360° 3D Photos from a Single 360° Input Image

Manuel Rey-Area and Christian Richardt

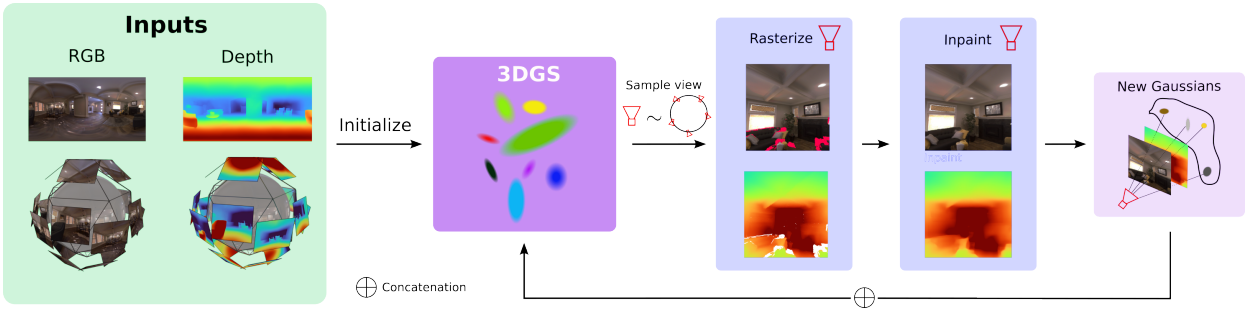

Fig. 1: Our method upgrades 360° images to free-viewpoint renderings with six degrees-of-freedom. We train a 3D Gaussian point cloud to represent the input 360° image and iteratively insert novel content in previously unseen regions to fill disoccluded content. This produces more comfortable and immersive VR viewing experiences.

Abstract— 360° images are a popular medium for bringing photography into virtual reality. While users can look in any direction by rotating their heads, 360° images ultimately look flat. That is because they lack depth information and thus cannot create motion parallax when translating the head. To achieve a fully immersive VR experience from a single 360° image, we introduce a novel method to upgrade 360° images to free-viewpoint renderings with 6 degrees of freedom. Alternative approaches reconstruct textured 3D geometry, which is fast to render but suffers from visible reconstruction artifacts, or use neural radiance fields that produce high-quality novel views but too slowly for VR applications. Our 360° 3D photos build on 3D Gaussian splatting as the underlying scene representation to simultaneously achieve high visual quality and real-time rendering speed. To fill plausible content in previously unseen regions, we introduce a novel combination of latent diffusion inpainting and monocular depth estimation with Poisson-based blending. Our results demonstrate state-of-the-art visual and depth quality at rendering rates of 105 FPS per megapixel on a commodity GPU.

Index Terms—Novel-view synthesis, inpainting, real time

1 Introduction

360° images can capture a complete scene in a single snapshot. However, they look flat in virtual reality (VR), where they are usually rendered as a simple textured sphere or cube map that lacks important depth cues such as binocular disparity or motion parallax [55]. We present a novel method to make 360° images fully immersive by giving them depth using monocular depth estimation, and filling in previously unseen regions that become visible as a user looks around the scene and behind objects. Our resulting 360° 3D photos can be freely explored in six degrees-of-freedom (6DoF), including arbitrary 3D head translation and rotation.

Most approaches for capturing scenes as 360° VR experiences in a single shot require multi-view capture of the scene using a camera rig [34] with as many as 16 [1, 49], 46 [6] or even 60 cameras [73], which is not practical for casual users. Alternative approaches reconstruct a scene representation from a 360° input video of a moving camera under the assumption that the scene is static [5, 7–9, 20, 21, 25, 30, 35], which is not always robust. Our approach enables the creation of 360° 3D photos from a single 360° input image, which is the most practical option for casual users. This makes our approach beneficial to a variety of VR applications, such as virtual tourism, real estate, or education.

Existing single-image 3D photo techniques [32, 33, 59] turn a single (perspective) input image into a compelling 3D photo by addressing three key challenges: 1) 3D reconstruction from a monocular image to explain the 3D structure of the underlying scene, 2) 3D scene completion to fill in plausible content in unseen and missing parts, and 3) real-time rendering for free-viewpoint 6-DoF novel-view synthesis. In our work, we address these three challenges for a single 360° input image, so they can be experienced in VR.

No current approach solves all three challenges jointly for 360° images. Xu et al. [69] use the estimated layout of an indoor scene for reconstructing 3D scene structure, which limits the level of scene detail and the type of scenes that are supported. PanoSynthVR [66] extends multi-plane images to multi-cylinder images, which enables real-time rendering, but severely limits inpainting performance. Most recently, Wang et al. [67] optimize a panoramic neural radiance field (PERF) from a 360° image by iteratively inpainting and refining novel 360° RGBD views. However, the geometry filling strategy conflicts with the used 3D representation as they are decoupled. Rendering novel views is also slow due to the underlying NeRF backbone (Instant-NGP).

Our approach leverages 3D Gaussian splatting [28] to efficiently solve all three challenges. We start by estimating a depth map for the 360° input image [53], and projecting the resulting 360° RGBD image into 20 perspective RGBD images corresponding to the faces of an icosahedron, shown to be one of the least distorted approximations of the sphere [12]. We then train an initial scene reconstruction using depth-supervised 3D Gaussian splatting from these 20 views. We then iteratively sample novel virtual viewpoints, to inpaint any remaining holes using a pre-trained latent diffusion model and a monocular depth estimator, and fine-tune the scene model using the initial and inpainted novel views. We use the rendered 3D Gaussian splats to allow for soft inpainting, where the continuous opacity determines how soft or strong content needs to be filled in 3D regions. Our main contributions are:

1. A novel approach for upgrading single 360° images to immersive 3D photos that can be experienced in virtual reality.

- *Manuel Rey-Area is with the University of Bath. E-mail: mra59@bath.ac.uk.*
- *Christian Richardt is with Meta Reality Labs. E-mail: crichardt@meta.com.*

All experiments, including all dataset and model access and use, were run exclusively at the University of Bath by University of Bath researchers.

2. A scene representation based on 3D Gaussian splatting that dynamically refines occluded areas as new views become available.
3. A new soft inpainting technique for RGBD images that combines latent diffusion with monocular depth estimation, and harmoniously blends the results during scene reconstruction.

2 Related Work

Most techniques for synthesising novel views for VR applications expect multi-view image or video input [55], which is hardly practical for casual users. There are also single-image novel-view synthesis methods, but most only support perspective images, which severely limits the available field of view in VR compared to 360° images.

2.1 Novel-View Synthesis for Virtual Reality

Conceptually the simplest form of view synthesis is directly interpolating between the nearest subset of views, e.g. using a basic proxy geometry [4], optical flow fields [41], or view-dependent geometry [18,48]. This view interpolation works best if there are many available viewpoints, ideally tens to hundreds. On the other hand, novel-view synthesis using a single 360° camera generally assumes a static scene captured by a moving camera [5, 7, 21], such that subsequent frames capture different viewpoints of the same scene. In contrast, our approach only needs a single 360° image as input.

A wide variety of intermediate representations have been applied for VR novel-view synthesis. One of the earliest is omnidirectional stereo [1,31,43,50,54], the standard format for streaming 360° stereo videos. However, the format is lacking depth for rendering motion parallax; it only supports looking sideways. While textured meshes [25,34,49,58] enable view synthesis with full 6-DoF parallax, they tend to lack visual fidelity due to flat surfaces and hard triangle edges. Softer view synthesis is enabled by multiplane images [14,44,74], which are stacked parallel image planes with transparency. This limits multiplane images to forward-facing scenes. 360° views can be achieved with multicylinder images [66], optionally with per-layer depth maps [39], or multisphere images [2,6,46]. However, the key limitation of multi-layer images is the rather small region of motion before the view synthesis quality degrades.

