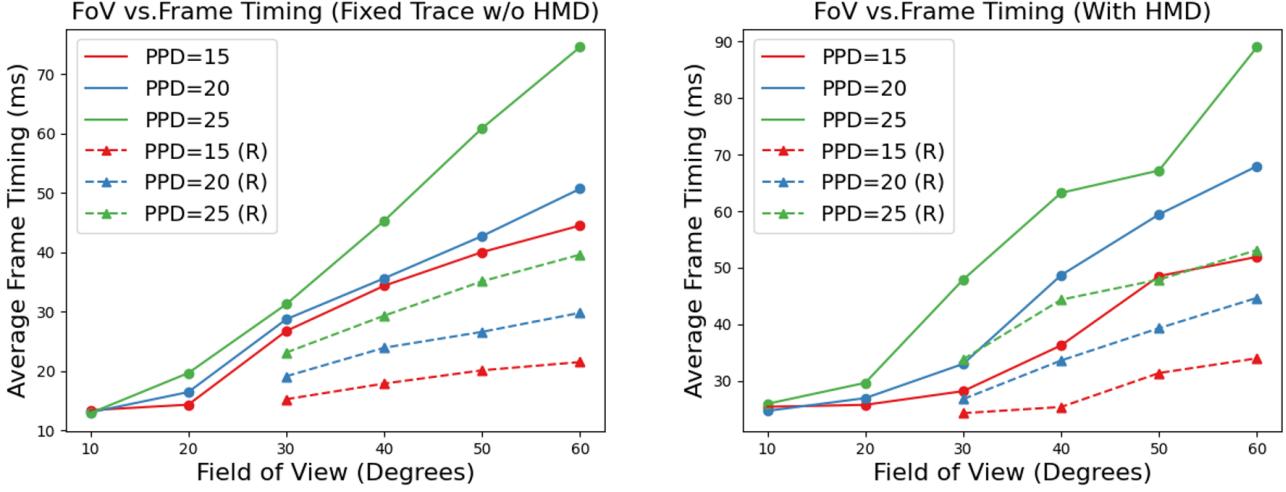
to be modeled accurately and the environment is bound to be updated frequently [14]. In addition, semantic information typically embedded in CAD models is not represented in NeRF models without additional expensive network training and voxel-wise segmentation. Other benefits of fusing a CAD model with a NeRF include a more efficient depth occlusion effect using the CAD model without the need to perform pixel-wise depth testing, while potentially laying the foundation for more advanced NeRF editing interactions such as those proposed in NeRFShop [28].

Figure 4 provides an overview of the data fusion pipeline, which is divided into the following six steps.

1) **Preprocessing.** Since most 2D images do not contain their camera poses, the 2D images must be preprocessed using the conventional Structure from Motion (SfM) algorithm [42] to estimate the camera poses using software such as COLMAP [43]. However, this step can be skipped for cameras that can track their own poses.

2) **Training.** The processed data is trained in the instant-ngp framework to generate an initial estimate of the scene function F_Θ and an initial occupancy grid σ_0 whose size is defined by a preset AABB bounding box.

3) **Scene cleaning.** The quality of NeRF rendering can be affected by artifacts of real-world 2D images, such as motion blur, lens distortion, or insufficient images around the viewing angle. This often results in false clouds in the 3D NeRF scene, which can degrade the user's viewing experience and needs to be cleaned up before the NeRF model can be applied to practical applications. Therefore, our VR NeRF framework provides the ability to modify the pre-trained instant-ngp NeRF model F_θ. The initial density grid estimation σ_0

is inspected and cleaned using the interactive eraser function described in section IV-B. Users can manually remove regions with cloudy prediction errors or low render quality. The edited density grid σ_1 and the binary bitmask of the density grid σ_{bit} are stored in the NeRF model for reuse. When reloading the cleaned NeRF model, our framework automatically checks the saved bitmask, so that the erroneous network prediction that has already been removed will no longer be sampled and rendered.

4) **Scene Alignment.** The user could manually align the cleaned NeRF model with the CAD model using the NeRF object manipulation and crop box editing functions described in section IV-B. If the NeRF model is adjusted to the one-to-one scale of the real facility, alignment using the entire NeRF model will result in a rendering volume too large to be efficiently manipulated in VR. Therefore, it is recommended to perform the alignment process by focusing on a small part of the scene for accurate object manipulation.

5) **Scene Merging**. In the merge step, the user could perceptually validate the alignment by reducing the FoV of the NeRF camera and adjusting the transparency of the shader that renders the NeRF images so that both the CAD drawing and the NeRF rendering are simultaneously visible. The user can iteratively move back and forth between the alignment and fusion steps until the two 3D representations are spatially aligned. The relative transformation matrix between the NeRF model and the CAD model can be saved for reuse.

