This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Seeing A 3D World in A Grain of Sand

Yufan Zhang[1] Yu Ji[2] Yu Guo[1] Jinwei Ye[1]
[1]George Mason University [2]LightThought LLC
https://miniature-3dgs.github.io/

Abstract

We present a snapshot imaging technique for recovering 3D surrounding views of miniature scenes. Due to their intricacy, miniature scenes with objects sized in millimeters are difficult to reconstruct, yet miniatures are common in life and their 3D digitalization is desirable. We design a catadioptric imaging system with a single camera and eight pairs of planar mirrors for snapshot 3D reconstruction from a dollhouse perspective. We place paired mirrors on nested pyramid surfaces for capturing surrounding multi-view images in a single shot. Our mirror design is customizable based on the size of the scene for optimized view coverage. We use the 3D Gaussian Splatting (3DGS) representation for scene reconstruction and novel view synthesis. We overcome the challenge posed by our sparse view input by integrating visual hull-derived depth constraint. Our method demonstrates state-of-the-art performance on a variety of synthetic and real miniature scenes.

1. Introduction

To see a World in a Grain of Sand
And a Heaven in a Wild Flower,
Hold Infinity in the palm of your hand
And Eternity in an hour.
- William Blake

Most existing works on 3D reconstruction or novel view synthesis focus on large-scale or life-size scenes. The tiny world of miniatures (*e.g.*, objects sized in centimeters or even millimeters) is somewhat neglected. 3D reconstruction of miniature scenes is challenging due to limitations on lenses, image resolution and reconstruction accuracy. To take images of miniatures, one would need a macro lens to magnify the tiny objects, such that they could cover substantial amount of pixels in an image. Due to high magnification ratio, macro images usually have shallow depth of field, making it hard to capture all-in-focus image. In addition, many miniatures have little textures (as limited by their size), which poses challenge to photogrammetry-based 3D reconstruction. Yet miniatures scenes are common in life (see examples in Fig. 1): from toys and decorations in household to artisanal crafts and antiques in design studios and museums, high-quality 3D reconstruction not only opens up new ways for viewing and interacting with those miniatures, and also benefits their preservation [4] .

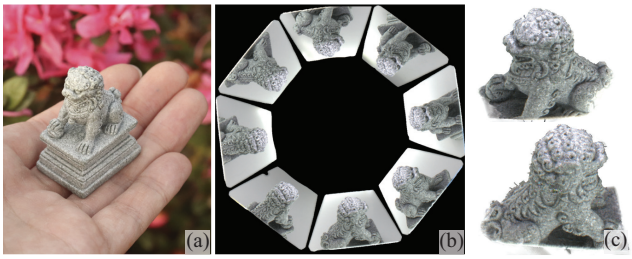

Figure 1. (a) A miniature scene with tiny object; (b) Example image captured by our catadioptric imaging system; (c) Novel view synthesis results.

In this work, we present a snapshot solution for reconstructing miniature scenes and synthesizing their images in full surround 360° views. We design a novel planar catadioptric lens to enable zoom-in all-focus imaging and time-synchronized full surround multi-view acquisition. Specifically, we use the reflection between paired mirrors to fold the light path in order to zoom-in onto small object in short distance. We arrange eight pairs of mirrors on the sides of two nested octagonal pyramids to capture surrounding views of the scene in a single shot. We analyze the ray geometry of our lens and derive closed-form formulas for optimizing the mirror configuration based on the size of the scene, such that the multi-view coverage of the scene is optimal. We map our multi-view sub-images to virtual cameras and pre-calibrate the virtual camera parameters. The calibration only needs to run once when the mirror configuration is set. Since our calibrated camera parameters are highly accurate, they greatly benefit scene reconstruction. In contrast, self-calibration methods (*e.g.*, structure from motion [27]) heavily rely on scene features and are not reliable for miniature scenes with little textures.

We use the 3D Gaussian Splatting (3DGS) [12] for scene

representation, and synthesize full surround novel views using our multi-view images and pre-calibrated camera parameters. Since our input views are sparse (*e.g.*, 8 views on a 360° circle), we use additional depth constraints derived from visual hull to improve the 3DGS reconstruction. Specifically, we first calculate visual hull using the object's silhouette masks extracted from multi-view images. We then project the visual hull to depth map under each viewpoint. Since visual hull is a convex volume that fully enclose the object, we propose a weighted depth loss that penalizes more on depth values greater than the visual hull depth (which indicates the point is outside of the visual hull). Our depth loss is especially effective for surfaces with little textures, and it can be used for generic 3DGS when visual hulls are available.

We validate our method on both synthetic and real data. For synthetic experiments, we render images by emulating our imaging setup and quantitatively evaluate the novel view synthesis results. For real experiments, we custom-build lens prototypes using 3D-printed housing and acrylic mirrors, and capture images of a variety of miniature scenes. We compare our method with recent sparse-view 3DGS approaches, and demonstrate better rendering quality. Our main contributions are summarized as follows:

- We design a planar catadioptric imaging system for recovering 3D surrounding views of miniature scenes in a single shot.
- We analyze the ray geometry of our catadioptric lens and derive closed-form formulas for calculating optimized lens parameters given scene information.
- We propose a novel weighted depth loss based on visual hull to improve 3DGS with sparse view input.
- We build a prototype for the proposed imaging system and validate our approach on real miniature scenes.

2. Related Work

Here we briefly review prior works that are most relevant to our imaging system and reconstruction algorithm.

Mirror-based imaging system. Mirrors, either planar or curved, are widely used for building imaging systems with extended field-of-view or snapshot multi-view capacities. Imaging systems that involve both mirrors and refractive lenses are called catadioptric systems. They have been extensively studied and used for stereo [23, 28, 35], panoramic [3, 29], surround-view [1, 17] and light field [6, 10, 22] imaging. Reshetouski and Ihrke [25] provide a comprehensive survey on the design and applications of various mirror-based imaging systems. Gluckman and Nayar [7, 8] show all possible configurations for catadioptric stereo and derive the reflection transformation for mirror-based image formation. Notably, kaleidoscopic imaging systems use the inter-reflection of mirrors to generate variations in viewpoints and illumination for 3D shape [26, 33] and reflectance [9, 11] reconstruction. Various techniques are explored for 3D reconstruction, including space carving [26], multi-view stereo [20], structured light [1] and neural surface representation [2]. Our work uses eight pairs of mirrors for full-surround imaging and the 3DGS framework for scene reconstruction. In contrast to kaleidoscopes, we avoid inter-reflection in our mirror system. Thus, our multi-view images are much easier to separate and analyze, although our angular resolution is sacrificed.