Recently, neural radiance fields (NeRFs) [45] have reinvigorated the field of novel-view synthesis due to their high-quality results and conceptual simplicity. Mip-NeRF 360 [3] showed the first results on 360° scenes, although the camera is always inward-facing. VR-NeRF [70] demonstrated the first end-to-end NeRF system for real-time VR rendering. SMERF [11] also achieves real-time rendering of NeRFs using a clever baking scheme. Most similar to our approach is PERF [67], which synthesizes novel 360° RGBD views that help train a NeRF starting from a single panorama. 3D Gaussian splatting [28] delivers faster training and rendering times than most NeRFs techniques. This is especially attractive for VR applications, such as dynamics-aware interactions [27] or text-driven scene generation [10, 36, 42]. Our approach also benefits from the rendering speed of 3D Gaussian splatting, as well as its advantageous optimization properties.

2.2 Single-Image View Synthesis

The key idea of lifting a single input image into the 3D realm has been approached in many different ways. SynSin [68] projects the input image to a feature point cloud that can be rendered from any view, and postprocessed to create a color image using a CNN. Worldsheet [19] shrink-wraps the scene with a mesh that is textured to enable novel-view synthesis from arbitrary viewpoints. The meshes can also be cut into multiple layers [33, 59], which enables higher quality occlusions but requires inpainting of previously unseen regions. Alternatively, multiplane images can provide smooth view synthesis for a limited region of viewpoints [24,29,65]. Using a learned image encoder, pixelNeRF [72] can create novel views from a single input image in a feed-forward fashion. However, these approaches are limited to perspective images and do not work correctly for 360° images due to the different image projection model.

Recent diffusion-based approaches try to jointly solve the 3D reconstruction and scene completion problems by fine-tuning a pretrained latent diffusion model [16,56] to synthesize increasingly high-quality novel views of objects [17,22,40,62–64,75] or scenes [15,26,47,61] from a single perspective input image. Methods that work at a scene level generally follow the three stages of incremental synthesis, alignment and refinement. Invisible Stitch [13], Text2Room [23] and LucidDreamer [10] use pretrained latent diffusion models and monocular depth estimation for outpainting RGBD scene content as a textured mesh, a point cloud and 3D Gaussian splats, respectively. These methods extend a single perspective image into panoramic scenes, which differs from our goal of converting an existing 360° image into a 360° 3D photo.

3 Method

The core of our method is an iterative Gaussian point cloud completion approach that is outlined in Fig. 1. We start from a single 360° image and estimate its depth. Next, we project the input into 20 perspective RGBD images and initialize a preliminary set of 3D Gaussian splats [28] to represent the underlying scene that explains the input (Sec. 3.2). Moving away from the centre of projection reveals empty areas that need filling. We sample virtual novel views inside an action radius and inpaint both color and depth in the empty regions (Sec. 3.3). We add this inpainted content to the preliminary point cloud as new 3D Gaussians and optimize them to match the inpainted and original input images.

3.1 Initial Setting

The input of our method is a single 360° image I_{360}. First, we estimate its depth map D_{360} using an off-the-shelf 360° monocular depth estimator. Specifically, we use 360MonoDepth [53] for its state-of-the-art performance in indoor and outdoor scenes.

We project the resulting 360° RGBD image onto an icosahedron as perspective RGBD images with poses $\{E_i\}_{i=1}^{20}$, where $E_i = [R_i \,|\, \mathbf{0}] \mid R_i \in \mathbb{R}^{3\times3}, \mathbf{0} \in \mathbb{R}^3$ and shared intrinsics $K \in \mathbb{R}^{3\times3}$, resulting in a set of initial color images $\{I_i\}_{i=1}^{20} \in \mathbb{R}^{H\times W\times3}$ and depths $\{D_i\}_{i=1}^{20} \in \mathbb{R}^{H\times W}$.

360MonoDepth outputs depth maps using Euclidean distance or ray length. For rasterization, 3D Gaussian splatting expects depth as the z-component of the ray, where flat regions perpendicular to the view direction share the same depth. However, Euclidean distance increases with distance from the principal point, making it incompatible with 3D Gaussian splatting. Therefore, we convert the depth maps D_i from Euclidean distance to z-depth using

$$d_z = \frac{r}{\|\mathbf{p}-\mathbf{u}\|^2}. \tag{1}$$

Here, r is the Euclidean distance of a pixel, and $\|\mathbf{p}-\mathbf{u}\|^2$ represent the distance of a pixel $\mathbf{u}$ to the principal point $\mathbf{p}$.

3.2 3D Gaussian Splatting Preliminaries

3D Gaussian splatting [28] represents the underlying space from a set of calibrated images as 3D Gaussians parameterized by their centre $x \in \mathbb{R}^3$, spherical harmonics (SH) coefficients $c \in \mathbb{R}^D$, an opacity value $\alpha \in \mathbb{R}$, a rotation vector represented as a quaternion $q \in \mathbb{H}$, and a scaling vector $s \in \mathbb{R}^3$. Upon projecting the 3D Gaussians into 2D splats, the color C of a pixel is computed via volumetric rendering, using front-to-back depth ordering:

$$C = \sum_{j\in N} c_j \alpha_j T_j, \tag{2}$$

where N is the set of ordered Gaussian indices, and $T_j = \prod_{k=1}^{j-1}(1-\alpha_k)$ is the transmittance, defined as the accumulated transparency for Gaussians overlapping the same pixel. Similarly, we render the accumulated depth using

$$D = \sum_{j\in N} d_j \alpha_j T_j, \tag{3}$$

where $d_j = (Rx_j)_z$ is the depth of each splat from the camera.

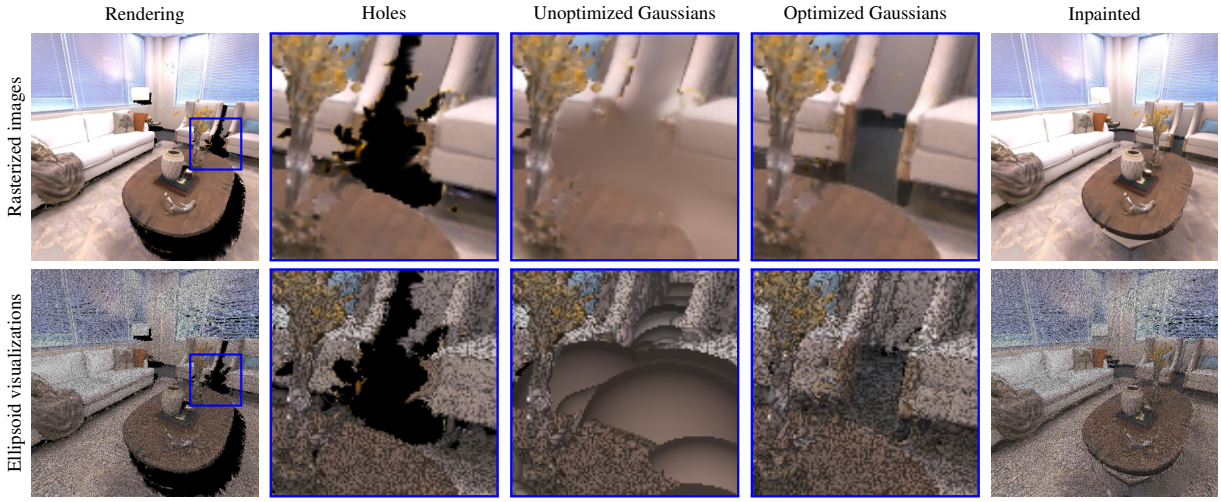

Fig. 2: Overview of our iterative Gaussian point cloud completion. Rasterizing a novel view from the preliminary point cloud results in renderings with holes. We insert unoptimized Gaussians in empty regions and iteratively refine them to match the inpainted image.

