UniSHARP: Universal Sharp Monocular View Synthesis

Meixi Song[1] **Dizhe Zhang**[1]* **Hao Ren**[1,2] **Ruiyang Zhang**[1,3]

Bo Du[4] **Ming-Hsuan Yang**[5] **Lu Qi**[1,4]*

[1] Insta360 Research [2] Sun Yat-sen University [3] Beihang University

[4] Wuhan University [5] University of California, Merced

Figure 1: UniSHARP performs monocular novel view synthesis across diverse camera types. Given a single image from a perspective, wide-FoV, fisheye, or panoramic camera, UniSHARP predicts a 3D Gaussian point cloud and renders high-quality novel views.

Abstract

In this work, we focus on extending SHARP, the popular photorealistic view synthesis method, for universal monocular rendering across a continuum of camera systems, from conventional perspective cameras to wide-field-of-view, fisheye and omnidirectional panoramic settings. To overcome the pinhole-specific assumptions of SHARP, our key idea is to align various images in a unified omnidirectional latent space. Thus, we propose UniSHARP, which performs implicit alignment in both feature and Gaussian spaces. Specifically, Gaussian primitives are arranged along rays and radial distances in a ray-based universal representation, while 2D semantic and 3D spatial features extracted from UniK3D-inspired encoders are jointly decoded to generate the complete Gaussian cloud. To comprehensively evaluate our method, we construct a benchmark covering diverse imaging systems across various scenes. The benchmark is further stratified by field of view (FoV) to enable fine-grained assessment of the universal monocular rendering task. Extensive experiments on the proposed benchmark demonstrate the effectiveness of UniSHARP, outperforming alternative methods by a large margin. The project page can be found at: https://insta360-research-team.github.io/Unisharp-website/

*indicates corresponding author.

Preprint.

1 Introduction

Novel view synthesis is a fundamental capability for spatial visual intelligence, enabling captured images to support robotic navigation, AR/VR interaction, immersive telepresence, and 3D content creation [1–9]. Albeit the success achieved by neural radiance fields (NeRF) [10] and 3D Gaussian Splatting (3DGS) [11], the monocular view synthesis remains ill-posed due to the severe spatial information loss inherent in a single image.

Recent monocular 3DGS methods, such as SHARP [12] and Flash3D [13], learn feedforward Gaussian priors from perspective image collections and regress renderable primitives from a single input. However, trained primarily on narrow-FoV perspective images, these regressors fail to generalize to diverse imaging systems [14–18], including wide-FoV, fisheye, and panoramic cameras. These practical constraints motivate universal monocular novel view synthesis, where a model must infer 3D structure, visibility, and appearance from a single image while remaining applicable to heterogeneous camera models.

To address this issue, an intuitive solution is to fine-tune SHARP on wide-FoV or panoramic images. However, since SHARP maps every pixel in normalized space under the pinhole camera assumption, it inherently fails to predict geometry in non-pinhole domain. Another approach is to re-project images into multiple views, but this introduces additional computational overhead and requires extra processing to handle stitching artifacts. Thus, one question raised: how can widely used methods such as SHARP be adapted to diverse camera systems in a simple manner?

Inspired by panoramic vision [19–25], we propose UniSHARP, which extends the popular SHARP framework for universal monocular view synthesis via a unified omnidirectional representation across diverse camera systems. Specifically, UniSHARP performs implicit alignment in the latent rather than the image space along two dimensions. On one hand, a ray-based universal representation organizes Gaussian primitives along rays and radial distances, enabling initialization and refinement in a shared 3D space rather than projection-specific image coordinates. On the other hand, a unified feature space decodes both 2D semantic embeddings and 3D spatial features, providing complementary appearance and geometry priors for single-image reconstruction. With such minor modifications, the robustness of UniSHARP to diverse camera systems is enhanced.

For scenarios where camera parameters are unavailable, UniSHARP also supports pose-free monocular inference from a single RGB image. It uses the predicted ray field to infer the input camera type and recover the rendering geometry, allowing the same feedforward Gaussian predictor to operate without manually provided intrinsics.

To well evaluate our performance, we build a comprehensive benchmark that collects narrow perspective, wide-FoV, fisheye, and panoramic validation data across real-world and simulated scenes. The benchmark combines established validation datasets with OmniRooms, our AirSim-based indoor panoramic dataset, and its projected wide-FoV variant. This FoV-stratified design enables controlled analysis of how rendering quality changes as the camera FoV increases from 60° to 360°.

Our contributions are summarized as follows:

- We propose UniSHARP, a universal-camera feedforward 3DGS framework for monocular novel view synthesis across standard perspective, wide-FoV, fisheye, and panoramic inputs. It reformulates SHARP-style Gaussian prediction in ray-distance space and can operate with predicted ray fields when calibrated camera parameters are unavailable.
- We develop a feature-space Gaussian prediction pipeline that fuses 2D semantic encodings with 3D spatial features and allocates Gaussians at native input resolution, preserving geometric priors and high-frequency image details without camera-specific resizing.
- We introduce panoramic-specific adaptations, including spherical Gaussian initialization and distortion-aware probabilistic dropout, to regularize Gaussian distributions under severe equirectangular projection distortion.
- We introduce a FoV-stratified benchmark spanning perspective, wide-FoV, fisheye, and panoramic cameras. We validate UniSHARP on this benchmark, demonstrating state-of-the-art rendering quality and strong cross-camera generalization.