Obrázek 5: Systematic benchmark results showing (left): the relationship between rendering FoV and average frame time following a predefined fixed path without a VR HMD attached, and (right): the trend between rendering FoV and average frame timing for a test user following only approximately the same 3D trace. (R) indicates reduced rendering via 3D NeRF drawing.

V. PERFORMANCE BENCHMARKING

A. Experiment Design

We examine the performance trend when varying the NeRF rendering FoV from $10°$, which covers the foveal region of human vision, to $60°$, which covers the average central visual field for most people [44]. We also vary the PPD value to match the resolution requirements of VR displays of different quality, including medium display resolution (PPD=15, e.g. Oculus Quest 2), medium to high-end displays (PPD=20, e.g. Meta Quest Pro), or high-end displays (PPD=25, e.g. Varjo XR-3). The final NeRF render resolution per eye ($R \times R$) for each configuration is calculated using the following equation:

$$R = FoV \times PPD \times 2 \tag{5}$$

where we multiply the pixel density per degree by 2 to account for the upsampling required for aliasing reduction and edge smoothing via supersampling [37].
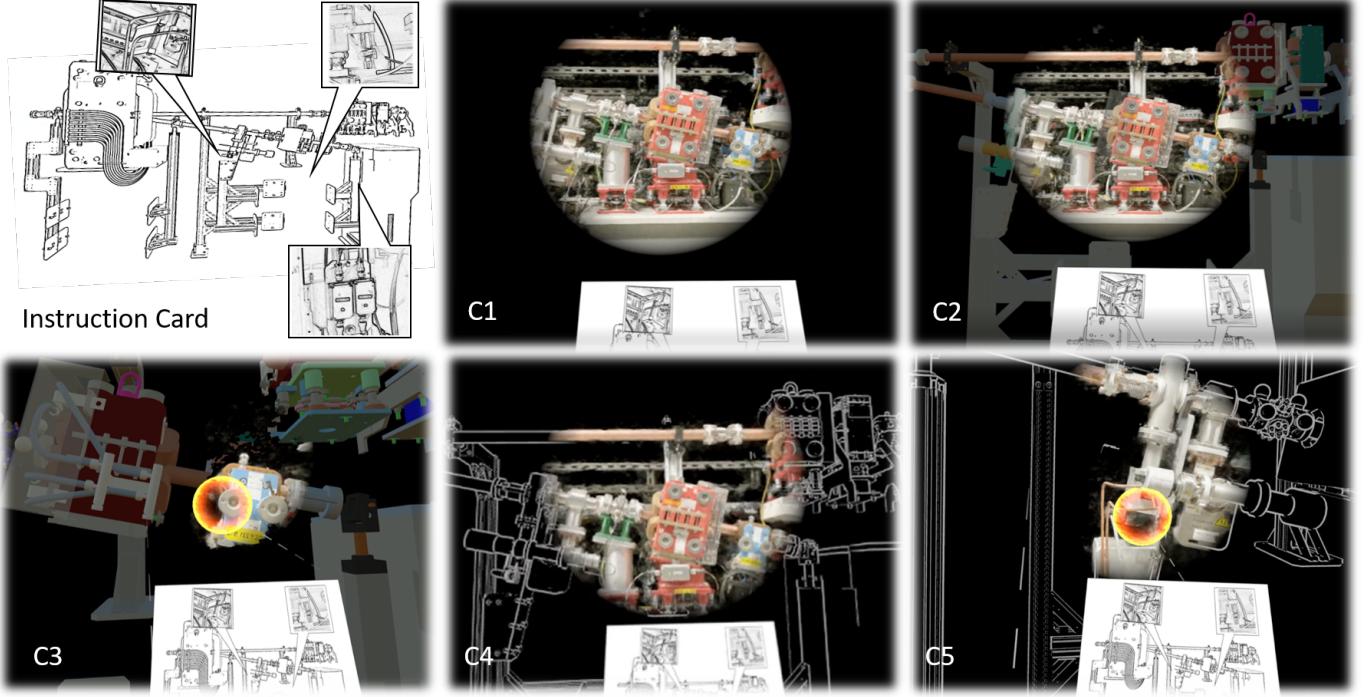
B. Materials

We evaluate the performance of our system on a self-generated real-world dataset consisting of 60 images reconstructing a section of the EuXFEL particle accelerator. Its VR NeRF reconstruction is shown in Figure 6. Since the CAD model is designed to be at an exact scale of the real facility, we aligned the NeRF model with the CAD model using the data fusion pipeline described in section IV-C and were able to measure that the NeRF rendered volume of this NeRF model corresponds to approximately 2.2 m $\times$ 1.47 m $\times$ 2.08 m in real scale, which would require a full FoV VR rendering to examine closely. All of our benchmarks are run on an Nvidia Geforce RTX 3090 GPU and an Intel(R) Core(TM) i7-11700K CPU with 32GB of RAM. As demonstrated in previous research [16], DLSS is critical to achieving real-time immersive VR NeRF rendering performance, so DLSS is also enabled for all benchmarking efforts.