3DGS with sparse view input. 3D Gaussian Splatting (3DGS) [12] is an efficient radiance field representation that allows for high-quality real-time rendering of novel views, with relatively short training time. But the original 3DGS requires dense view supervision to achieve high quality rendering. With few reference views, the algorithm tends to overfit on inputs, resulting artifacts in unseen views. The problem gets even worse for surround-view rendering. Many recent techniques are proposed to allow robust 3DGS-based novel view synthesis when few views are available. A major trend is to use depth maps to provide additional supervision. FSGS [41] and SparseGS [37] use monocular depth predicted by pre-trained model [24]. GS2Mesh [34] and InstantSplat [5] adopt depth learned from multi-view stereo. ReconFusion [36] and GaussianObject [38] use diffusion model to synthesize addition views in order to provide dense viewpoint and depth supervision. RaDe-GS [39] improves the depth rasterization in 3DGS. In this work, we propose a new weighted depth loss that leverages depth map generated from visual hull to improve the quality of novel view synthesis with sparse surround-view input.

3. Imaging System

In this section, we introduce our imaging system for snapshot full-surround miniature scene reconstruction. We first describe the design of our planar catadioptric lens, and then analyze the ray geometry of the lens and derive optimized lens parameters given scene information.

3.1. Catadioptric Lens Design

Our imaging system consists of a camera and a planar catadioptric lens that allows for synchronized full-surround multi-view image acquisition. The conceptual design is illustrated in Fig. 2. Our catadioptric lens is made with eight pairs of flat mirrors that are circularly arranged on the sides of two nested octagonal pyramids. In each pair, the mirrors are facing each other with different tilting angles and shifted locations. Through two times of reflections, the mirror pair guides light from a scene that is underneath to the viewing camera on top. In this way, the light paths are folded

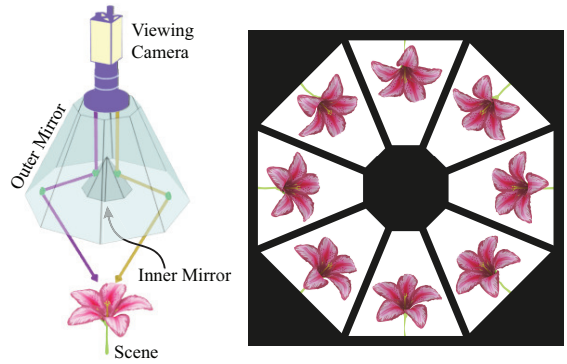

Figure 2. Left: A schematic illustration of our imaging system; Right: Sample image captured by the system.

such that we are able to image small objects in short distance, as if they are imaged by a zoom lens. Similar idea on light path folding is used in Folded Optics (or Origami Optics) [31, 32] for compact wide-angle lens design.

Then we arrange eight such mirror pairs along a circle, on surfaces of two nested octagonal pyramids, in order to capture full-surround multi-view images. A sample image captured by our imaging system is shown in Fig. 2. Each sub-image is formed by a pair of mirrors. With eight mirror pairs, we obtain eight sub-images that surround the scene in 360° and are time-synchronized by optics. Note that we purposely arrange the mirrors such that light only bounces once on a piece of mirror. Therefore, there is no inter-reflection in our sub-images. Similar design with mirror pairs is studied for stereo image acquisition [14]. But two mirror pairs on opposite sides have non-overlapping views and is not practical for stereo vision, whereas our circular arrangement guarantees overlapping in neighboring views and provides full-surround coverage of the scene.

Mapping to virtual cameras. By unfolding the reflection light paths, we are able to map our sub-images to virtual camera views (as if there are no mirrors). This unfolding process and mapped virtual cameras are illustrated in Fig. 3. The virtual cameras are evenly distributed on a circle, looking inward to the scene in the center. All cameras share the same intrinsic parameters, since mirror reflection linearly maps an image. The camera poses are determined by the mirror parameters (*e.g.*, tilting angles and positions). By varying the mirror configuration, we are able to adjust the viewing directions and baseline of surrounding views, so as to optimize their coverage of the scene. In Sec. 3.2, we analyze the multiview coverage in relationship to the mirror configuration in details.

Comparison with kaleidoscopic imaging. Kaleidoscopic imaging systems use the inter-reflection of mirror chambers for snapshot surrounding view acquisition [1, 11]. One key

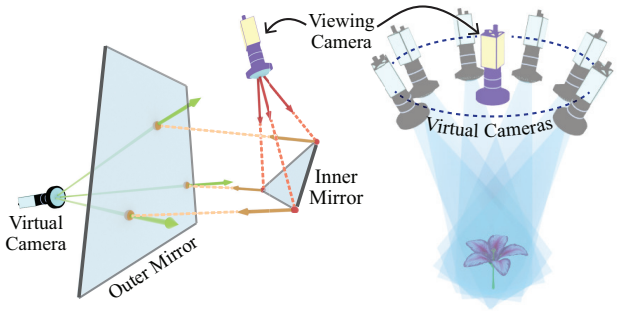

Figure 3. We map mirror reflections to virtual camera views by unfolding the light paths.

distinction between our design and kaleidoscopes is that we only allow one-bounce reflection on mirrors and so we do not have inter-reflection in images. With simpler ray geometry, our system is much easier to calibrate and we mitigate the challenging "labeling" problem on determining the mirror sequence involved in inter-reflections. However, the angular resolution is traded off in our design: we have fewer viewing directions comparing to kaleidoscope since the inter-reflected views are absent. Nevertheless, we show in Sec. 4 and through experiments that our sparse surrounding views are sufficient for 3DGS-based reconstruction and full surround novel view synthesis.