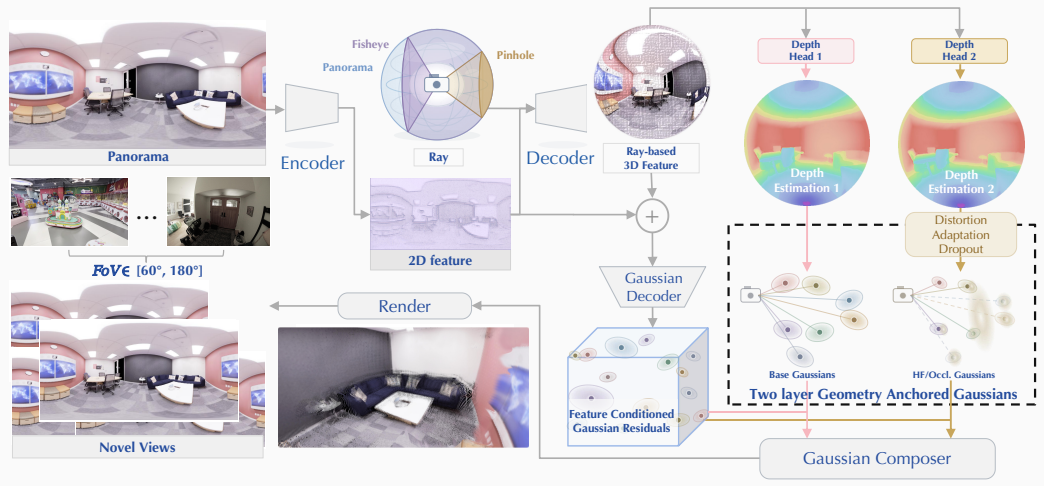

Figure 2: UniSHARP pipeline for universal-camera monocular novel view synthesis. Given a single source image, UniSHARP estimates ray-distance geometry and multi-scale features, initializes two-layer Gaussians in ray-distance space, predicts Feature Conditioned Gaussian residuals, and renders target views with the unified Gaussian representation across perspective, wide-FoV, fisheye, and panoramic cameras.

2 Related work

Multi-image novel view synthesis. Multi-image NVS reconstructs scenes from several posed or nearby observations by exploiting cross-view consistency. Neural radiance fields established continuous scene representations for photorealistic rendering [10], while later anti-aliased variants improved unbounded and high-resolution reconstruction [26, 27]. 3D Gaussian Splatting replaced expensive volume rendering with explicit primitives and real-time rasterization [11]. Feedforward methods further reduce per-scene optimization through learned image-based rendering and multi-view stereo cost volumes [28, 29], while recent sparse-view Gaussian models predict 3DGS representations from image pairs or posed image sets [2, 3, 30]. Large reconstruction models and pose-free systems extend this trend to larger baselines or unconstrained captures [31–34]. Despite their strong quality, these methods fundamentally depend on camera poses, correspondences, or multi-view feature aggregation. UniSHARP instead targets the strictly monocular setting and predicts a renderable Gaussian representation from a single image.

Single-image novel view synthesis. Single-image NVS must infer geometry, appearance, and visibility from priors rather than direct triangulation. Early learning-based approaches predicted neural radiance fields or multiplane images from a single input [35, 36], and other methods used layered geometry, inpainting, or adaptive MPI layouts to handle disocclusion [37–40]. Larger reconstruction and synthesis models show that strong feedforward networks can infer plausible 3D structure from a single photograph [41, 42]. More recently, monocular 3DGS methods directly regress explicit Gaussian primitives, enabling efficient rendering from single images [12, 13, 43]. Generative methods improve extrapolation to larger camera motion through diffusion priors [44–46], but they often trade rendering speed or geometric explicitness for generative flexibility. These works establish single-image NVS as a compelling direction, yet they primarily assume perspective imagery. UniSHARP extends monocular 3DGS to a unified camera setting covering perspective, wide-FoV, fisheye, and panoramic inputs.

Wide-FoV novel view synthesis. Wide-FoV NVS introduces projection distortion, nonuniform angular sampling, and nontrivial boundary topology that are absent in perspective images. Universal monocular geometry estimators show that ray-based representations are important for handling arbitrary cameras [47]. In reconstruction, recent 3DGS systems adapt the rasterizer or camera model to omnidirectional and fisheye inputs [14, 48], and self-calibrating variants jointly optimize camera models, poses, and Gaussian scenes for wide-FoV captures [18, 49, 50]. In parallel, panoramic feedforward methods use spherical radiance fields, spherical cost volumes, Gaussian pyramids,

or Yin-Yang grids for 360-degree synthesis [15–17, 51]. These methods address important non-perspective settings, but they commonly rely on optimized scenes, calibrated captures, or multiple panoramic observations. UniSHARP instead unifies perspective, wide-FoV, fisheye, and panoramic monocular NVS in a single feedforward 3DGS model.