C. Systematic Benchmark (w/o HMD)

To simulate the system performance of how a user would use our framework to inspect different parts of the NeRF model, we first asked a test user to closely inspect three fixed locations within the NeRF model. A custom script recorded the user's exact 3D trajectory with the camera transformation information at each frame. The relative position of the three target components with respect to the CAD model was visualized as a schematic sketch, as shown in Figure 6-(Instruction Card). A video recording of the 3D trace is also available in the supplementary material.

First, we performed a systematic benchmark where we collected the average frame timing ($\overline{f_t}$) for stereoscopic rendering of different display FoV and PPD configurations by replaying the same pre-recorded 3D trace. Figure 5 (left) plots the average frame timing result for each rendering configuration. As expected, the rendering latency scales linearly with increasing FoV for the basic NeRF lens effect. Additionally, it is shown that our system could theoretically achieve real-time VR rendering of 30 frames per second ($\overline{f_t} < 33.33ms$) at less than $50°$ FoV.

To evaluate whether additional contextual rendering could lead to real performance gains, we create an edited NeRF model by 3D drawing interaction, where we define the 3D density grid so that only the target components are visible. Since FoV settings below $20°$ could already achieve good real-time performance without reducing the rendering volume, we only examine FoV and IPD settings above $30°$ in this section. Figure 5 (left) also plots the average frame timing for each rendering configuration for rendering the selected regions. As expected, exposing only the target render volume reduces the overall render load due to instant-ngp's empty space skipping and early ray termination implementation. This confirms the performance optimization insight that removing unimportant scene details using the 3D NeRF drawing effect could be an

Obrázek 6: Screenshots for different visualization and interaction conditions of Magic NeRF Lens. (C1): Baseline condition with only the NeRF model and the FoV restrictor. (C2): Magic NeRF Lens with a textured CAD model as context. (C3): 3D NeRF drawing with a textured CAD model as context. (C4): Magic NeRF Lens using only the wireframe representation of the CAD model. (C5): 3D NeRF drawing with wireframe representation of the CAD model.

option to gain additional performance.

D. Empirical User Benchmark Results (With HMD)

In the second benchmarking experiment, we evaluate the same set of rendering configurations. However, instead of replaying a pre-recorded 3D trace, a test user was asked to repeat each configuration, following roughly the same path to inspect the three target components to evaluate the overall performance of the actual VR system. For high-resolution configurations where real-time performance could not be achieved, the test user was able to use keyboard control instead, while keeping the VR HMD connected to the entire rendering pipeline. Figure 5 (right) shows the general trend between FoV and $\overline{f_t}$. Compared to the systematic performance results, the actual total frame timing is about $5 - 10$ ms higher because of the additional computational resources needed to drive the HMD. However, the general performance trend in an actual system with VR HMD is similar to the systematic benchmark result. Additionally, we could derive that for an actual VR NeRF system, the optimal render resolution is 1200×1200 at $30°$ FoV (PPD=20) or at $40°$ FoV (PPD=15).

VI. USER STUDY

Understanding human factors is an important step in the development of VR systems and interaction techniques. Moreover, the impact of using NeRF in an interactive VR framework on actual users is currently unknown and unexplored in the community. In this section, we present a perceptual experiment where we invited 15 users to perform a facility inspection task through visual search activities using different magic NeRF lens effects. We collect various perceptual metrics to report on users' perceived motion sickness, sense of presence, task load, and perceived usability, from which we derive design insights for future applications.

A. Study Design

a) Conditions: In this user study, we are interested in the effects of different magic NeRF lenses on the performance of virtual facility inspection tasks. In particular, we focus our study design on the perceptual effects of fusing NeRF with its corresponding CAD models. We investigate the two magic lens designs with the two most common types of CAD model visualizations in large-scale industrial facilities. As shown in Figure 6, (C1) is the baseline condition with only FoV restriction but no data fusion for comparison, (C2, C4) implement the magic NeRF lens effect with FoV restriction, and (C3, C5) implement the magic NeRF lens effect with 3D drawing. (C2, C3) use a CAD model with colored abstract texture, while (C4, C5) use a CAD model with only wireframe representation, visualizing only the contextual outline.

b) Materials: The user study was conducted on the same graphics workstation and software configuration used for the system benchmark described in section V. For all conditions, the FoV of the NeRF rendering camera was set to $30°$ with a per-eye render resolution of 1200×1200 pixel, which is one of the optimal rendering configurations determined in our benchmark experiments. For the 3D NeRF drawing conditions

(C3) and (C5), the reveal sphere is placed at $0.7m$ in front of the right VR HMD controller. We used an Oculus Quest Pro VR HMD, which has a PDD value of 22. However, our VR NeRF framework is compatible with other VR headsets that support the SteamVR and OpenVR desktop runtime.