3.2. Ray Geometry Analysis

Next we analyze the light transport inside of our lens and derive the relationship between our viewing volume and mirror parameters. We then use the formulas to optimize the mirror configuration based on the scene dimensions, such that the multi-view coverage of the scene is optimal.

Effective viewing volume. The effective viewing volume of our imaging system is the intersection of viewing frustums of all virtual cameras. Here we derive the relationship between the dimension of effective viewing volume and the mirror parameters.

Since our sub-images exhibit weak perspective with long effective focal length, we assume orthographic camera model for simplicity. The effective viewing volume is carved out by eight beams of parallel light as a polyhedron that can be approximated as a symmetric double pyramid. We characterize this volume by its side angle at apex (θ) and the widest length at the base (l). WLOG, we illustrate the formation of effective viewing volume in a 2D cross-section (see Fig. 4), since our mirror design is symmetric. We parameterize the mirrors in a pair using their tilting angles ($\alpha_{1,2}$), vertically projected heights ($h_{1,2}$), and distances ($d_{1,2}$) between their upper edges and the central ray (*i.e.*, vertical ray from the imager center). We denote the inner mirror as M_1 and the outer one as M_2. Subscript of the parameters indicate their correspondence to the mirrors.

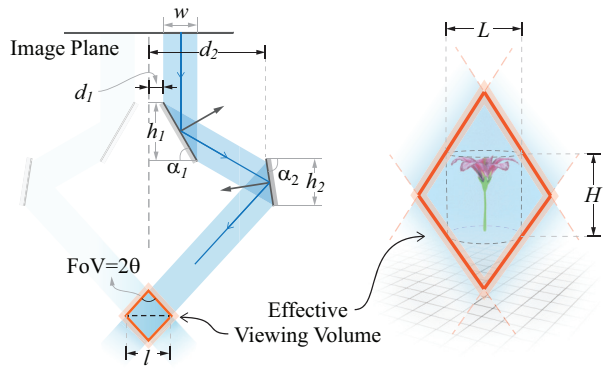

Figure 4. The reflection light path of our mirror pair and the optimal coverage of a scene by our effective viewing volume.

Since light path is reversible, we trace rays from the camera to the scene. Given a vertical ray from camera, it is first bounced on M_1 and then on M_2 before reaching the scene. The ray's incident angle to M_1 (*i.e.*, the angle between the incident ray and the mirror's normal) is: $\omega_1 = \alpha_1$. It's incident angle to M_2 can then be calculated as: $\omega_2 = 2\alpha_1 - \alpha_2$. Since rays reflected from M_2 intersect with the symmetric beam from the opposite side to form the effective viewing volume, the pyramid's side angle at base can be calculated as: $\beta = 90° - \alpha_2 + \omega_2 = 90° - 2(\alpha_2 - \alpha_1)$. The side angle at apex is thus:

$$\theta = 90° - \beta = 2(\alpha_2 - \alpha_1). \tag{1}$$

Let $\Delta\alpha = \alpha_2 - \alpha_1$. We define $2\theta = 4\Delta\alpha$ as the field-of-view (FoV) of our catadioptric lens. Detailed derivation of these angles can be found in the supplementary material.

The width of parallel light beam that can enter the camera is determined by M_1's size and angle: $w = h_1/\tan\alpha_1$. Since w remains constant after reflection, we can calculate the base length l of the viewing volume as:

$$l = \frac{w}{\cos\theta} = \frac{h_1}{\tan\alpha_1 \cdot \cos 2\Delta\alpha}. \tag{2}$$

Discussions on mirror design. From Eqs. 1 and 2, we can see that the dimension of effective viewing volume is determined by the angle and size of M_1 (h_1 and α_1) and the angle difference between M_1 and M_2 ($\Delta\alpha$). The greater the $\Delta\alpha$, the larger the FoV. In order to form reflection images from underneath without inter-reflection, the mirror parameters need to satisfy the following three conditions:

(i) $45° < \alpha_1 < \alpha_2 < 90°$, such that light from scene underneath can be reflected to the imager on top;

(ii) $h_2 \geq \dfrac{\sin\alpha_2}{\tan\alpha_1 \cdot \cos(\alpha_2 - 2\alpha_1)} \cdot h_1$, such that M_2 can cover the entire light beam reflected from M_1;

(iii) $d_2 \geq \dfrac{\tan\alpha_1 + \cot 2\Delta\alpha}{\tan\alpha_1 \cdot (\cot 2\Delta\alpha - \cot 2\alpha_1)} \cdot h_1 + d_1$, such that no inter-reflection occurs between M_1 and M_2.

Please see the supplementary material for derivation of these conditions. Note that although the size and location of M_2 (h_2 and d_2) would not affect the dimension of effective viewing volume dimension, changing these parameters would result in the viewing volume to be shifted vertically. It is preferable for d_2 to take smaller values, since the resulting viewing volume would be closer to the camera.

Optimized mirror configuration. We then show how to find optimal configuration for the mirror pair, given the size of the scene. Since mirror parameters determine the dimension of effective viewing volume. We first find a viewing volume that provides the optimal scene coverage, and then use Eqs. 1 and 2, along with the three conditions to calculate mirror parameters. We use the following two criteria to find the optimal viewing volume: 1) in order to have a complete reconstruction of the scene, the viewing volume should be large enough to fully enclose the scene; 2) larger FoV is preferred in order to have more coverage on the lateral sides. Since FoV is intersected by the viewing directions of virtual cameras, the larger the FoV, the more oblique the viewing angles are, and thus the more coverage on the side.