3 Method

Given a single source image $I_s \in \mathbb{R}^{3\times H\times W}$, UniSHARP predicts a set of 3D Gaussian primitives to enable high-fidelity novel-view synthesis. Unlike conventional feedforward methods that regress Gaussian attributes in the image plane, UniSHARP decouples camera projection from scene representation via a unified ray-distance space. This is achieved by: (1) introducing a ray-based universal representation for heterogeneous cameras (Sec. 3.1); (2) composing the scene using Geometry Anchored Gaussians and Feature Conditioned Gaussian residuals (Sec. 3.2); (3) employing a mixed-camera training strategy to supervise rendering across diverse projection types (Sec. 3.3); and (4) enabling pose-free inference by estimating camera geometry from predicted rays (Sec. 3.4).

3.1 Ray Based Universal Representation

Standard image-plane coordinates fail to generalize across heterogeneous projections because a single-pixel displacement corresponds to vastly different angular changes in perspective, fisheye, or panoramic views. To address this, UniSHARP adopts a unified ray-distance space inspired by UniK3D [47]. By decoupling viewing direction from scene range, we ensure that Gaussian primitives defined by 3D centers, covariances, and appearance are optimized in a consistent metric space instead of being tied to projection-specific image grids.

Formally, let $\Omega = \{1,\dots,H\}\times\{1,\dots,W\}$ denote the pixel domain. UniSHARP uses a predicted per-pixel unit ray field

$$\mathbf{r}_p \in \mathbb{S}^2, \qquad \|\mathbf{r}_p\|_2 = 1, \tag{1}$$

and a radial distance $d_p > 0$ from the camera center. The corresponding 3D point is then $\mathbf{x}_p = d_p\mathbf{r}_p$.

This formulation provides a universal coordinate system where Gaussian attributes (placement, scale, and color) are defined consistently across diverse camera models. By measuring spatial footprints along rays rather than in the rasterized plane, UniSHARP enables robust initialization and refinement of Gaussian scenes regardless of the input projection type.

3.2 Model Design

Building upon the unified ray system, UniSHARP first constructs Geometry Anchored Gaussians (3.2.1), providing a camera-unified gaussian space initialization. It then predicts Feature Conditioned Gaussian residuals (3.2.2) from 2D semantic and 3D ray-based features, composing them with the anchors to obtain the final renderable Gaussians. The overall pipeline is illustrated in Figure 2.

3.2.1 Geometry Anchored Gaussians

For each input image, we construct two-layer Geometry Anchored Gaussians on a native ray grid. Let H_g and W_g be the Gaussian grid resolution. At pixel p and layer ℓ, each Geometry Anchored Gaussian is represented as

$$\mathcal{B}_{p,\ell} = \left\{\mathbf{r}_p, \rho_{p,\ell}, \mathbf{s}^0_{p,\ell}, \mathbf{q}^0, \mathbf{c}^0_p, \alpha^0_\ell\right\}, \tag{2}$$

where $\mathbf{r}_p$ is the unit ray, $\rho_{p,\ell}$ is inverse radial distance, $\mathbf{s}^0_{p,\ell}$ is the base scale, $\mathbf{q}^0$ is the identity quaternion, $\mathbf{c}^0_p$ is obtained from the input color, and α^0_ℓ is opacity initialization. The first layer aligns with the visible surface, while the second layer captures disocclusions and high-frequency structures beyond a single surface hypothesis. This second radial layer is predicted by an additional depth head same as the UniK3D radial head, giving the geometry-anchored representation a separate distance hypothesis that can specialize through rendering supervision and regularization.

By allocating primitives according to native ray sampling, UniSHARP enables resolution-adaptive construction that preserves angular consistency and high-frequency details across wide-FoV and panoramic inputs without the distortions of a fixed grid.

3.2.2 Feature Conditioned Gaussian Residuals

While Geometry Anchored Gaussians provide a camera-unified layout, UniSHARP further predicts Feature Conditioned Gaussian residuals to incorporate the semantic context and geometric priors necessary for high-fidelity synthesis. Unlike conventional monocular predictors that feed RGB images and depth images directly into a decoder, UniSHARP predicts residuals within a unified space by fusing 2D semantic image features with 3D ray-based geometric features.