To supervise training, we use the default loss from 3D Gaussian splatting consisting of an L1 loss and D-SSIM for the RGB renders. We also supervise the rendered depths using an additional L1 loss:

$$L = \lambda_{\text{RGB}} L_1(I_i, C) + (1 - \lambda_{\text{RGB}}) L_{\text{D-SSIM}}(I_i, C) + \lambda_D L_1(D_i, D), \quad (4)$$

using weights $\lambda_{\text{RGB}} = 0.8$ and $\lambda_{\text{D}} = 1.3$. The depth loss serves as a 3D prior to encourage geometric consistency and avoid trivial solutions like a flat sphere.

3.3 Iterative Scene Completion

The preliminary Gaussian point cloud represents a partial observation of the underlying 3D scene. Rendering the scene from a new viewpoint that is different from the original reveals empty areas with no content (see Fig. 2). Thus, we aim to fill the entire empty space seen by any view inside an action sphere of radius a. At each iteration, we randomly sample views placed on the sphere surface with intrinsics K and extrinsics $E = [R \,|\, \mathbf{C}]$:

$$\mathbf{C} = \frac{\mathbf{x} \cdot a}{\|\mathbf{x}\|^2}, \quad (5)$$

where $\mathbf{x} \sim \mathcal{N}(0, \mathbf{I})$. We construct the random rotations R by sampling longitude θ and latitude ϕ angles from uniform distributions:

$$\theta \sim \mathcal{U}(-\pi, \pi) \quad \text{and} \quad \phi \sim \mathcal{U}\left(-\frac{\pi}{2}, \frac{\pi}{2}\right). \quad (6)$$

Next, we rasterize the 3D Gaussians to obtain the rendered image I and depth map D, and a soft mask m that depicts pixels' transmittance. We binarize the transmittance mask using a threshold of 0.05, unless stated otherwise.

The rendered image I has empty regions or holes (Fig. 2) that need inpainting. We leverage a text-to-image inpainting latent diffusion model $\mathscr{F}_{\text{t2i}}$ to inpaint unobserved pixels:

$$I_{\text{inp}} = \mathscr{F}_{\text{t2i}}(I, m, t), \quad (7)$$

where t is a text prompt.

The rendered depth map D is also incomplete. Unfortunately, state-of-the-art dense depth inpainting models fall short at preserving high-frequency and sharp details. Therefore, we feed the inpainted image I_{inp} to a powerful perspective monocular depth estimator [71] to estimate its depth map $\tilde{D}$.

The output depth map $\tilde{D}$ matches the scene up to an unknown scale factor and offset. We find the scale s and offset o that best align $\tilde{D}$ with D via least squares [52]. Specifically, we use only the opaque values from m, namely $\neg m$, where information exists in D.

$$s, o = \arg\min_{s,o} \sum_{p \in \neg m} (s\tilde{D}_p + o - D_p)^2, \quad (8)$$

$$D_{\text{inp}} = s\tilde{D} + o. \quad (9)$$

3.3.1 Seamless RGBD Blending

Both the inpainted image I_{inp} and the aligned depth map D_{inp} result in a prediction with novel content across the entire pixel grid. However, content outside of holes is already known and has been optimized to match the initial 360 image (Sec. 3.2), so we want to preserve it as much as possible. To do this, we seamlessly blend the rendered images I and depth maps D with their respective inpainted versions I_{inp}, D_{inp} by solving a Poisson equation with Dirichlet boundary conditions [51]. This results in I_{b} and D_{b}. Figure 3 shows the smooth transitions between existing and inpainted content.

Critically, we assume that depth in empty regions must be further than its surrounding foreground depth. However, Poisson blending does not guarantee this. We therefore perform near-depth clipping on the blended depth D_b that forces depth values in empty regions to be behind the smallest enclosing ring of foreground depth δ_Ω:

$$D_b = \begin{cases} D_b, & \text{if } m = 0 \\ \max(D_b, D_b|_{\delta_\Omega}) & \text{if } m = 1. \end{cases} \quad (10)$$

Here, '$D_b|_{\delta_\Omega}$' represents the depth values of D_b at the points on the enclosing ring δ_Ω.

3.3.2 Gaussian Insertion

As discussed, the image I_{b} and depth D_{b} have been optimized to faithfully match old content in known regions while synthesizing plausible color and geometry inside holes. However, our 3D scene representation is not yet aware of this content. We convert novel content to 3D Gaussians and add them to the point cloud. The new Gaussian 3D means are computed like:

$$\mathbf{x} = E^{-1} K^{-1} \tilde{p}_d. \quad (11)$$

Where $\tilde{p}_d$ is the augmented homogeneus coordinate of a 2D pixel with its corresponding depth $\tilde{p}_d = (x, y, D_{\text{b}}(x, y))$. The Gaussian color $\mathbf{c}$ is fetched from I_{b}. We initialize Gaussian rotations $\mathbf{q}$ as unit quaternions. Scale s is initialized as the distance to its closest neighbours. Finally, we add the new virtual image to the training set consisting of the 20 original training cameras and a growing set of novel virtual views.

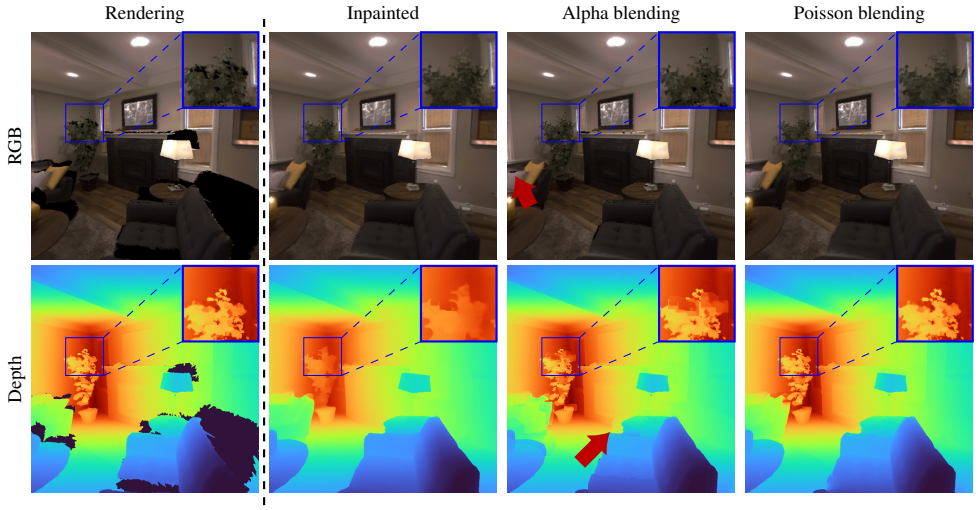

Fig. 3: Using latent inpainting models result in reconstructions that are not identical to the incomplete rendering outside holes. Naive alpha blending reuses outside content but leaves holes and non smooth transitions. Poisson blending smoothly interpolates between old and new content without leaving holes nor visible seams.

4 Experiments and Results

4.1 Experiment Settings

Implementation. The size of the input 360 image I_{360} is 2048×1024. We estimate the depth D_{360} at native resolution and initialize a Gaussian point cloud with one Gaussian per pixel. The size of the perspective images I and D is 512×512. We optimize the preliminary 3D Gaussian splatting for 7,000 iterations. We optimize only for level-1 SH harmonics due to the lack of multi-view input. Work on novel-view synthesis [58] argues that head movement in VR interactive exploration is usually limited to 35 cm or less. We use an action sphere of radius $a = 0.5$ (equivalent to 50 cm for scenes with metric depth). To avoid trespassing scene boundaries, we truncate the action sphere if the camera center in Eq. (5) is too close to scene boundaries. We run the Gaussian point cloud completion for 100 iterations. At each iteration, we render a pool of $V = 100$ candidate virtual views and sort them reversely by their transmittance. We choose the top-1 ranked view. This way we always encourage filling the bigger gaps. Each virtual view is optimized for 1000 iterations. During the iterative process, initial views $\{E_i\}_{i=1}^{20}$ start with a higher chance of being selected. We monotonously increase the sampling weight of virtual poses until equal weight is achieved. We observed that resetting opacity was harmful in the overall method, as newly added Gaussians were incorrectly culled when virtual views had lower chance of being selected. We instead chose a less-severed opacity control mechanism [57] where opacity is periodically decreased by 0.001. For the latent diffusion inpaint model we found that using the prompt *empty* with a strength of 0.9 works best. Training a scene usually takes around 50 minutes in a single RTX 3090 with 24GB of memory.