c) Tasks: To simulate how a user would perform virtual inspection tasks, we designed a visual search activity where the user was asked to locate three detailed components in the NeRF model based on a schematic sketch. As shown in Figure 6-(Instruction Card), the schematic sketch consisted of an overview of the CAD model, with each search target highlighted in a dialog box. Arrows are provided to indicate an approximate location where the detailed component might be found. The instructions for each search target were generated from the real-world images and therefore represent updated real-world conditions compared to the abstract CAD model. All search targets are detailed elements that could not be effectively updated in the CAD model due to maintenance activities or the difference between actual installation and original design. A pilot test with 3 users confirmed that all target components in each scene could be found in 3-5 minutes for all proposed conditions. To avoid learning effects in the course of the study, 5 different scenes were prepared, each showing a different part of the actual particle accelerator tunnel at the EuXFEL.

d) Participants: We invited 15 participants, 4 female, 10 male, and 1 who preferred not to disclose their gender. 5 participants were between 18 and 24 years old, and 10 participants were between 25 and 34 years old. All were students or researchers in HCI, Computer Science, or Physics, with HCI students receiving compensation in the form of course credits. 8 participants use VR systems regularly (at least once a month), and only 2 use them less than once a year, 5 never used VR before.

e) Procedures: Each participant first signed a consent form and completed a demographic questionnaire. The experimenter then presented a test VE in which participants could practice locomotion and 3D VR drawing controls for a maximum of 3 minutes. Each participant was then presented with the 5 conditions in order. To counterbalance the order and assignment of scenes to conditions, we used a replicated Latin square design with one treatment factor (condition) and three blocking factors (participant, trial number, and scene). As a result, each condition/scene combination was experienced by 3 participants. During each condition, participants could press a "start"button to begin the study and begin time recording to assess task performance. The experimenter monitored the participants' VR interactions via a secondary screen that mirrored the VR displays. After the participant informed the experimenter when they had found a component, the experimenter confirmed its correctness, after which the participant could begin searching for the next search target. Once all three components were found, the participant could press an "end"button, which marked the end of time recording for the search tasks. The total study time was approximately 60 minutes.

B. Results

We collected multiple objective and subjective measures to assess the user experience as study participants interacted with our magic NeRF lens framework. Tables listing the overall ratings for each subscale of the rating scales are included in the supplementary material.
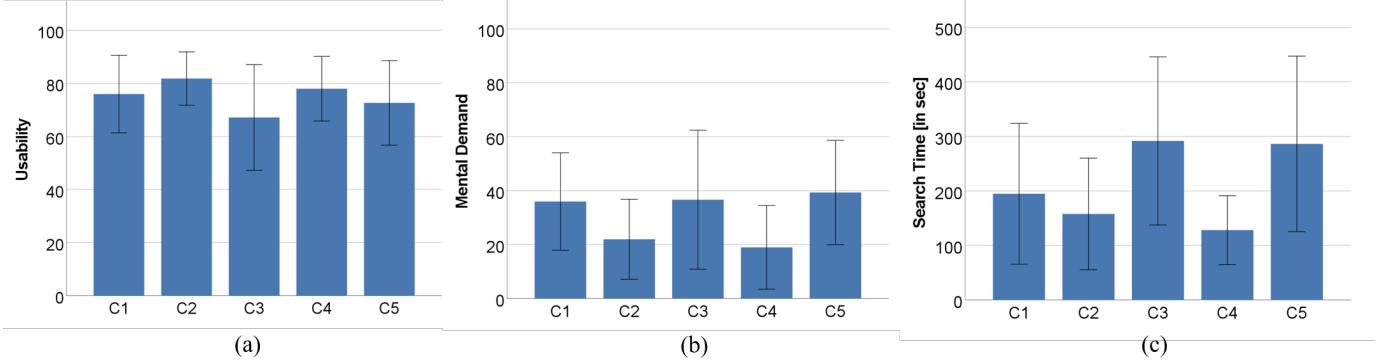
a) Task Performance: Task performance was measured as the time it took participants to find the three locations indicated on the instruction card for each scene. A Shapiro-Wilk test indicated that the residuals of search time were not normally distributed, which was confirmed by visual inspection of the QQ plots. Therefore, a Friedman test was performed, which revealed a significant effect of visualization method on search time ($\chi^2(4) = 23.307, p < .001$). Post-hoc Wilcoxon signed-rank tests with Bonferroni-Holm adjustment of p-values showed significant differences between C2 and C3 ($Z = -3.010, p = .018$), C2 and C5 ($Z = -3.124, p = .014$), C4 and C3 ($Z = -3.237, p = .011$), and C4 and C5 ($Z = -3.351, p = .008$).