For each geometry-anchored location, a Gaussian decoder predicts a residual tensor $\Delta \in \mathbb{R}^{B\times 14\times L\times H_g\times W_g}$, whose channels correspond to tangent-plane center offsets, inverse-distance, scale, quaternion, color, and opacity residuals. The final Gaussian is obtained by composing the anchor and residual:

$$\mathcal{G}_{p,\ell} = \mathrm{Compose}(\mathcal{B}_{p,\ell}, \Delta_{p,\ell}) = \{\boldsymbol{\mu}_{p,\ell}, \mathbf{s}_{p,\ell}, \mathbf{q}_{p,\ell}, \mathbf{c}_{p,\ell}, \alpha_{p,\ell}\}. \tag{3}$$

3.3 Training Strategy and Objective

Mixed-camera training. UniSHARP is trained under a mixed-camera regime, jointly optimizing perspective, wide-FoV, fisheye, and panoramic data within a single model. During training, a weighted sampler draws source-target image pairs from all supported datasets and groups each mini-batch by dataset for efficient collation and rendering. Although these data differ in image formation process, FoV, and valid target regions, UniSHARP does not introduce camera-specific branches. Instead, each sample is converted into the same ray-based training interface, and all camera types share a unified network architecture. As a result, UniSHARP learns a camera-unified Gaussian representation that transfers supervision across heterogeneous cameras while avoiding separate predictors for individual camera models.

Panoramic distortion adaptation. Equirectangular panoramas oversample polar regions because pixels near the poles correspond to narrower solid angles than those at the equator. To address this, we apply a latitude-dependent probabilistic dropout on the second Gaussian layer during training:

$$p_y = 1 - \frac{\max(\cos\theta_y, 0)}{\max_{y'} \max(\cos\theta_{y'}, 0)}, \tag{4}$$

where θ_y is the latitude of row y. The first layer is always preserved to maintain visible surface coverage. While the first layer is preserved for surface coverage, the second is then selectively suppressed via a Bernoulli mask $m_{p,2} \sim \mathrm{Bernoulli}(p_y)$ that biases opacity residuals. This approach shifts panoramic distortion adaptation from a specialized prediction branch to a training-time allocation strategy.

Objective. Let s and t denote source and target views. The training objective includes appearance supervision, depth supervision, and Gaussian regularization. We use $\hat{\mathbf{I}}_v$, $\hat{\mathbf{A}}_v$, and $\hat{\mathbf{D}}_v$ for rendered color, opacity, and distance at view v, and $\mathbf{I}_v$ and $\mathbf{D}_v$ for image and depth supervision. Appearance supervision encourages the accumulated opacity to cover valid pixels, and applies a perceptual term on target views where novel view artifacts are most visible.

$$\mathcal{L}_{\mathrm{app}} = \lambda_c \sum_{v\in\{s,t\}} \left\| \hat{\mathbf{I}}_v - \mathbf{I}_v \right\|_1 + \lambda_a \sum_{v\in\{s,t\}} \mathrm{BCE}(\hat{\mathbf{A}}_v, \mathbf{1}) + \lambda_p \Phi(\hat{\mathbf{I}}_t, \mathbf{I}_t), \tag{5}$$

where Φ is a perceptual loss computed from deep features and Gram statistics.

Depth supervision anchors both the source geometry used to initialize Gaussians and the target geometry produced after splatting.

$$\mathcal{L}_{\mathrm{dep}} = \lambda_d \left(\left\| \tilde{\mathbf{D}}_s^{-1} - \mathbf{D}_s^{-1} \right\|_1 + \left\| \hat{\mathbf{D}}_t^{-1} - \mathbf{D}_t^{-1} \right\|_1 \right), \tag{6}$$

where $\tilde{\mathbf{D}}_s$ is the first source radial layer and $\hat{\mathbf{D}}_t$ is the rendered target distance.

Gaussian regularization stabilizes degrees of freedom that are weakly constrained by photometric loss. It includes total variation on the second radial layer, floater suppression near abrupt first-layer

Table 1: Composition of the proposed FoV stratified benchmark for universal-camera monocular novel view synthesis. Validation pairs are grouped by effective FoV and projection type, and sample counts denote evaluated source-target pairs.

Camera group	FoV range	Validation datasets	Validation samples
Perspective	60°–90°	DL3DV [52], RealEstate10K [53], Tanks and Temples [54], WildRGB-D [55]	36,873
Wide FoV	90°–140°	OmniRooms-Wide (ours), projected from OmniRooms	10,692
Fisheye	140°–180°	ScanNet++ Fisheye [56]	14,163
Panorama	360°	Replica [57], HM3D [58], OmniRooms (ours)	42,754

inverse-distance changes, and multi-scale Sobel alignment in log-distance space:

$$\mathcal{L}_{\text{geo}} = \lambda_{\text{tv}} \left\langle \left\| \nabla \hat{\mathbf{D}}_{s,2}^{-1} \right\|_1 \right\rangle + \lambda_g \left\langle \hat{\mathbf{O}} \left(1 - \exp\left(- \left[\left\| \nabla \hat{\mathbf{D}}_{s,1}^{-1} \right\|_1 - \tau \right]_+ / \sigma \right) \right) \right\rangle + \lambda_{\text{gi}} \frac{1}{K} \sum_{v \in \{s,t\}} \sum_{k=1}^{K} \left\langle \left\| \nabla_{\text{Sobel}} \mathcal{P}_k \left(\log \hat{\mathbf{D}}_v - \log \mathbf{D}_v \right) \right\|_2 \right\rangle, \tag{7}$$

where $\hat{\mathbf{O}}$ is the predicted Gaussian opacity, K is the number of depth pyramid scales, $\mathcal{P}_k$ downsamples its input to the k-th scale, and $[x]_+ = \max(x, 0)$. The three terms respectively smooth the inverse distance of the second source layer, suppress opaque floaters around sharp first-layer distance discontinuities, and align rendered and supervised distance edges with multi-scale Sobel gradients.