Datasets. We conduct experiments on two datasets. The Replica dataset [60] consists of 12 synthetic indoor scenes. We render a consistent camera trajectory path with 255 poses traversing the room. We report qualitative and quantitative results. We also evaluate the proposed method and baselines on in-the-wild data from the OmniPhotos dataset. It contains 30 scenes of real outdoors footage. Each scene has only one single 360° color image and no ground-truth depth map.

Evaluation Metrics. We use the three standard image quality metrics to evaluate our results: PNSR, SSIM and LPIPS. We evaluate depth quality using standard metrics for monocular depth estimation. Absolute Relative Difference (Abs Rel), Root Mean Squared Error (RMSE) and delta inliers $\delta < 1.25$. We report VR related metrics like FPS per megapixel (FPS/MP). VR aims to render 4 megapixels per eye at interactive speed. Additionally, we report VMAF[1] [37] as a perceptual video quality assessment metric, which demonstrates the strongest correlations with human perceptual assessments [38].

Baselines. We compare our results to Text2Room [23] and PERF [67]. The former generates textured 3D meshes from a text prompt. We modified to initialize the textured mesh from a 360 RGBD input. PERF is a single panorama neural radiance field method. For fair comparison, all methods are provided with the same 360 RGBD input and evaluated with identical camera trajectory.

4.2 Quantitative Evaluation

Tab. 1 shows the quantitative results in Replica. Our method achieves the best performance in image quality metrics, depth quality metrics and FPS/MP. The metrics validate some advantages of our method with respect to the baselines. We aim to reuse known content from the preliminary 3D Gaussian splatting to the greatest extent. This makes supervision to not conflict. The Gaussians that contribute to non-empty regions always see the same content regardless of the viewing position. Text2Room and PERF also aim to reuse the input when possible. However, their inpainting masks are estimated with *ad-hoc* strategies that fall short with complex scenes. We just use the disocclusion of Gaussians which were optimized to fit an initial image. Fig. 4 compares VMAF video quality metrics for all Replica scenes. Our method scores the highest average VMAF in 10 out of 12 scenes while consistently achieving the lowest standard deviation over time. This highlights that our method not only produces high-quality renderings, but also has the best temporal performance compared to PeRF and Text2Room.

4.3 Qualitative Evaluation

Fig. 5 shows qualitative results for all the methods on all datasets. Our method reconstructs existing content with the highest consistency. Overall our results show fewer errors in thin structures. On Replica, our results have sharper object contours. Text2Room incorrectly removes objects with thin structures while PeRF deforms the contours of the foreground. Our approach also shows less blurry transitions between inpainted depth and rendered depth. Text2Room struggles in regions with large holes. On the other side, PeRF depth renderings do not align accurately. Results on in-the-wild examples from OmniPhotos (with additional examples in Fig. 6), show that our approach clearly outperforms at inpainting content at sides of foreground outline compared with the baselines. Sec. 4.3 shows the best, median and worst frames on the test trajectory in terms of PSNR for three scenes of Replica.

[1] Video Multimethod Assessment Fusion

Table 1: Quantitative results for Replica360-2K, evaluated at 512×512 resolution. Highlighting: best, second-best.

Method	Visual quality			Depth quality					Speed
	PSNR ▲	SSIM ▲	LPIPS ▼	AbsRel ▼	RMSE ▼	$\delta < 1.25$ ▲	$\delta < 1.25^2$ ▲	$\delta < 1.25^3$ ▲	FPS/MP ▲
Text2Room [23]	28.018	0.887	0.066	0.008	0.034	0.990	0.995	0.997	0.78
PeRF [67]	29.181	0.913	0.058	0.208	0.321	0.633	0.996	0.998	0.2
Ours	31.192	0.925	0.050	0.004	0.007	0.997	0.999	0.999	105

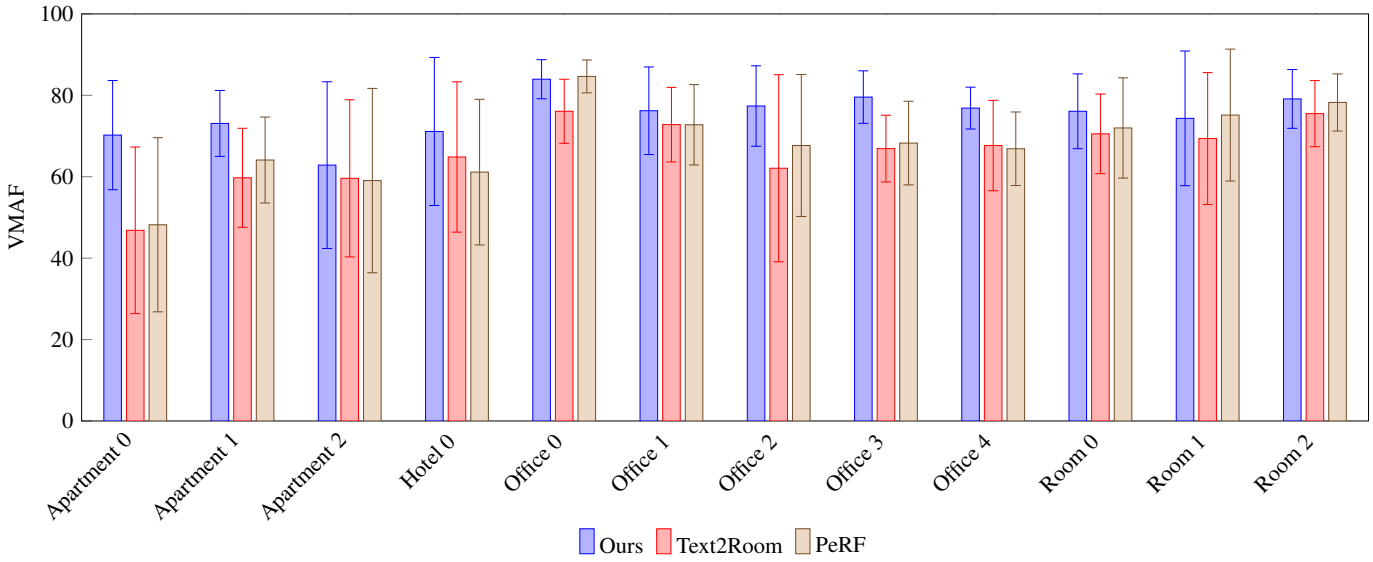

Fig. 4: Video perceptual quality for the 12 scenes of Replica [60]. Each bar shows the average VMAF across 255 test frames. The overlaid error bar indicates the standard deviation. Our method shows the best VMAF in 10 out of 12 scenes, and is a close second in the remaining 2 scenes.

The worst PSNR belongs to viewpoints that reveal large disocclusions. Figure 1, Figure 2 and Figure 3 in the supplemental material show the distribution of image quality metrics for the three corresponding Replica scenes using our method and baselines.

4.4 Ablation Studies

We perform several ablation studies to justify the design choices of our training strategy for the iterative Gaussian point cloud completion. The ablations are summarised in Tab. 2 and Fig. 7.

Single virtual view. We reduce the pool of virtual views from 100 to 1 when sampling novel views. With our approach, we ensure empty space is always filled first.

w/o training views. The initial 20 views are discarded after training the preliminary scene model. We show that using them during the scene completion stage helps anchor the scene and avoids drifting during the 3D Gaussian splatting optimization.