b) Perceived Workload: We measured six aspects of perceived workload using the NASA-TLX questionnaire [45]. Because each aspect was represented by a single Likert scale ranging from 0 to 100 (with 21 levels), we used nonparametric Friedman tests to analyze the responses. There was a statistically significant difference in mental demand depending on the condition, $\chi^2(4) = 17.857, p = .001$. Post-hoc analysis with Wilcoxon signed-rank tests and Bonferroni-Holm p-value adjustment revealed a significant difference between C2 and C5 ($Z = -3.257, p = .011$). Friedman tests also showed significant effects of visualization method on effort ($\chi^2(4) = 15.366, p = .004$) and frustration ($\chi^2(4) = 10.183, p = .037$), although none of the post-hoc tests were significant after p-value adjustment. No significant differences were found for physical demand, temporal demand, and performance.

c) Usability: Usability was measured using the System Usability Scale (SUS) [46] and converted to a value range [0,100]. A repeated measures ANOVA with Greenhouse-Geisser correction revealed a significant effect of visualization method on usability, $F(2.514, 35.202) = 4.269, p = .015, \eta_p^2 = .234$. Post-hoc tests with Bonferroni correction showed a significant difference between C3 and C4 ($p = 0.48$).

d) Presence: We measured the sense of presence using the Igroup Presence Questionnaire (IPQ) [47] with all three subscales *Spatial Presence*, *Involvement*, and *Experienced Realism*, as well as a single item assessing the overall "sense of being there". Three ANOVAs for the subscales and a Friedman test for the single item revealed no significant differences.

e) Cybersickness: As a subjective measure of cybersickness, participants rated the perceived severity of 16 symptoms covered by the Simulator Sickness Questionnaire (SSQ) [48]. From the ratings, we calculated subscores for nausea, oculomotor, and disorientation, as well as a total cybersickness score, as suggested by Kennedy et al. [48]. No increase in cybersickness across trials was observed, so we analyzed absolute values rather than relative differences between trials. Since the residuals were not normally distributed, we performed a Friedman test for each of the

Obrázek 7: Mean (a) usability, (b) mental demand as measured by the NASA-TLX, and (c) search time to finish the task in seconds. Vertical bars indicates the standard deviation.

four (sub)scores. For nausea, a significant difference between conditions was found, $\chi^2(4) = 10.330, p = .035$, but this could not be confirmed by post-hoc pairwise Wilcoxon signed-rank tests with Bonferroni-Holm correction. For oculomotor, disorientation, and total cybersickness scores, no significant effects were found using Friedman tests.

C. Discussion

a) Spatial Orientation: The magic lens effects with FoV restriction (C2, C4) achieved better ratings in terms of mental effort and higher task performance than C1, though not being significant. Qualitative user feedback indicated that C2 and C4 "*make(s) it more confident /easier to navigate around the machines*" (N=2). Without the contextual guidance of CAD models, "*it was difficult to get a feeling of scale or to iden-tify different places correctly.*"Detailed comments mentioning spatial orientation confirmed that these effects further "*helped with general orientation*" and "*provided a sense of [the user's] position*". In contrast, the 3D NeRF drawing interaction (C3, C5) performed significantly worse than the 3D NeRF drawing effect (C2, C4). The main problem reported with these con-ditions was "*difficulty to find the correct depth*" (N=4). This could be because most participants were unfamiliar with the complex environments, and using only a 2D schematic sketch as visual instruction does not provide users with sufficient spatial orientation to navigate efficiently to the search target. As a result, future design of the magic NeRF lens effect with 3D drawing could provide 3D spatial anchors as visual cues for non-expert users.

b) Task Load, Performance, Usability, and Cyber-sickness: Across all conditions, there were no significant differences in perceived cybersickness, with participants expe-riencing little to no motion sickness. In general, users found the magic lens effects with FoV restrictor (C2, C4) highly usable ($SUS_{C2} = 81.83$, and $SUS_{C4} = 78$), with better objective task performance and lower perceived mental effort. However, the magic NeRF lens effects with 3D drawing received lower usability scores, with a significant difference between (C3) and (C4). As Figure 7 shows, usability scores were strongly correlated with cognitive load and task perfor-mance. Qualitative feedback indicates that participants often repeatedly unfolded and erased the NeRF models because they were unsure where to find the components, leading to frustration and even framerate jitters when more areas than necessary were revealed. In addition, participants mentioned that the CAD is often outdated and incomplete compared to the NeRF model. This discrepancy makes it even more difficult for users without a facility management background to complete the tasks. Nevertheless, two participants mentioned the 3D NeRF drawing effect as their preferred condition because "*it was more fun* " and has "*a nice property of reducing complexity and putting the focus on the spots you want to investigate*". Therefore, we recommend future use of 3D NeRF drawing effect mainly for facility management experts, but still make it an option for non-expert users to accommodate different preferences.

c) Context Rendering Style: With respect to the rende-ring style of the CAD model (colored solid vs. wireframe), the conditions yielded similar results in both subjective ratings and task performance. Since users reported different preferences in the open-ended feedback, a customization option could be offered in a practical application.