Since the amount of light that can be received by the imager is bounded by its size, we first allow the width of the parallel beam w to take its maximum value, which is half of the sensor size[1], and we denote it as w_{max}. Since $w = h_1/\tan\alpha_1$, we can adjust the angle and length of M_1 to achieve w_{max}. Given w_{max}, the dimension of effective viewing volume is only related to $\Delta\alpha$. The greater the $\Delta\alpha$, the larger the FoV and base length l, but the smaller the vertical height $h = w_{\text{max}}/\sin 2\Delta\alpha$. We then set out to find the viewing volume with the largest FoV that can fully enclose a scene with known size. We approximate a scene using its bounding box with size $W \times L \times H$ (where H is the vertical height and we assume $L \geq W$). The FoV takes its largest value when the scene's bounding box is inscribed in the viewing volume (see Fig. 4). This is because when FoV further increases beyond this value, the vertical height h decreases, which would result in the scene being partially outside of the viewing volume.

Next we derive the FoV value under this circumstance. With similar triangles, we have $(h - H)/h = L/l$. By substituting $h = w_{\text{max}}/\sin 2\Delta\alpha$ and $l = w_{\text{max}}/\cos 2\Delta\alpha$, we have $H \cdot \sin 2\Delta\alpha + L \cdot \cos 2\Delta\alpha = w_{\text{max}}$. By applying the sine angle addition identity, we have:

$$\Delta\alpha = \frac{1}{2}\left(\arcsin\left(\frac{w_{\text{max}}}{\sqrt{L^2 + H^2}}\right) - \arctan\left(\frac{L}{H}\right)\right). \tag{3}$$

The largest FoV is $4\Delta\alpha$, which can be calculated given the scene size. Eq. 3 indicates the angle difference between

[1]We assume orthographic model. Under perspective model, this value also depends on the lens.

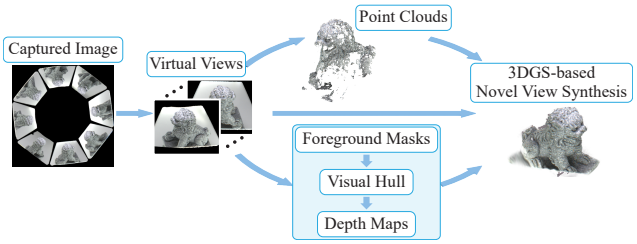

Figure 5. Our overall algorithmic pipeline.

M_1 and M_2, which is the optimal mirror configuration to achieve the largest FoV.

4. Scene Reconstruction

In this section, we show how to use our capture image for scene reconstruction and novel view synthesis. Specifically, we adopt the 3D Gaussian Splatting (3DGS) framework. We use depth supervision constrained by visual hull to improve the quality of full-surround novel view synthesis with sparse reference views. The overall algorithmic pipeline of our method is shown in Fig. 5.

4.1. Pre-processing Steps

Since our multi-view images are combined into a single image, we first segment our captured image into individual multi-view images. We obtain the segmentation mask by taking one image of a solid color background. By applying the mask, we obtain eight images as if captured by surrounding virtual cameras. Since the orientation of virtual cameras are flipped on opposite sides, we re-project the multi-view images onto new viewing planes, such that the transition of view poses is smooth on a circle. We also create foreground masks to segment out the object of interest. This can be done using off-the-shelf segmentation tools, such as Segment Anything [15]. Please see examples of our input and processed images in the supplementary material.

We also pre-calibrate our imaging system using a small checkerboard target to obtain the intrinsic and extrinsic parameters of virtual cameras. The extrinsic parameters are re-calculated after view re-projection. The calibration process only needs to run once when our catadioptric lens is fixed on the camera. Pre-calibrating these parameters greatly benefit miniature scene reconstruction, since these tiny scenes often lack textures, and this would result in self-calibration methods (*e.g.*, COLMAP [27]) to fail. With the camera parameters, we estimate an initial point cloud using structure from motion. We then use the eight multi-view images, along with the camera parameters and the initial point cloud as input for 3DGS-based scene reconstruction.

4.2. 3DGS Representation

3DGS, first introduced by Kerbl *et al.* [12], represents a scene using a set of 3D Gaussian elements. It is a more compact and efficient 3D representation, comparing to NeRF [21] and classical triangle meshes. It has been demonstrated great success in many rendering applications [16, 30, 40].

A 3D Gaussian is defined by its center position $\boldsymbol{\mu} \in \mathbb{R}^3$ and a covariance matrix $\Sigma \in \mathbb{R}^{3\times3}$:

$$G(\boldsymbol{x}) = e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}, \tag{4}$$

where $\boldsymbol{x}$ is a point on the Gaussian, and Σ can be further decomposed into rotation and scaling matrices (R and S respectively): $\Sigma = RSS^\top R^\top$.

To render an image, the 3D Gaussians are projected by: $\Sigma' = JW\Sigma W^\top J^\top$, where W is the viewing transformation matrix and J is the Jacobian of the affine approximation of the projective transformation. A pixel's color $\mathbf{C}$ is calculated by blending the color of ordered Gaussians that overlap the pixel, similar to the NeRF-style rendering:

$$\mathbf{C} = \sum_{i=1}^{n} c_i \alpha_i T_i, \tag{5}$$

where c_i is the color of the Gaussian, α_i is the opacity of the 2D projected Gaussian and $T_i = \prod_{j=1}^{i-1}(1-\alpha_j)$ is the transmittance along the ray.

4.3. Optimization

To optimize the Gaussian parameters, we compare the rendered output against our input reference views using the following loss function:

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_2 \mathcal{L}_{\text{D-SSIM}} + \lambda_3 \mathcal{L}_{\text{depth}}, \tag{6}$$

where $\lambda_{1,2,3}$ are weighting factors for balancing the terms. We use $\lambda_1 = 0.8$, $\lambda_1 = 0.2$ and $\lambda_3 = 0.5$ in our experiments. $\mathcal{L}_1$ and $\mathcal{L}_{\text{D-SSIM}}$ are standard losses that evaluate the color similarity between the rendered image and the reference image. Since our input views are sparse with limited overlapping, using the color losses alone results in severe artifacts when synthesizing unseen views. We therefore add the third loss $\mathcal{L}_{\text{depth}}$ (which will be described in the following paragraph), a depth loss constrained by visual hull-generated depth maps, to improve the rendering quality of novel views.

Visual hull-constrained depth loss. By using the camera parameters and foreground object masks (which indicate their sillouettes), we can carve out a visual hull [18] for objects in the scene. We then project the visual hull under each reference view and render depth map $\mathbf{D}_{\text{VH}}$ for depth regularization.