Opacity reset. The original Gaussian splatting implementation periodically resets opacity. We show that this harms training as the optimization adds fuzzy Gaussians in empty space to explain certain views.

No blending. We use the inpainted RGBD reconstruction of Eq. (9) directly as supervision for virtual views. We notice that the noisy nature of the latent space compression makes convergence difficult. Eq. (9) does not guarantee that depth values in holes are behind the surrounding ones, resulting in ghosting artifacts due to conflicting supervision between initial and virtual views.

4.5 Limitations

Our method does not come without limitations (see Fig. 9). The generative nature of diffusion models can result in inpainting erratic content. This translates in wrong depth estimates which corrupts 3D Gaussian splatting. Gaussians placed at foreground contours might bleed into the background causing outline artifacts. Our method show artifacts when rendered from viewpoints close to scene boundaries. This is an inherent problem of 3D Gaussian splatting which some works address. The lack of real-world annotated datasets makes quantitative evaluation only possible in synthetic data. Although reporting VMAF in synthetic data somewhat demonstrates human preference, subjective studies with human participants would be beneficial for a more thorough real-world evaluation.

5 Discussion

We show results in a bounded sphere where the user is expected to move. However, in a real VR setting, user motion is unconstrained and can lead to much larger movements. Inpainting larger holes first may result in a lack of context for the diffusion model, potentially causing large semantic gaps between existing and inpainted content. Also, our method relies on accurate depth maps for supervision. Incorrect depths can cause disocclusions to manifest in regions where they should not, resulting in erratic inpainted content.

6 Conclusion

Our proposed method is the first to achieve high-quality renderings of indoor and outdoor scenes at high frame rates. Using 3D Gaussian splatting as our scene representation with the novel combination of latent diffusion inpainting and Poisson-based blending, we demonstrate the highest quantitative performance at rates of 105 FPS per megapixel. Our method is an important step towards enabling free-viewpoint rendering of casually captured 360° images. As future work, investigating inpainting empty space in a more principled way, without drifting, could lead to inpainted content that is not overly hallucinated.

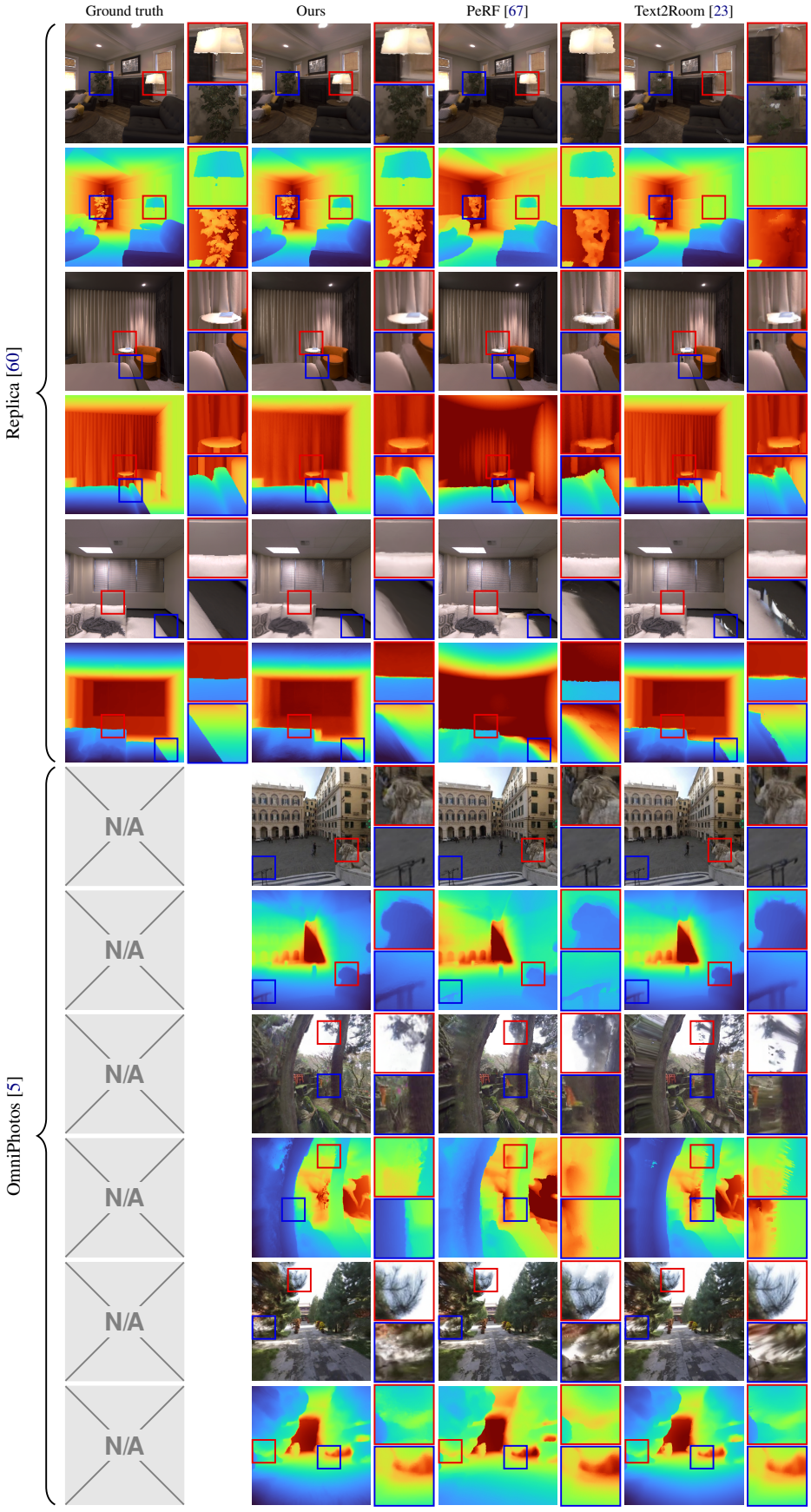

Fig. 5: Qualitative comparison to different methods on different datasets. Our results show the highest level of detail of all predictions.

Table 2: Ablations evaluated on Replica360-2K. Highlighting: best, second-best.

Method	PSNR ▲	SSIM ▲	LPIPS ▼	AbsRel ▼	RMSE ▼	$\delta < 1.25$ ▲	$\delta < 1.25^2$ ▲	$\delta < 1.25^3$ ▲
Ours (Single virtual view)	30.988	0.925	0.051	0.005	0.007	0.996	0.998	0.999
Ours (w/o training views)	28.017	0.890	0.093	0.008	0.017	0.992	0.996	0.998
Ours (Opacity Reset)	27.621	0.892	0.095	0.010	0.015	0.993	0.997	0.999
Ours (No blending)	29.939	0.908	0.075	0.022	0.025	0.985	0.995	0.998
Ours (Full)	31.192	0.925	0.050	0.004	0.007	0.997	0.999	0.999

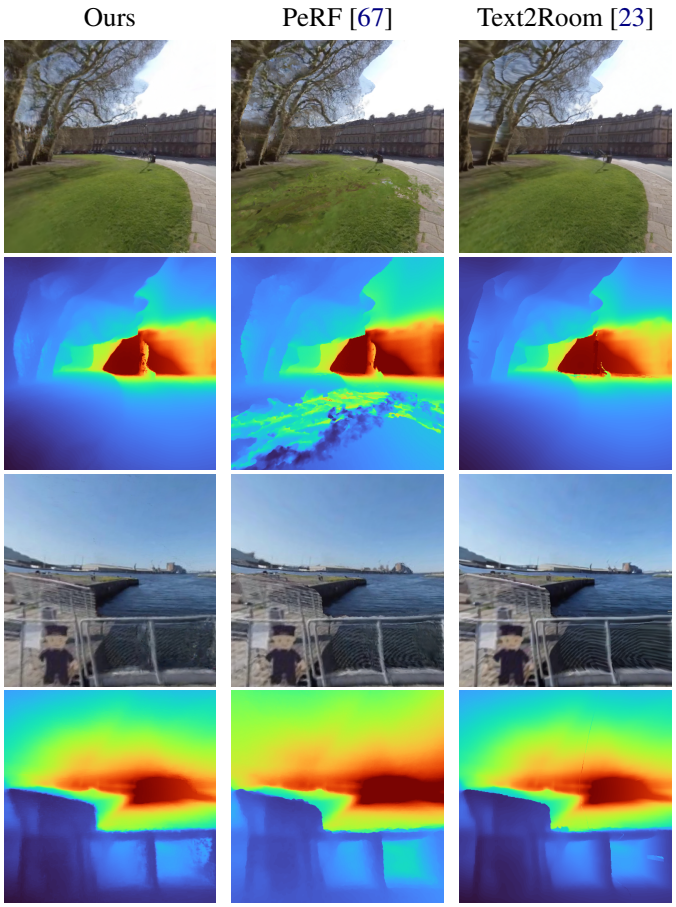

Fig. 6: Qualitative comparison on OmniPhotos outdoor scenes [5].