For equirectangular panoramas, horizontal finite differences use circular boundary handling to respect the wrap-around topology. The full training objective is

$$\mathcal{L} = \mathcal{L}_{\text{app}} + \mathcal{L}_{\text{dep}} + \mathcal{L}_{\text{geo}}. \tag{8}$$

3.4 Extension to Pose-Free Model

UniSHARP naturally supports a calibrated setting where the input camera model and intrinsics are known. For in-the-wild deployment, however, a user may provide only a single RGB image. We therefore introduce a pose-free model that replaces external calibration with camera geometry recovered from the predicted UniK3D ray field. Specifically, we determine the camera model from the angular coverage of the predicted rays and then recover the corresponding rendering geometry. For perspective and fisheye inputs, the ray field is converted into a compact parametric camera by fitting pinhole intrinsics or Fisheye parameters, while panoramic inputs use the deterministic spherical camera model. The recovered camera is used consistently for ray-distance Gaussian initialization and novel-view rendering, so the Gaussian predictor remains shared with the calibrated model rather than becoming a separate branch. This design lets UniSHARP render orbit and forward-motion views from an uncalibrated image while preserving the same feedforward inference pipeline.

4 Experiments

Training. We train one unified model on a mixture of perspective, wide-FoV, fisheye, and panoramic datasets. For perspective images, we use RealEstate10K [53], DL3DV [52], and WildRGB-D [55]. For wide-FoV images, we use OmniRooms-Wide, which is projected from our simulated indoor panoramic dataset. For fisheye images, we use ScanNet++ Fisheye [56]. For panoramic images, we use HM3D panorama datasets [58] and OmniRooms, our simulated indoor ERP dataset constructed with the AirSim platform [59] by rendering collision-free camera trajectories in synthetic indoor scenes. All datasets are converted into source-target training pairs under their native camera models. At each iteration, we first sample a dataset according to a fixed dataset-level distribution and then draw a batch from that dataset. For datasets without ground-truth depth, we use UniK3D [47] to generate pseudo depth labels. Additional implementation details are provided in Appendix A.1.

4.1 Benchmark & Metrics.

We evaluate UniSHARP on monocular NVS across perspective, wide-FoV, fisheye, and panoramic cameras. To address the limitations of evaluations tied to single projection families, we introduce a