We use the rasterizer provided by [13] to render depth map $\mathbf{D}_{\text{render}}$ from 3D Gaussians. Specifically, the depth is rendered by replacing the color c_i in Eq. 5 with the depth d_i of the Gaussian's center: $\mathbf{D}_{\text{render}} = \sum_{i=1}^{n} d_i \alpha_i T_i$. Since $\mathbf{D}_{\text{render}}$ and $\mathbf{D}_{\text{VH}}$ have consistent scale, we calculate the depth loss $\mathcal{L}_{\text{depth}}$ as a weighted L1 norm:

$$\mathcal{L}_{\text{depth}} = \frac{2}{1 + e^{\Delta d_i}} |\mathbf{D}_{\text{render}} - \mathbf{D}_{\text{VH}}|, \tag{7}$$

where $\Delta d_i = \mathbf{D}_{\text{render}}(\boldsymbol{p}_i) - \mathbf{D}_{\text{VH}}(\boldsymbol{p}_i)$, with $\boldsymbol{p}_i$ refers to a pixel in image.

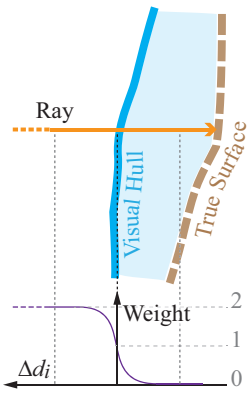

Figure 6. S-shape weight.

$\mathcal{L}_{\text{depth}}$ is weighted by a s-shaped logistic function with values greater than 1 when $\Delta d_i > 0$ (see Fig. 6). We introduce this weight based on the observation that visual hull is a convex enclosure of the actual geometry. Therefore, we use larger weight for points that are outside of the visual hull (*i.e.*, $\Delta d_i > 0$). For points inside of the visual hull, we gradually decrease the weight to zero, since its depth may still be correct even though $\Delta d_i \neq 0$, if the surface is concave. This weight allows us to model the visual hull-constrained depth regularization more precisely.

5. Experiments

We evaluate our method on both synthetic and real-captured data. Our codes and input data will be made available on our project website.

5.1. Experimental Setup

Synthetic data generation. We simulate synthetic images as captured by our proposed catadioptric imaging system using Autodesk 3ds Max. We build 3D models of mirror pairs and render image using a viewing on top of the mirror pyramid. We render 7 scenes with various levels of complexity in terms of texture and geometry. We render ground truth masks for segmenting the multi-view images and foreground objects. In order to calibrate the virtual cameras, we render checkerboard images with different poses. We use ground truth intrinsic parameters and only calibration the extrinsic parameters. We also render ground truth novel view images for quantitative evaluation. Since our 8 virtual cameras are arranged on a circle (which are our input reference views), we render 24 views along the same circle and use them as ground truth for evaluating novel view synthesis results.

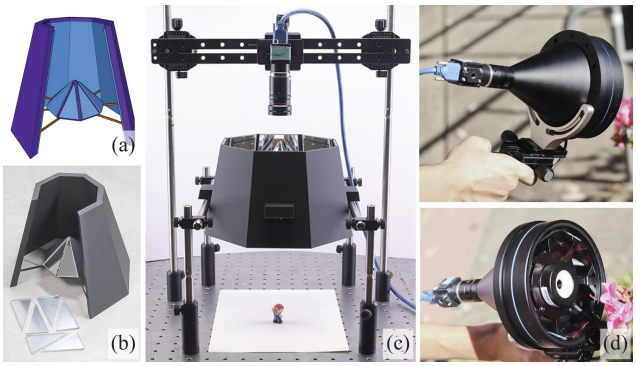

Figure 7. Prototypes of our mirror lens. (a) 3D model of our lens housing; (b) 3D printed housing with acrylic mirror pairs adhered to the inside; (c) Our imaging setup; (d) Our custom-made portable prototype.

Real experiment setup. We build prototypes for our proposed mirror lens in order to perform real experiments. In order to validate our derivation on mirror angle optimization, we custom-build lens with different mirror angles using 3D-printed housing and acrylic mirrors. The lens 3D model and prototype components are shown in Fig. 7 (a) and (b). We mount a 5 megapixel FLIR camera with 16mm lens on top of the lens to capture images of a scene that is place below the mirror lens. Since our lens block light from above the scene, we mount LED light underneath the inner pyramid in order for the scene to be well lit. We try lenses with different mirror angles and spacing. Effects of these mirror configuration changes with respect to scene coverage are consistent with our derivations in Sec. 3.2. Please see the supplementary material for example images captured with different mirror configurations.

We also order a customized mirror lens with sharping imaging quality from a lens maker (see Fig. 7 (d)). We use this lens for capturing real scene data. We use the same 5 megapixel FLIR camera with 12mm lens for image acquisition. Since the lens is compact and portable, we capture miniature scenes from both indoor and outdoor. In total, we capture 10 indoor scenes and 2 outdoor scenes. The size of miniature objects in our scene is in the range of 1cm to 5cm.

We pre-calibrate our imaging system to obtain intrinsic and extrinsic parameters for the virtual cameras. Since mirror reflection doesn't change intrinsic parameters, we first estimate the intrinsics with all available views and use the same set of parameters for all virtual cameras. We then calibrate the extrinsics after mounting the mirror lens. We also run bundle adjustment to further refine the camera poses. The final re-projection error of our calibration is 0.77.