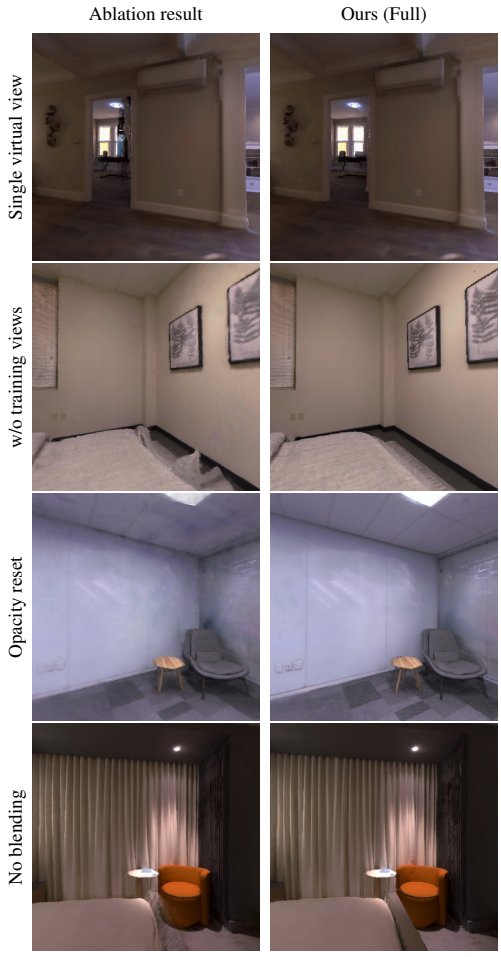

Fig. 7: Common artifacts of tested ablations. Single virtual view doesn't guarantee all holes are filled. The lack of initial training views manifests in wrong generated content. Opacity reset produces fuzzy floaters. No blending causes ghosting artifacts due to non smooth old-new content transition. Scenes from the Replica dataset [60].

Acknowledgments

We would like to thank Wenbin Li for his support.

References

[1] R. Anderson, D. Gallup, J. T. Barron, J. Kontkanen, N. Snavely, C. Hernandez, S. Agarwal, and S. M. Seitz. Jump: Virtual reality video. *ACM Trans. Graph.*, 35(6):198:1–13, 2016. doi: 10.1145/2980179.2980257 1, 2

[2] B. Attal, S. Ling, A. Gokaslan, C. Richardt, and J. Tompkin. MatryODShka: Real-time 6DoF video view synthesis using multi-sphere images. In *ECCV*, 2020. doi: 10.1007/978-3-030-58452-8_26 2

[3] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022. doi: 10.1109/CVPR52688.2022.00539 2

[4] T. Bertel, N. D. F. Campbell, and C. Richardt. MegaParallax: Casual 360° panoramas with motion parallax. *TVCG*, 25(5):1828–1835, 2019. doi: 10.1109/TVCG.2019.2898799 2

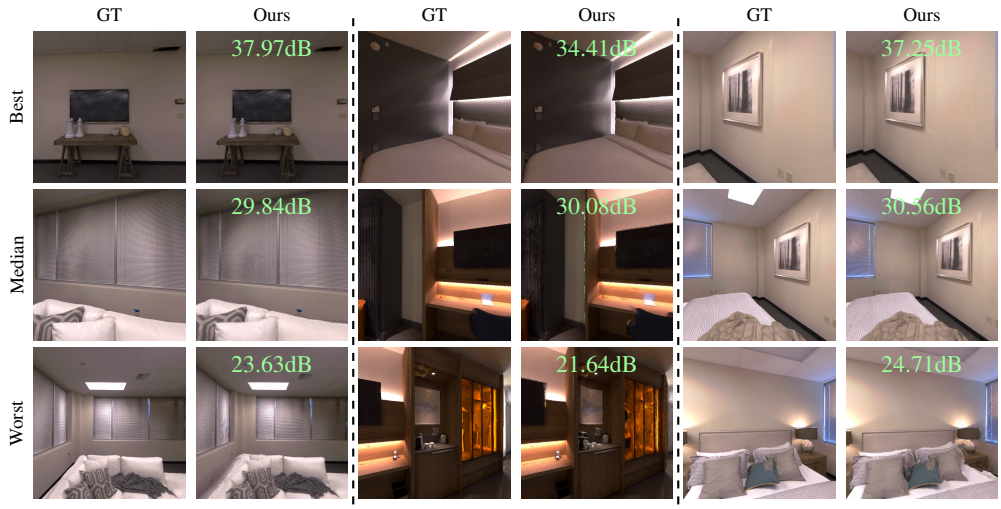

Fig. 8: Best, median and worst frames of the test trajectory for three Replica scenes in terms of PSNR.

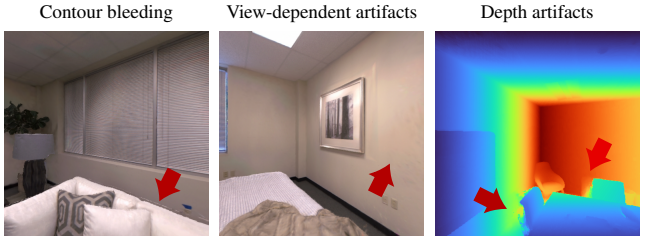

Fig. 9: Limitations of our method. **Left:** In some cases, Gaussians placed at object boundaries bleed to background which produces outline artifacts. **Middle:** Approaching the walls too much results in popping artifacts. **Right:** Sometimes, the inpainting latent diffusion model can hallucinate erratic content resulting in wrong depth estimates. Also, seamless depth blending is not always successfull.

[5] T. Bertel, M. Yuan, R. Lindroos, and C. Richardt. OmniPhotos: Casual 360° VR photography. *ACM Trans. Graph.*, 39(6):267:1–12, 2020. doi: 10.1145/3414685.3417770 1, 2, 6, 7

[6] M. Broxton, J. Flynn, R. Overbeck, D. Erickson, P. Hedman, M. DuVall, J. Dourgarian, J. Busch, M. Whalen, and P. Debevec. Immersive light field video with a layered mesh representation. *ACM Trans. Graph.*, 39(4):86:1–15, 2020. doi: 10.1145/3386569.3392485 1, 2

[7] R. Chen, F.-L. Zhang, S. Finnie, A. Chalmers, and T. Rhee. Casual 6-DoF: free-viewpoint panorama using a handheld 360° camera. *TVCG*, 29(9):3976–3988, 2023. doi: 10.1109/TVCG.2022.3176832 1, 2

[8] C. Choi, S. M. Kim, and Y. M. Kim. Balanced spherical grid for egocentric view synthesis. In *CVPR*, 2023. 1

[9] D. Choi, H. Jang, and M. H. Kim. OmniLocalRF: Omnidirectional local radiance fields from dynamic videos. In *CVPR*, 2024. 1