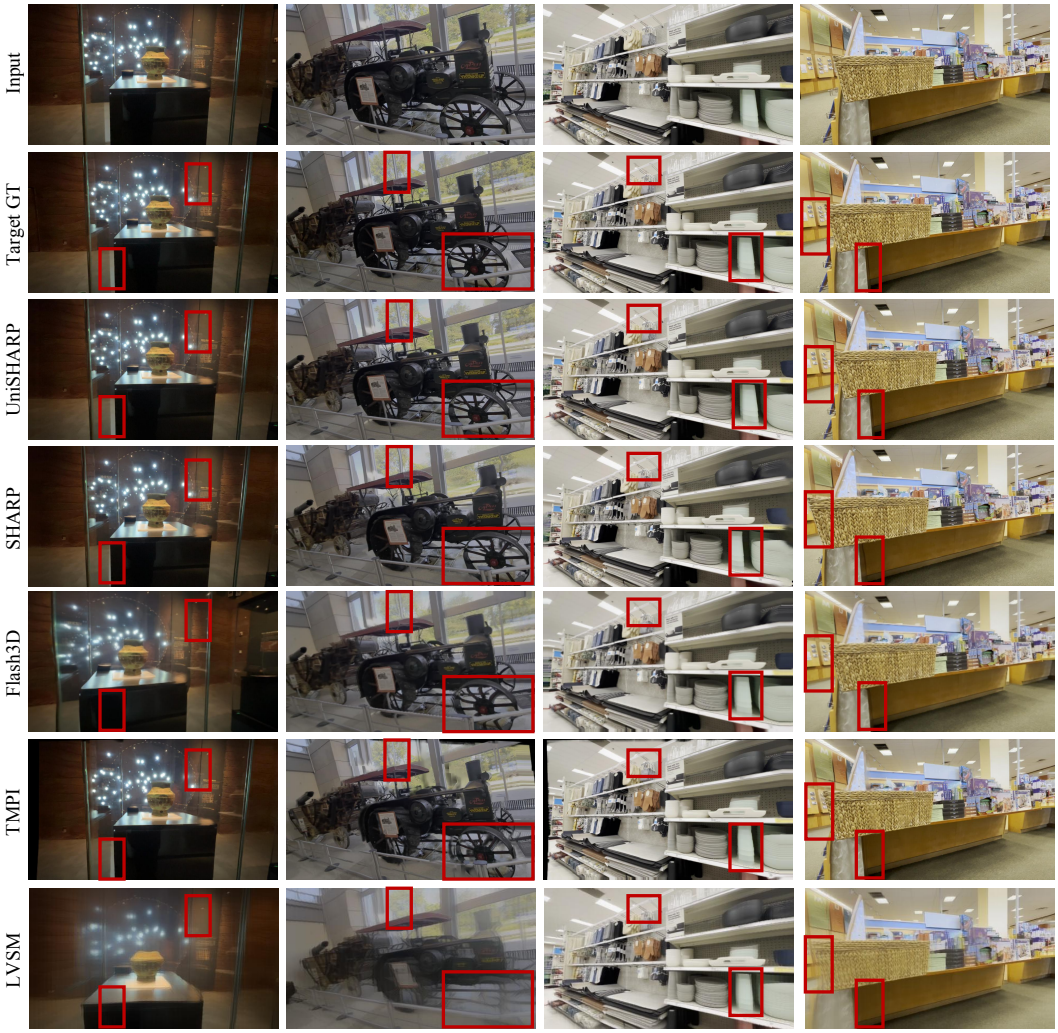

Figure 3: Qualitative comparison on perspective novel view synthesis. Given a single source image, UniSHARP produces sharper target-view geometry and fewer disocclusion artifacts than perspective monocular baselines, while preserving view-consistent scene structure.

Table 2: Quantitative comparison on perspective datasets included in the mixed training distribution. The best results are shown in red and the second-best results are in orange.

Method	WildRGB-D [55]			DL3DV [52]			RealEstate10K [53]		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
TMPI [40]	15.969	0.493	0.330	16.002	0.490	0.340	21.707	0.768	0.142
LVSM [42]	15.970	0.362	0.279	15.848	0.446	0.374	22.109	0.731	0.138
Flash3D [13]	17.378	0.553	0.395	18.418	0.574	0.179	23.573	0.785	0.172
SHARP [12]	18.300	0.568	0.212	17.368	0.545	0.286	24.215	0.789	0.101
UniSHARP	**21.556**	**0.674**	**0.143**	**19.468**	**0.594**	**0.196**	**24.495**	**0.795**	**0.087**

FoV stratified benchmark (Table 1). This benchmark provides a unified protocol to diagnose model behavior as camera geometry scales from narrow perspective to full 360° equirectangular inputs.

OmniRooms. OmniRooms is a simulated indoor ERP dataset collected via AirSim [59], with OmniRooms-Wide derived by projecting these panoramas into 130° equidistant fisheye views. For each anchor point on a 0.5m voxel grid, we render one central camera and 29 others randomly sampled within a local axis-aligned cube of edge length 30 cm around the source camera. Each frame is rendered as a 1024×2048 ERP image and all cameras share a fixed orientation.

Figure 4: Qualitative comparison on panorama novel view synthesis. UniSHARP reconstructs coherent Gaussian geometry from a single panoramic input and renders sharper target views with fewer distortion-induced artifacts.

Table 3: Quantitative comparison on panoramic novel view synthesis. Results are reported on real and simulated ERP datasets using PSNR↑, SSIM↑, and LPIPS↓; best results are shown in red.

	HM3D [58]			OmniRooms			Replica [57]		
Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
PanoDreamer [60]	21.856	0.708	0.152	18.567	0.692	0.240	21.289	0.793	0.137
Matrix3D [61]	23.398	0.793	0.114	21.635	0.799	0.144	25.379	0.887	0.069
UniSHARP	**29.244**	**0.895**	**0.065**	**24.004**	**0.856**	**0.123**	**30.182**	**0.933**	**0.038**

Benchmark details. To align with single-source monocular NVS, we restrict target views to locally reachable positions: we require $> 60\%$ source-target overlap, a camera-center distance < 0.5m, and an image-index gap < 10. This design focuses evaluation on geometry inference under meaningful motion rather than unconstrained long-range hallucination. The protocol serves as a unified testbed for universal-camera NVS, enabling a direct diagnostic of how rendering quality scales across perspective, wide-FoV, fisheye, and 360-degree projections. We evaluate in a single-source, multi-target setting, using the first frame of each sequence as the source and the subsequent ten frames as targets. Results are reported using PSNR, SSIM, and LPIPS, averaged over all valid target views.

Baselines. For perspective novel view synthesis, we compare with representative single-image 3D Gaussian regressors, including SHARP [12] and Flash3D [13], as well as methods based on different scene representations, including LVSM [42], a large view synthesis model with minimal 3D inductive bias, and TMPI [40], a tiled multiplane-image representation for practical 3D photography. For wide-FoV, fisheye, and panoramic novel view synthesis, we compare with methods that cover

Table 4: Quantitative comparison on wide-FoV and fisheye novel view synthesis. Results are reported on OmniRooms-Wide and ScanNet++ Fisheye.