Code implementation. We implement the pre-processing steps, including image re-projection, visual hull generation and visual hull depth rendering *etc.*, using MATLAB. With our multi-view images and calibrated camera parame-

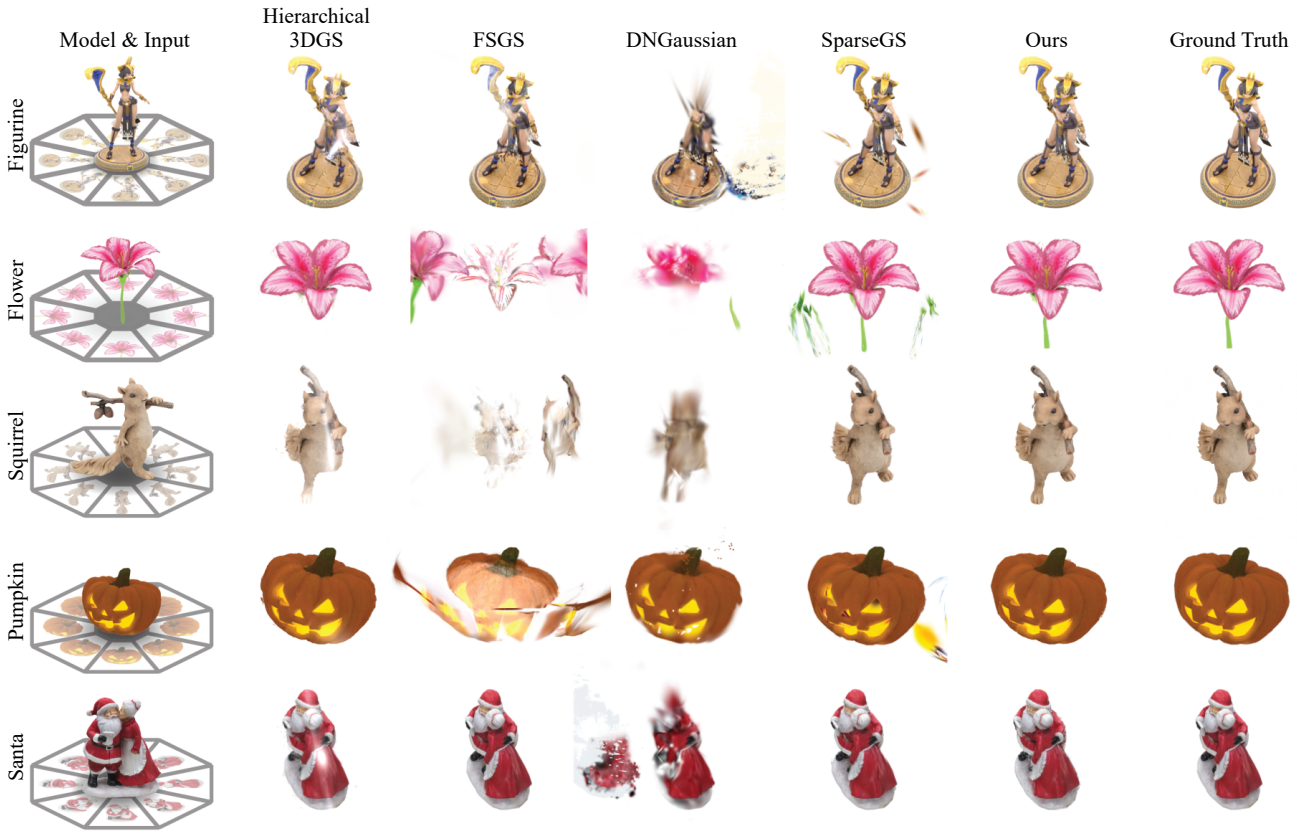

Figure 8. Visual comparison results on synthetic data. See supplementary material for more results and videos of full-surround rendering.

Table 1. Quantitative comparison results.

Method	SSIM ↑	PSNR ↑	LPIPS ↓
Hierarchical 3DGS	0.9750	26.8259	0.0298
FSGS	0.7844	18.9281	0.1100
DNGaussian	0.9128	21.3979	0.1296
SparseGS	0.9756	31.8415	0.0367
Ours	**0.9783**	**32.4792**	**0.0265**

ters, we use Agisoft Metashape[2] to generate an initial point cloud. We then convert the point cloud and camera parameters to COLMAP format, so they can be used by 3DGS. We use the latest Hierarchical 3DGS [13] as our backbone code. We integrate our visual hull-based depth regularization and optimize the rendering using foreground object mask, such that only the foreground is used for optimization. All our code are run on a computer with NVIDIA 4090 GPU. The average running for reconstructing a scene using 8 reference views, each with resolution 800×800, is around 2 minutes.

5.2. Synthetic Results

We validate our approach on our simulated data, and compare our results with recent state-of-the-art 3DGS algorithms: Hierarchical 3DGS [13], FSGS [41], DNGaussian [19] and SparseGS [37]. Most of these methods are optimized for taking sparse view input. It is worth noting that since our scenes are small and lack distinct features, COLMAP fails to run on all of our scenes. For fair comparison, we use our pre-calibrated camera parameters and initial point cloud as input for all compared algorithms. For each scene, we use 8 reference views as input and render 24 novel views. Visual comparison results on five scenes are shown in Fig. 8. Quantitative comparison results using standard metrics (*e.g.*, SSIM, PSNR and LPIPS) are shown in Table 1. The metric values are averaged over all scenes and all rendered novel views.

We can see that our method outperforms the state-of-the-arts in both qualitative and quantitative comparisons. The other sparse view methods do not perform very well. It is likely because that the monocular depth map they use is less accurate than our visual hull depth (see supplementary material for an ablation study on depth loss).

5.3. Real Results

We perform comparisons on real scene images taken with our mirror lens. Fig. 9 show visual comparison results against the state-of-the-arts. We also show our captured