[10] J. Chung, S. Lee, H. Nam, J. Lee, and K. M. Lee. LucidDreamer: Domain-free generation of 3D Gaussian splatting scenes. arXiv:2311.13384, 2023. 2

[11] D. Duckworth, P. Hedman, C. Reiser, P. Zhizhin, J.-F. Thibert, M. Lučić, R. Szeliski, and J. T. Barron. SMERF: Streamable memory efficient radiance fields for real-time large-scene exploration. *ACM Trans. Graph.*, 43(4):63:1–13, 2024. doi: 10.1145/3658193 2

[12] M. Eder and J.-M. Frahm. Convolutions on spherical images. In *CVPR Workshops*, 2019. 1

[13] P. Engstler, A. Vedaldi, I. Laina, and C. Rupprecht. Invisible stitch: Generating smooth 3D scenes with depth inpainting. arXiv:2404.19758, 2024. 2

[14] J. Flynn, M. Broxton, P. Debevec, M. DuVall, G. Fyffe, R. Overbeck, N. Snavely, and R. Tucker. DeepView: View synthesis with learned gradient descent. In *CVPR*, pp. 2367–2376, 2019. doi: 10.1109/CVPR.2019.00247 2

[15] R. Gao, A. Holynski, P. Henzler, A. Brussee, R. Martin-Brualla, P. Srinivasan, J. T. Barron, and B. Poole. CAT3D: Create anything in 3D with multi-view diffusion models. In *NeurIPS*, 2024. 2

[16] A. Gokaslan, A. F. Cooper, J. Collins, L. Seguin, A. Jacobson, M. Patel, J. Frankle, C. Stephenson, and V. Kuleshov. CommonCanvas: An open diffusion model trained with creative-commons images. In *CVPR*, 2024. 2

[17] J. Gu, A. Trevithick, K.-E. Lin, J. Susskind, C. Theobalt, L. Liu, and R. Ramamoorthi. NerfDiff: Single-image view synthesis with NeRF-guided distillation from 3D-aware diffusion. In *ICML*, 2023. 2

[18] P. Hedman, T. Ritschel, G. Drettakis, and G. Brostow. Scalable inside-out image-based rendering. *ACM Trans. Graph.*, 35(6):231:1–11, 2016. doi: 10.1145/2980179.2982420 2

[19] R. Hu, N. Ravi, A. C. Berg, and D. Pathak. Worldsheet: Wrapping the world in a 3D sheet for view synthesis from a single image. In *ICCV*, 2021. 2

[20] H. Huang, Y. Chen, T. Zhang, and S.-K. Yeung. 360Roam: real-time omnidirectional roaming in large scale indoor scenes. In *SIGGRAPH Asia Technical Communications*, pp. 20:1–5, 2022. doi: 10.1145/3550340.3564222 1

[21] J. Huang, Z. Chen, D. Ceylan, and H. Jin. 6-DOF VR videos with a single 360-camera. In *IEEE VR*, pp. 37–44, 2017. doi: 10.1109/VR.2017.7892229 1, 2

[22] L. Höllein, A. Božič, N. Müller, D. Novotny, H.-Y. Tseng, C. Richardt, M. Zollhöfer, and M. Nießner. ViewDiff: 3D-consistent image generation with text-to-image models. In *CVPR*, 2024. 2

[23] L. Höllein, A. Cao, A. Owens, J. Johnson, and M. Nießner. Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In *ICCV*, 2023. 2, 4, 5, 6, 7

[24] V. Jampani, H. Chang, K. Sargent, A. Kar, R. Tucker, M. Krainin, D. Kaeser, W. T. Freeman, D. Salesin, B. Curless, and C. Liu. SLIDE: Single image 3D photography with soft layering and depth-aware inpainting. In *ICCV*, pp. 12518–12527, 2021. 2

[25] H. Jang, A. Meuleman, D. Kang, D. Kim, C. Richardt, and M. H. Kim. Egocentric scene reconstruction from an omnidirectional video. *ACM Trans. Graph.*, 41(4):100:1–12, 2022. doi: 10.1145/3528223.3530074 1, 2

[26] W. Jang and L. Agapito. NViST: In the wild new view synthesis from a single image with transformers. In *CVPR*, 2024. 2

[27] Y. Jiang, C. Yu, T. Xie, X. Li, Y. Feng, H. Wang, M. Li, H. Lau, F. Gao, Y. Yang, and C. Jiang. VR-GS: A physical dynamics-aware interactive Gaussian splatting system in virtual reality. In *SIGGRAPH*, 2024. doi: 10.1145/3641519.3657448 2

[28] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–14, 2023. doi: 10.1145/3592433 1, 2

[29] N. Khan, L. Xiao, and D. Lanman. Tiled multiplane images for practical 3D photography. In *ICCV*, pp. 10454–10464, 2023. 2

[30] H. Kim, A. Meuleman, H. Jang, J. Tompkin, and M. H. Kim. OmniSDF: Scene reconstruction using omnidirectional signed distance functions and

adaptive binoctrees. In *CVPR*, 2024. 1

[31] R. Konrad, D. G. Dansereau, A. Masood, and G. Wetzstein. SpinVR: Towards live-streaming 3D virtual reality video. *ACM Trans. Graph.*, 36(6):209:1–12, 2017. doi: 10.1145/3130800.3130836 2

[32] J. Kopf, S. Alsisan, F. Ge, Y. Chong, K. Matzen, O. Quigley, J. Patterson, J. Tirado, S. Wu, and M. F. Cohen. Practical 3D photography. In *CVPR Workshops*, 2019. 1

[33] J. Kopf, K. Matzen, S. Alsisan, O. Quigley, F. Ge, Y. Chong, J. Patterson, J.-M. Frahm, S. Wu, M. Yu, P. Zhang, Z. He, P. Vajda, A. Saraf, and M. Cohen. One shot 3D photography. *ACM Trans. Graph.*, 39(4):76:1–13, 2020. doi: 10.1145/3386569.3392420 1, 2

[34] J. Lee, B. Kim, K. Kim, Y. Kim, and J. Noh. Rich360: Optimized spherical representation from structured panoramic camera arrays. *ACM Trans. Graph.*, 35(4):63:1–11, 2016. doi: 10.1145/2897824.2925983 1, 2

[35] L. Li, H. Huang, S.-K. Yeung, and H. Cheng. OmniGS: Fast radiance field reconstruction using omnidirectional gaussian splatting. arXiv:2404.03202, 2024. 1

[36] W. Li, Y. Mi, F. Cai, Z. Yang, W. Zuo, X. Wang, and X. Fan. SceneDreamer360: Text-driven 3D-consistent scene generation with panoramic gaussian splatting. arXiv:2408.13711, 2024. 2

[37] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara. Toward a practical perceptual video quality metric, 2016. Netflix TechBlog. 4

[38] H. Liang, T. Wu, P. Hanji, F. Banterle, H. Gao, R. Mantiuk, and C. Öztireli. Perceptual quality assessment of NeRF and neural view synthesis methods for front-facing views. *Computer Graphics Forum*, 43(2):e15036, 2024. doi: 10.1111/cgf.15036 4

[39] K.-E. Lin, Z. Xu, B. Mildenhall, P. P. Srinivasan, Y. Hold-Geoffroy, S. DiVerdi, Q. Sun, K. Sunkavalli, and R. Ramamoorthi. Deep multi depth panoramas for view synthesis. In *ECCV*, 2020. 2

[40] Y. Liu, C. Lin, Z. Zeng, X. Long, L. Liu, T. Komura, and W. Wang. SyncDreamer: Generating multiview-consistent images from a single-view image. In *ICLR*, 2024. 2

[41] B. Luo, F. Xu, C. Richardt, and J.-H. Yong. Parallax360: Stereoscopic 360° scene representation for head-motion parallax. *TVCG*, 24(4):1545–1553, 2018. doi: 10.1109/TVCG.2018.2794071 2