	ScanNet++ Fisheye [56]			OmniRooms-Wide		
Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
PanoDreamer [60]	15.131	0.682	0.383	18.062	0.760	0.199
Matrix3D [61]	16.382	0.690	0.371	18.881	0.795	0.173
UniSHARP	**20.660**	**0.771**	**0.184**	**25.243**	**0.854**	**0.076**

different generation paradigms: PanoDreamer [60], an optimization-based single-image-to-360 scene method using diffusion and 3DGS, and Matrix3D [61], a diffusion-based video generation model.

4.2 Qualitative and Quantitative Evaluation

Perspective performance.

Table 2 reports the in-domain results on RealEstate10K, DL3DV, and WildRGB-D. UniSHARP achieves the best PSNR and SSIM across the perspective datasets and the best or second-best LPIPS, showing that universal-camera training preserves strong standard perspective rendering. The gains are consistent across real-estate, object-centric, and in-the-wild scenes, indicating that the shared ray-distance Gaussian representation improves fidelity without overfitting to a single perspective dataset.

Table 5: Zero-shot perspective evaluation on Tanks and Temples [54].

Method	PSNR↑	SSIM↑	LPIPS↓
TMPI [40]	13.623	0.399	0.440
LVSM [42]	14.575	0.379	0.476
Flash3D [13]	15.985	**0.511**	0.312
SHARP [12]	15.964	0.502	0.301
UniSHARP	**16.315**	0.498	**0.282**

Table 5 evaluates out-of-domain generalization on the Tanks and Temples dataset. UniSHARP obtains the best PSNR and LPIPS compared with the strongest baselines. The SSIM remains close to Flash3D, suggesting that the unified representation preserves cross-dataset generalization while improving overall reconstruction fidelity.

Wide-FoV, fisheye, and panoramic performance.

Tables 3 and Table 4 evaluate UniSHARP on non-perspective cameras. On OmniRooms-Wide, UniSHARP consistently improves PSNR, SSIM, and LPIPS over both baselines. This shows that the ray-distance Gaussian representation remains effective for wide-FoV inputs, where the model must handle stronger projection distortion and larger angular coverage than standard perspective images. On ScanNet++ Fisheye, UniSHARP also outperforms PanoDreamer and Matrix3D by a clear margin, suggesting that the same geometry-aware parameterization transfers from projected wide-FoV views to native fisheye cameras. For panoramic novel view synthesis, UniSHARP achieves the best PSNR, SSIM, and LPIPS on HM3D, OmniRooms, and the out-of-domain Replica dataset. These results indicate that the proposed camera-unified design remains stable as the evaluation moves from wide-FoV and fisheye inputs to full 360° panoramas.

Table 6: Pose-free evaluation on WildRGB-D. Ours uses the available camera parameters, while Ours (pose-free) estimates camera geometry from predicted rays.

Setting	PSNR↑	SSIM↑	LPIPS↓
Ours (pose-free)	20.850	0.647	0.157
Ours	**21.556**	**0.674**	**0.143**

Pose-free performance. Table 6 evaluates the pose-free setting on WildRGB-D. Ours uses the available camera parameters, while Ours (pose-free) estimates the camera model and rendering geometry from the predicted ray field. The pose-free variant maintains competitive rendering quality without requiring camera calibration, demonstrating the practical value of ray-based camera recovery for unconstrained monocular inputs.

4.3 Ablation Studies

We conduct ablations on WildRGB-D and HM3D to analyze the main architectural and objective components of UniSHARP.

Table 7: Ablation study of model design components on WildRGB-D and HM3D. Each variant removes one component from the full model.

	WildRGB-D [55]			HM3D [58]		
Variant	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Full model	**21.56**	**0.674**	**0.143**	**29.24**	**0.895**	**0.065**
w/o native resolution allocation	21.21	0.664	0.155	28.72	0.872	0.071
w/o second Gaussian layer	20.63	0.642	0.168	28.29	0.877	0.083
w/o panoramic distortion adaptation	21.50	0.672	0.145	28.43	0.880	0.077
w/ depth-RGB images input	20.38	0.631	0.180	28.04	0.868	0.089

Model design. Table 7 summarizes the contribution of the main architectural components. Replacing the proposed 2D semantic and 3D geometric features with direct depth-RGB inputs causes the largest degradation on both datasets, showing that the learned feature space provides stronger context than direct RGB-depth conditioning. Removing the second Gaussian layer also hurts performance on both datasets, confirming the importance of an additional distance hypothesis for disocclusions and wide angular coverage. Native resolution allocation remains important for preserving input detail, while panoramic distortion adaptation has a clearer effect on HM3D, where regularizing equirectangular distortion is more critical than in perspective images.