[2]https://github.com/agisoft-llc

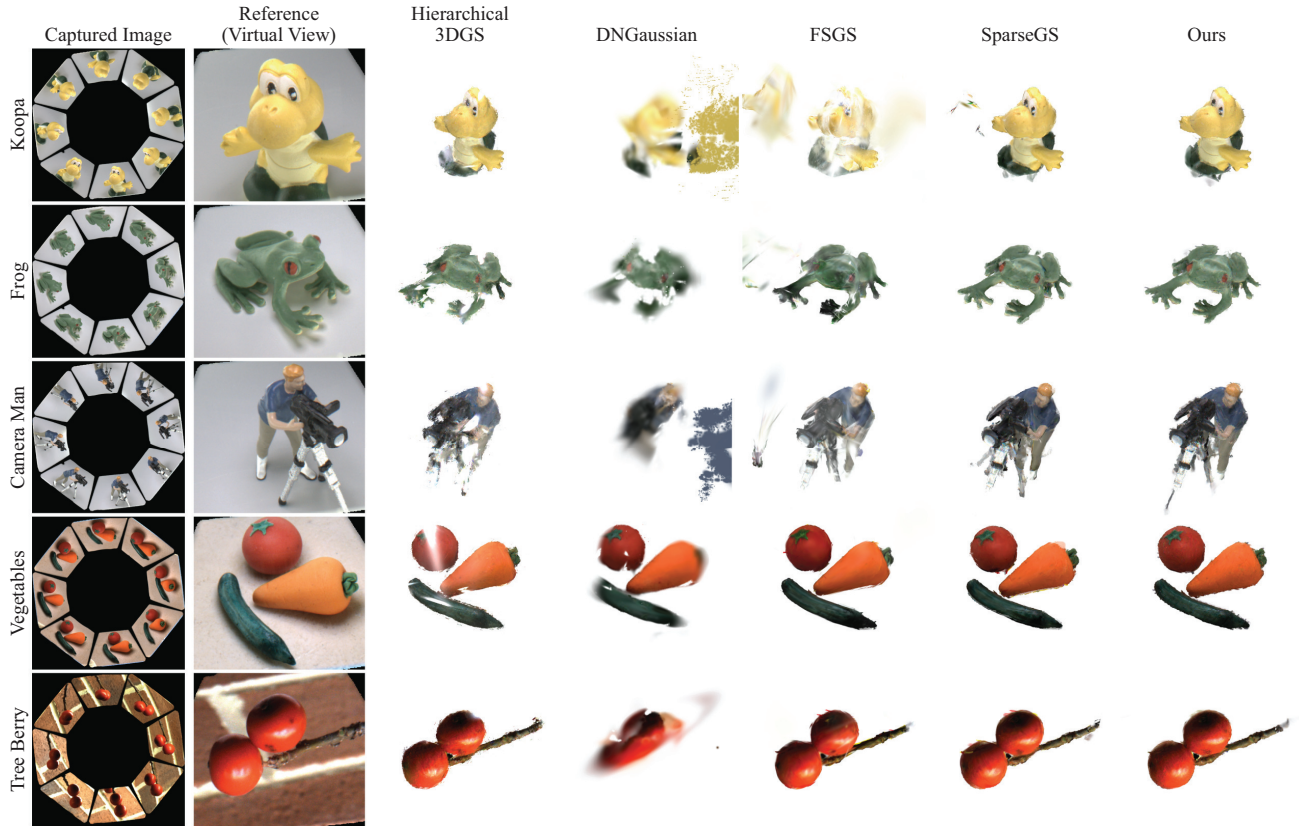

Figure 9. Visual comparison results on real data. See supplementary material for more results and videos of full-surround rendering.

image (processed with the multi-view mask) and one re-projected virtual view, which is used as input reference image for 3DGS (we use eight reference views in total). All the scenes are composed with miniature objects with size between 1cm and 4cm. The "tree berry" scene is captured outdoor with our portable lens. Same as the synthetic experiments, all compared methods take our pre-calibrated camera parameters and initial point cloud, since COLMAP fails on our scenes. The "camera man" scene is challenging with thin structures around 1mm (*e.g.*, the tripod legs). Our method is able to recover the thin legs, whereas all other methods cannot.

Since camera parameters are critical, we also compare our method against a recent COLMAP-free method, InstantSplat [5]. The method bypasses COLMAP by jointly optimizing Gaussian attributes and camera parameters. Visual comparison of a synthesized novel view is shown in Fig. 10. We also show their estimated camera poses and our calibrated poses. Their rendering quality is poor likely because of inaccurate camera pose estimation.

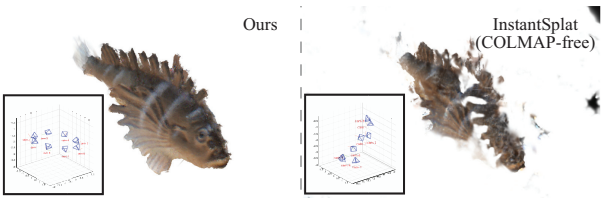

Figure 10. Comparison with a COLMAP-free method [5].

6. Conclusion

We have presented an imaging solution using circularly arranged mirror pairs for full-surround reconstruction and novel view synthesis of miniature scenes. We have thoroughly analyzed the design factors of our mirror lens and derived optimal parameters given the scene size. We have tailored the latest 3DGS framework to allow accurate and robust scene reconstruction using our sparse view input. Our method has been validated through synthetic and real experiments and demonstrated state-of-the-art performance. Since our approach is a single-shot solution, it can be applied to dynamic scenes. One viable direction is to incorporate temporal consistency to allow smooth reconstruction of dynamic scenes.

Acknowledgments

This project is partially supported by NSF awards 2225948 and 2238141.

References

[1] Byeongjoo Ahn, Ioannis Gkioulekas, and Aswin C Sankaranarayanan. Kaleidoscopic structured light. *ACM TOG*, 40 (6):1–15, 2021. 2, 3

[2] Byeongjoo Ahn, Michael De Zeeuw, Ioannis Gkioulekas, and Aswin C. Sankaranarayanan. Neural kaleidoscopic space sculpting. In *CVPR*, 2023. 2

[3] Javaan Singh Chahl and Mandyam Veerambudi Srinivasan. Reflective surfaces for panoramic imaging. *Applied optics*, 36(31):8275–8285, 1997. 2

[4] Jack Davy and Charlotte Dixon. *Worlds in Miniature*. UCL Press, 2019. 1

[5] Zhiwen Fan, Wenyan Cong, Kairun Wen, Kevin Wang, Jian Zhang, Xinghao Ding, Danfei Xu, Boris Ivanovic, Marco Pavone, Georgios Pavlakos, et al. Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds. *arXiv preprint arXiv:2403.20309*, 2, 2024. 2, 8