VII. EXPERT FEEDBACK

To explore how our framework could be used by practitioners in a real industrial setting, we validated our system through expert reviews at DESY, where the development of a VR NeRF system for particle accelerator maintenance was first proposed. Five facility managers and control system specialists participated in the reviews. All participants have a leading position in the design, coordination or control of particle accelerators at DESY, and two of them are also experienced VR experts who have already developed VR systems for facility inspection.

The expert reviews were conducted using an exploratory application that illustrates a section of the NeRF model of the EuXFEL particle accelerator. It provides many flexibility and customization options, allowing the user to freely adjust the NeRF camera's field of view, change the size of the NeRF editing sphere, vary the translucency of the merging effects, as well as the manipulation interactions mentioned in section IV. The application ran on an Alienware m17 R2 laptop with

16GB of RAM and an RTX 2080 GPU, where we reduced the resolution of the application to 800×800 pixels to achieve real-time performance.

Overall, participants felt "*very confident to use the system*" and "*it is something (they) could work with*", even though the application was running on a laptop with moderate performance and moderate resolution. All expert participants confirmed that using NeRF for virtual facility inspection could benefit their workflow. One facility management expert commented that NeRF is a compelling, low-cost alternative for 3D facility documentation: "*I think the system has a good advantage. It is quite nice to project the NeRF model on the CAD model, as it is a lot more effort to take laser scans of the facility*". In addition, most expert participants preferred to inspect the facilities in VR because the system "*helps them to see if (they) could reach anything*" or "*if (they) could fit any equipment through the existing environment*". In addition, two experts mentioned that having a 1:1 scaled NeRF model aligned with the CAD model in immersive VR also gives them a better spatial awareness of the complex machines than working with a 2D desktop application. For example, they mentioned that the VR NeRF environment could help them assess in advance if "*an operator's hand would fit through a narrow gap to handle components*" or if special equipment would need to be prepared in advance.

The expert who leads the design and upgrade of particle accelerators mentioned the benefits of data fusion visualization: "*With this system, I see the possibility to test something in theory before you build it in practice. For example, when you have a machine, and you want to test if you have enough space for installing it, it is quite nice you could test everything in the virtual area before you do it in reality*". He also mentioned that the CAD model of the particle accelerator usually only shows the initial design of the facility. Once the facility is operational, these CAD models can become incomplete and outdated. He found the contextual 3D NeRF drawing effect particularly helpful in comparing the difference between the original design and the actual implementation, which could even help operators update the original CAD designs accordingly.

For future development of the facility inspection system, one expert suggested the interesting idea of integrating contextual QR code scanning to further support information retrieval from their large inventory database. Sometimes, facility inspection tasks at the particle accelerator require scanning labels containing QR codes with manufacturing and maintenance information that are attached to all cables and equipments. The ability to retrieve such labels directly from the NeRF model would further streamline facility inspection processes. With regard to safety-critical processes such as immersive robot teleoperation, participants mentioned that although NeRF could provide photometrically accurate results, it's geometric accuracy also needs to be verified and compared with conventional 3D sensors. Nevertheless, as our proposed NeRF magic lens effects and data fusion pipeline could be applied to other types of conventional 3D models, we encourage further investigation of experimenting with data fusion with other types of 3D data of large-scale facilities.

VIII. Limitations

Our framework still has several limitations that could be addressed in future work. First, future research could investigate an automatic CAD-NeRF alignment approach to skip the manual hand-eye calibration process. Detailed investigation of an interactive point-matching algorithm could be a promising approach, taking into account the real-world mismatch between the NeRF model and the CAD model [49]. In addition, optimization techniques such as empty space skipping and early ray termination could also lead to frame rate jitter as the number of network queries becomes view-dependent. Although the use of a FoV restrictor could reduce such effects, such framerate jitter will be more noticeable on medium and low end graphics hardware. Therefore, we encourage future work to investigate the integration of further optimization techniques, such as foveated rendering combined with user visual saliency, to enable a comfortable VR NeRF system even on low-end graphics devices.

IX. Conclusion

We presented Magic NeRF Lens, an interactive immersive VR framework that can support virtual facility inspection using both photorealistic NeRF rendering and geometric rasterization. From the system performance benchmark, we show that the optimal VR NeRF rendering configuration for visualizing large-scale facilities at one to one real-world size is 20 PPD at $30°$ FoV, or 15 PPD at $40°$ FoV. To support rendering of large NeRF models, we used multimodal data fusion designs for contextual and on-demand NeRF rendering. Through a visual search user study, we demonstrate that our MR tunneling magic NeRF lens design achieves high usability and performance, while the 3D NeRF drawing effect is more interactive but more suitable for expert users. Follow-up system reviews with 5 experts confirmed the results of the user study and suggested various directions for future work. We believe that the interdisciplinary nature of this work could benefit both industrial practitioners and the VR community at large.