[42] Y. Ma, D. Zhan, and Z. Jin. FastScene: Text-driven fast 3D indoor scene generation via panoramic gaussian splatting. In *International Joint Conferences on Artifical Intelligence (IJCAI)*, 2024. 2

[43] K. Matzen, M. F. Cohen, B. Evans, J. Kopf, and R. Szeliski. Low-cost 360 stereo photography and video capture. *ACM Trans. Graph.*, 36(4):148:1–12, 2017. doi: 10.1145/3072959.3073645 2

[44] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Trans. Graph.*, 38(4):29:1–14, 2019. doi: 10.1145/3306346.3322980 2

[45] 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. *Communications of the ACM*, 65(1):99–106, 2022. doi: 10.1145/3503250 2

[46] M. Mühlhausen, M. Kappel, M. Kassubeck, L. Wöhler, S. Grogorick, S. Castillo, M. Eisemann, and M. Magnor. Immersive free-viewpoint panorama rendering from omnidirectional stereo video. *Comput. Graph. Forum*, 42(6):e14796, 2023. doi: 10.1111/cgf.14796 2

[47] N. Müller, K. Schwarz, B. Rössle, L. Porzi, S. Rota Bulò, M. Nießner, and P. Kontschieder. MultiDiff: Consistent novel view synthesis from a single image. In *CVPR*, 2024. 2

[48] R. S. Overbeck, D. Erickson, D. Evangelakos, M. Pharr, and P. Debevec. A system for acquiring, compressing, and rendering panoramic light field stills for virtual reality. *ACM Trans. Graph.*, 37(6):197:1–15, 2018. doi: 10.1145/3272127.3275031 2

[49] A. Parra Pozo, M. Toksvig, T. Filiba Schrager, J. Hsu, U. Mathur, A. Sorkine-Hornung, R. Szeliski, and B. Cabral. An integrated 6DoF video camera and system design. *ACM Trans. Graph.*, 38(6):216:1–16, 2019. doi: 10.1145/3355089.3356555 1, 2

[50] S. Peleg, M. Ben-Ezra, and Y. Pritch. Omnistereo: Panoramic stereo imaging. *TPAMI*, 23(3):279–290, 2001. doi: 10.1109/34.910880 2

[51] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318, 2003. doi: 10.1145/882262.882269 3

[52] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *TPAMI*, 44(3):1623–1637, 2022. doi: 10.1109/TPAMI.2020.3019967 3

[53] M. Rey-Area, M. Yuan, and C. Richardt. 360MonoDepth: High-resolution 360° monocular depth estimation. In *CVPR*, 2022. 1, 2

[54] C. Richardt, Y. Pritch, H. Zimmer, and A. Sorkine-Hornung. Megastereo: Constructing high-resolution stereo panoramas. In *CVPR*, pp. 1256–1263, 2013. doi: 10.1109/CVPR.2013.166 2

[55] C. Richardt, J. Tompkin, and G. Wetzstein. Capture, reconstruction, and representation of the visual real world for virtual reality. In *Real VR – Immersive Digital Reality: How to Import the Real World into Head-Mounted Immersive Displays*, pp. 3–32. Springer, 2020. doi: 10.1007/978-3-030-41816-8_1 1, 2

[56] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2

[57] S. Rota Bulò, L. Porzi, and P. Kontschieder. Revising densification in Gaussian splatting. In *ECCV*, 2024. 4

[58] A. Serrano, I. Kim, Z. Chen, S. DiVerdi, D. Gutierrez, A. Hertzmann, and B. Masia. Motion parallax for 360° RGBD video. *TVCG*, 25(5):1817–1827, 2019. doi: 10.1109/TVCG.2019.2898757 2, 4

[59] M.-L. Shih, S.-Y. Su, J. Kopf, and J.-B. Huang. 3D photography using context-aware layered depth inpainting. In *CVPR*, 2020. doi: 10.1109/CVPR42600.2020.00805 1, 2

[60] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma, A. Clarkson, M. Yan, B. Budge, Y. Yan, X. Pan, J. Yon, Y. Zou, K. Leon, N. Carter, J. Briales, T. Gillingham, E. Mueggler, L. Pesqueira, M. Savva, D. Batra, H. M. Strasdat, R. D. Nardi, M. Goesele, S. Lovegrove, and R. Newcombe. The Replica dataset: A digital replica of indoor spaces. arXiv:1906.05797, 2019. 4, 5, 6, 7

[61] S. Szymanowicz, E. Insafutdinov, C. Zheng, D. Campbell, J. F. Henriques, C. Rupprecht, and A. Vedaldi. Flash3D: Feed-forward generalisable 3D scene reconstruction from a single image. arXiv:2406.04343, 2024. 2

[62] S. Szymanowicz, C. Rupprecht, and A. Vedaldi. Viewset diffusion: (0-)image-conditioned 3D generative models from 2D data. In *ICCV*, 2023. 2

[63] S. Szymanowicz, C. Rupprecht, and A. Vedaldi. Splatter image: Ultra-fast single-view 3D reconstruction. In *CVPR*, 2024. 2

[64] A. Tewari, T. Yin, G. Cazenavette, S. Rezchikov, J. B. Tenenbaum, F. Durand, W. T. Freeman, and V. Sitzmann. Diffusion with forward models: Solving stochastic inverse problems without direct supervision. In *NeurIPS*, 2023. 2

[65] R. Tucker and N. Snavely. Single-view view synthesis with multiplane images. In *CVPR*, 2020. doi: 10.1109/CVPR42600.2020.00063 2

[66] J. Waidhofer, R. Gadgil, A. Dickson, S. Zollmann, and J. Ventura. PanoSynthVR: Toward light-weight 360-degree view synthesis from a single panoramic input. In *ISMAR*, pp. 584–592, 2022. doi: 10.1109/ISMAR55827.2022.00075 1, 2

[67] G. Wang, P. Wang, Z. Chen, W. Wang, C. C. Loy, and Z. Liu. PERF: Panoramic neural radiance field from a single panorama. *TPAMI*, 2024. doi: 10.1109/TPAMI.2024.3387307 1, 2, 4, 5, 6, 7

[68] O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson. SynSin: End-to-end view synthesis from a single image. In *CVPR*, 2020. doi: 10.1109/CVPR42600.2020.00749 2

[69] J. Xu, J. Zheng, Y. Xu, R. Tang, and S. Gao. Layout-guided novel view synthesis from a single indoor panorama. In *CVPR*, 2021. 1

[70] L. Xu, V. Agrawal, W. Laney, T. Garcia, A. Bansal, C. Kim, S. Rota Bulò, L. Porzi, P. Kontschieder, A. Božič, D. Lin, M. Zollhöfer, and C. Richardt. VR-NeRF: High-fidelity virtualized walkable spaces. In *SIGGRAPH Asia*, 2023. doi: 10.1145/3610548.3618139 2

[71] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao. Depth anything V2. arXiv:2406.09414, 2024. 3

[72] A. Yu, V. Ye, M. Tancik, and A. Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In *CVPR*, 2021. 2

[73] J. Zhang, T. Zhu, A. Zhang, X. Yuan, Z. Wang, S. Beetschen, L. Xu, X. Lin, Q. Dai, and L. Fang. Multiscale-VR: Multiscale gigapixel 3D panoramic videography for virtual reality. In *ICCP*, 2020. doi: 10.1109/ICCP48838.2020.9105244 1

[74] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ACM Trans. Graph.*, 37(4):65:1–12, 2018. doi: 10.1145/3197517.3201323 2

[75] Z.-X. Zou, Z. Yu, Y.-C. Guo, Y. Li, D. Liang, Y.-P. Cao, and S.-H. Zhang. Triplane meets Gaussian splatting: Fast and generalizable single-view 3D reconstruction with transformers. In *CVPR*, 2024. 2