[6] Martin Fuchs, Markus Kächele, and Szymon Rusinkiewicz. Design and fabrication of faceted mirror arrays for light field capture. *Comput. Graph. Forum*, 32, 2013. 2

[7] Joshua Gluckman and Shree Nayar. Rectified catadioptric stereo sensors. In *CVPR*, 2000. 2

[8] Joshua Gluckman and Shree Nayar. Catadioptric stereo using planar mirrors. *IJCV*, 44:65–79, 2001. 2

[9] Jefferson Y. Han and Ken Perlin. Measuring bidirectional texture reflectance with a kaleidoscope. *ACM TOG*, 22(3), 2003. 2

[10] Ivo Ihrke, Timo Stich, Heiko Gottschlich, Marcus Magnor, and Hans-Peter Seidel. Fast incident light field acquisition and rendering. *Journal of WSCG*, 16:25–32, 2008. 2

[11] Ivo Ihrke, Ilya Reshetouski, Alkhazur Manakov, Art Tevs, Michael Wand, and Hans-Peter Seidel. A kaleidoscopic approach to surround geometry and reflectance acquisition. In *CVPRW*, 2012. 2, 3

[12] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM TOG*, 42(4), 2023. 1, 2, 5

[13] Bernhard Kerbl, Andreas Meuleman, Georgios Kopanas, Michael Wimmer, Alexandre Lanvin, and George Drettakis. A hierarchical 3d gaussian representation for real-time rendering of very large datasets. *ACM TOG*, 43(4), 2024. 6, 7

[14] Jungho Kim, Kuk-jin Yoon, Jun-sik Kim, and Inso Kweon. Visual slam by single-camera catadioptric stereo. In *SICE-ICASE International Joint Conference*, 2006. 3

[15] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 5

[16] Muhammed Kocabas, Jen-Hao Rick Chang, James Gabriel, Oncel Tuzel, and Anurag Ranjan. Hugs: Human gaussian splats. In *CVPR*, pages 505–515, 2024. 5

[17] Douglas Lanman, Daniel Crispell, and Gabriel Taubin. Surround structured lighting for full object scanning. In *International Conference on 3-D Digital Imaging and Modeling (3DIM)*, 2007. 2

[18] Aldo Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE TPAMI*, 16(2):150–162, 1994. 5

[19] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *CVPR*, 2024. 7

[20] Adrien Mas, Guillaume Druart, Maxime Vaché, Sylvain Favier, Aymeric Alazarine, Eric Compain, Nathalie Morin, and Nicolas Guérineau. Kaleidoscope-based multi-view infrared system. *Optics Letters*, 44(20):4977–4980, 2019. 2

[21] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 5

[22] Yasuhiro Mukaigawa, Seiichi Tagawa, Jaewon Kim, Ramesh Raskar, Yasuyuki Matsushita, and Yasushi Yagi. Hemispherical confocal imaging using turtleback reflector. In *ACCV*, 2011. 2

[23] Sameer A. Nene and Shree K. Nayar. Stereo with mirrors. *ICCV*, pages 1087–1094, 1998. 2

[24] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE TPAMI*, 44(3), 2022. 2

[25] Ilya Reshetouski and Ivo Ihrke. *Mirrors in computer graphics, computer vision and time-of-flight imaging*. 2013. 2

[26] Ilya Reshetouski, Alkhazur Manakov, Hans-Peter Seidel, and Ivo Ihrke. Three-dimensional kaleidoscopic imaging. In *CVPR*, 2011. 2

[27] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1, 5

[28] Tomáš Svoboda and Tomáš Pajdla. Epipolar geometry for central catadioptric cameras. *IJCV*, 49:23–37, 2002. 2

[29] Rahul Swaminathan, Shree K Nayar, and Michael D Grossberg. Framework for designing catadioptric projection and imaging systems. *ICCV*, 2003. 2

[30] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *ICLR*, 2024. 5

[31] Eric J. Tremblay, Ronald A. Stack, Rick L. Morrison, and Joseph E. Ford. Ultrathin cameras using annular folded optics. *Appl. Opt.*, 46(4):463–471, 2007. 3

[32] Eric J. Tremblay, Ronald A. Stack, Rick L. Morrison, Jason Harris Karp, and Joseph E. Ford. Ultrathin four-reflection imager. *Applied optics*, 48 2:343–54, 2009. 3

[33] Felix Wechsler, Rainer Heintzmann, and Ivo Ihrke. Kaleidomicroscope-a kaleidoscopic multiview microscope. In *Computational Optical Sensing and Imaging*, pages CTu4F–5, 2022. 2

[34] Yaniv Wolf, Amit Bracha, and Ron Kimmel. Gs2mesh: Surface reconstruction from gaussian splatting via novel stereo views. In *ECCV*, 2024. 2

[35] Hsien-Huang P. Wu and Shih-Hsin Chang. Design of stereoscopic viewing system based on a compact mirror and dual monitor. *Optical Engineering*, 49:027401, 2010. 2

[36] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor

Verbin, Jonathan T Barron, Ben Poole, and Aleksander Holynski. Reconfusion: 3d reconstruction with diffusion priors. In *CVPR*, 2024. 2

[37] Haolin Xiong, Sairisheek Muttukuru, Rishi Upadhyay, Pradyumna Chari, and Achuta Kadambi. SparseGS: Real-time 360° sparse view synthesis using gaussian splatting. *Arxiv*, 2023. 2, 7

[38] Chen Yang, Sikuang Li, Jiemin Fang, Ruofan Liang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Gaussianobject: High-quality 3d object reconstruction from four views with gaussian splatting. *ACM TOG*, 43(6), 2024. 2

[39] Baowen Zhang, Chuan Fang, Rakesh Shrestha, Yixun Liang, Xiaoxiao Long, and Ping Tan. Rade-gs: Rasterizing depth in gaussian splatting. *arXiv preprint arXiv:2406.01467*, 2024. 2

[40] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. In *CVPR*, pages 21634–21643, 2024. 5

[41] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. FSGS: Real-time few-shot view synthesis using gaussian splatting. In *ECCV*, 2024. 2, 7