Acknowledgments

Reference

[1] M. di Castro, M. Ferre, and A. Masi, "Cerntauro: A modular architecture for robotic inspection and telemanipulation in harsh and semi-structured environments," *IEEE Access*, vol. 6, pp. 37 506–37 522, 2018.

[2] K. Li, S. Schmidt, R. Bacher, W. P. Leemans, and F. Steinicke, "Mixed reality tunneling effects for stereoscopic untethered video-see-through head-mounted displays," *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 44–53, 2022.

[3] M. Zollhöfer, P. Stotko, A. Görlitz, C. Theobalt, M. Nießner, R. Klein, and A. Kolb, "State of the art on 3d reconstruction with rgb-d cameras," *Computer Graphics Forum*, vol. 37, 2018.

[4] F. Remondino, "Heritage recording and 3d modeling with photogrammetry and 3d scanning," *Remote. Sens.*, vol. 3, pp. 1104–1138, 2011.

[5] L. Gomes, O. Regina Pereira Bellon, and L. Silva, "3d reconstruction methods for digital preservation of cultural heritage: A survey," *Pattern Recognition Letters*, vol. 50, pp. 3–14, 2014, depth Image Analysis. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167865514001032

[6] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision*, 2020.

[7] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (TOG)*, vol. 41, pp. 1 – 15, 2022.

[8] A. Tewari, O. Fried, J. Thies, V. Sitzmann, S. Lombardi, Z. Xu, T. Simon, M. Nießner, E. Tretschk, L. Liu, B. Mildenhall, P. Srinivasan, R. Pandey, S. Orts-Escolano, S. Fanello, M. G. Guo, G. Wetzstein, J. y Zhu, C. Theobalt, M. Agrawala, D. B. Goldman, and M. Zollhöfer, "Advances in neural rendering," *Computer Graphics Forum*, vol. 41, 2021.

[9] F. Lukačević, S. Škec, M. M. Perišić, N. Horvat, and M. Štorga, "Spatial perception of 3d cad model dimensions and affordances in virtual environments," *IEEE Access*, vol. 8, pp. 174 587–174 604, 2020.

[10] N. Horvat, Stankokec, T. Martinec, F. Lukaevic, and M. M. Perisic, "Comparing virtual reality and desktop interface for reviewing 3d cad models," *Proceedings of the Design Society: International Conference on Engineering Design*, 2019.

[11] N. Horvat, S. Škec, T. Martinec, F. Lukačević, and M. M. Perišić, "Comparing virtual reality and desktop interface for reviewing 3d cad models," *Proceedings of the Design Society: International Conference on Engineering Design*, vol. 1, no. 1, p. 1923–1932, 2019.

[12] K. A. Szczurek, R. M. Prades, E. Matheson, J. Rodriguez-Nogueira, and M. D. Castro, "Multimodal multi-user mixed reality human–robot interface for remote operations in hazardous environments," *IEEE Access*, vol. 11, pp. 17 305–17 333, 2023.

[13] G. Lunghi, R. Marín, M. di Castro, A. Masi, and P. J. Sanz, "Multimodal human-robot interface for accessible remote robotic interventions in hazardous environments," *IEEE Access*, vol. 7, pp. 127 290–127 319, 2019.

[14] L. Gong, J. Berglund, Å. Fast-Berglund, B. J. E. Johansson, Z. Wang, and T. Börjesson, "Development of virtual reality support to factory layout planning," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, pp. 1–11, 2019.

[15] K. Li, R. Bacher, W. P. Leemans, and F. Steinicke, "Towards robust exocentric mobile robot tele-operation in mixed reality," 2022.

[16] K. Li, T. Rolff, S. Schmidt, R. Bacher, S. Frintrop, W. P. Leemans, and F. Steinicke, "Immersive neural graphics primitives," *ArXiv*, vol. abs/2211.13494, 2022.

[17] K. Li, T. Rolff, S. Schmidt, R. Bacher, W. P. Leemans, and F. Steinicke, "Interacting with neural radiance fields in immersive virtual reality," *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023.

[18] A. Goshaw, "The atlas experiment at the cern large hadron collider," *Bulletin of the American Physical Society*, vol. 75, 2008.

[19] A. Dehne, R. Bacher, T. Hermes, and N. Moeller, "Marwin: A mobile autonomous robot for maintenance and inspection," 2017.

[20] H. Bae, M. Golparvar-Fard, and J. White, "High-precision vision-based mobile augmented reality system for context-aware architectural, engineering, construction and facility management (aec/fm) applications," *Visualization in Engineering*, vol. 1, pp. 1–13, 2013.

[21] L. Angrisani, P. Arpaia, D. Gatti, A. Masi, and M. D. Castro, "Augmented reality monitoring of robot-assisted intervention in harsh environments at cern," *Journal of Physics: Conference Series*, vol. 1065, 2018.

[22] S. Zhang, "High-speed 3d shape measurement with structured light methods: A review," *Optics and Lasers in Engineering*, 2018.

[23] M. Tancik, V. Casser, X. Yan, S. Pradhan, B. Mildenhall, P. P. Srinivasan, J. T. Barron, and H. Kretzschmar, "Block-nerf: Scalable large scene neural view synthesis," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8238–8248, 2022.

[24] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5835–5844, 2021.

[25] N. Deng, Z. He, J. Ye, B. Duinkharjav, P. Chakravarthula, X. Yang, and Q. Sun, "Fov-nerf: Foveated neural radiance fields for virtual reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, pp. 3854–3864, 2021.

[26] V. Lazova, V. Guzov, K. Olszewski, S. Tulyakov, and G. Pons-Moll, "Control-nerf: Editable feature volumes for scene rendering and manipulation," *arXiv preprint arXiv:2204.10850*, 2022.

[27] A. Haque, M. Tancik, A. Efros, A. Holynski, and A. Kanazawa, "Instruct-nerf2nerf: Editing 3d scenes with instructions," 2023.

[28] C. Jambon, B. Kerbl, G. Kopanas, S. Diolatzis, T. Leimkühler, and G. Drettakis, "Nerfshop: Interactive editing of neural radiance fields"," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 6, no. 1, May 2023. [Online]. Available: https://repo-sam.inria.fr/fungraph/nerfshop/

[29] M. Tancik, E. Weber, E. Ng, R. Li, B. Yi, J. Kerr, T. Wang, A. Kristoffersen, J. Austin, K. Salahi, A. Ahuja, D. McAllister, and A. Kanazawa, "Nerfstudio: A modular framework for neural radiance field development," *arXiv preprint arXiv:2302.04264*, 2023.

[30] S. Li, H. Li, Y. Wang, Y. Liao, and L. Yu, "Steernerf: Accelerating nerf rendering via smooth viewpoint trajectory," *ArXiv*, vol. abs/2212.08476, 2022.

[31] E. A. Bier, M. C. Stone, K. A. Pier, W. Buxton, and T. DeRose, "Toolglass and magic lenses: the see-through interface," *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, 1993.

[32] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann, "A survey on interactive lenses in visualization," in *Eurographics Conference on Visualization*, 2014.

[33] L. Wang, Y. Zhao, K. Mueller, and A. E. Kaufman, "The magic volume lens: an interactive focus+context technique for volume rendering," *VIS 05. IEEE Visualization, 2005.*, pp. 367–374, 2005.

[34] J. Viega, M. Conway, G. H. Williams, and R. F. Pausch, "3d magic lenses," in *ACM Symposium on User Interface Software and Technology*, 1996.

[35] L. D. Brown and H. Hua, "Magic lenses for augmented virtual environments," *IEEE Computer Graphics and Applications*, vol. 26, pp. 64–73, 2006.

[36] D. Barčević, C. Lee, M. A. Turk, T. Höllerer, and D. A. Bowman, "A hand-held ar magic lens with user-perspective rendering," *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 197–206, 2012.

[37] K. Akeley, "Realityengine graphics," 2000.

[38] R. Li, M. Tancik, and A. Kanazawa, "Nerfacc: A general nerf acceleration toolbox," *ArXiv*, vol. abs/2210.04847, 2022.

[39] J.-Y. Lee, P.-H. Han, L. Tsai, R.-D. Peng, Y.-S. Chen, K.-W. Chen, and Y.-P. Hung, "Estimating the simulator sickness in immersive virtual reality with optical flow analysis," *SIGGRAPH Asia 2017 Posters*, 2017.

[40] A. F. Seay, D. M. Krum, L. F. Hodges, and W. Ribarsky, "Simulator sickness and presence in a high fov virtual environment," *Proceedings IEEE Virtual Reality 2001*, pp. 299–300, 2001.

[41] R. Du, E. Turner, M. Dzitsiuk, L. Prasso, I. Duarte, J. Dourgarian, J. Afonso, J. Pascoal, J. Gladstone, N. Cruces, S. Izadi, A. Kowdle, K. Tsotsos, and D. Kim, "Depthlab: Real-time 3d interaction with depth maps for mobile augmented reality," in *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '20. ACM, 2020, pp. 829–843.

[42] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4104–4113, 2016.

[43] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[44] G. Osterberg, "Topography of the layer of rods and cones in the human retina," *Journal of the American Medical Association*, vol. 108, no. 3, pp. 232–232, 01 1937.

[45] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.

[46] J. Brooke, "Sus: a "quick and dirty"usability scale," *Usability evaluation in industry*, vol. 189, no. 3, 1996.

[47] T. W. Schubert, "The sense of presence in virtual environments: A three-component scale measuring spatial presence, involvement, and realness." *Z. für Medienpsychologie*, vol. 15, no. 2, pp. 69–71, 2003.

[48] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness," *The international journal of aviation psychology*, vol. 3, no. 3, pp. 203–220, 1993.

[49] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. Spie, 1992, pp. 586–606.