5 Conclusion

We presented UniSHARP, a universal-camera feedforward 3DGS framework for monocular novel view synthesis. Starting from the observation that perspective-trained Gaussian regressors do not transfer reliably to heterogeneous camera systems, UniSHARP reformulates Gaussian prediction in a shared ray-distance space and composes Geometry Anchored Gaussians with Feature Conditioned Gaussian residuals. This design preserves the efficiency of single-image Gaussian regression while supporting perspective, wide-FoV, fisheye, and panoramic inputs within one prediction model. To evaluate this setting systematically, we further introduced a FoV stratified benchmark covering real and simulated scenes across narrow perspective to full panoramic cameras. Experiments on this benchmark show that UniSHARP maintains strong performance on perspective datasets and substantially improves novel view synthesis across diverse camera systems. We hope this work provides a practical foundation for monocular 3D Gaussian rendering in real-world imaging systems beyond the pinhole camera model.

References

[1] Xin Lin, Shi Luo, Xiaojun Shan, Xiaoyu Zhou, Chao Ren, Lu Qi, Ming-Hsuan Yang, and Nuno Vasconcelos. Hqgs: High-quality novel view synthesis with gaussian splatting in degraded scenes. In *ICLR*, 2025. 2

[2] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *CVPR*, 2024. 3

[3] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *ECCV*, 2024. 3

[4] Meixi Song, Xin Lin, Dizhe Zhang, Haodong Li, Xiangtai Li, Bo Du, and Lu Qi. D^2GS: Depth-and-density guided gaussian splatting for stable and accurate sparse-view reconstruction. *arXiv preprint arXiv:2510.08566*, 2025.

[5] Hao Ren, Yiming Zeng, Zetong Bi, Zhaoliang Wan, Junlong Huang, and Hui Cheng. Prior does matter: Visual navigation via denoising diffusion bridge models. In *CVPR*, pages 12100–12110, 2025.

[6] Xiaoyuan Wang, Yizhou Zhao, Botao Ye, Shan Xiaojun, Weijie Lyu, Lu Qi, Kelvin Chan, Yinxiao Li, and Ming-Hsuan Yang. Holigs: Holistic gaussian splatting for embodied view synthesis. *NeurIPS*, 38:96820–96849, 2026.

[7] Chunjiang Liu, Xiaoyuan Wang, Qingran Lin, Albert Xiao, Haoyu Chen, Shizheng Wen, Hao Zhang, Lu Qi, Ming-Hsuan Yang, Laszlo A Jeni, et al. Mosiv: Multi-object system identification from videos. *arXiv preprint arXiv:2603.06022*, 2026.

[8] Jingtong Yue, Zhiwei Lin, Xin Lin, Xiaoyu Zhou, Xiangtai Li, Lu Qi, Yongtao Wang, and Ming-Hsuan Yang. Roburcdet: Enhancing robustness of radar-camera fusion in bird's eye view for 3d object detection. *arXiv preprint arXiv:2502.13071*, 2025.

[9] Hao Ren, Zetong Bi, Yiming Zeng, Zhaoliang Wan, Lu Qi, and Hui Cheng. Strnet: Visual navigation with spatio-temporal representation through dynamic graph aggregation. In *CVPR*, pages 42464–42473, 2026. 2

[10] 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. *Communications of the ACM*, 2021. 2, 3

[11] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *TOG*, 2023. 2, 3

[12] Lars Mescheder, Wei Dong, Shiwei Li, Xuyang Bai, Marcel Santos, Peiyun Hu, Bruno Lecouat, Mingmin Zhen, Amael Delaunoy, Tian Fang, Yanghai Tsin, Stephan R. Richter, and Vladlen Koltun. Sharp monocular view synthesis in less than a second. In *ICLR*, 2026. 2, 3, 7, 8, 9, 15

[13] Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Flash3d: Feed-forward generalisable 3d scene reconstruction from a single image. *arXiv preprint arXiv:2406.04343*, 2024. 2, 3, 7, 8, 9

[14] Longwei Li, Huajian Huang, Sai-Kit Yeung, and Hui Cheng. Omnigs: Fast radiance field reconstruction using omnidirectional gaussian splatting. *arXiv preprint arXiv:2404.03202*, 2024. 2, 3

[15] Zheng Chen, Chenming Wu, Zhelun Shen, Chen Zhao, Weicai Ye, Haocheng Feng, Errui Ding, and Song-Hai Zhang. Splatter-360: Generalizable 360 gaussian splatting for wide-baseline panoramic images. In *CVPR*, 2025. 4

[16] Cheng Zhang, Haofei Xu, Qianyi Wu, Camilo Cruz Gambardella, Dinh Phung, and Jianfei Cai. Pansplat: 4k panorama synthesis with feed-forward gaussian splatting. In *CVPR*, 2025.

[17] Suyoung Lee, Jaeyoung Chung, Kihoon Kim, Jaeyoo Huh, Gunhee Lee, Minsoo Lee, and Kyoung Mu Lee. Omnisplat: Taming feed-forward 3d gaussian splatting for omnidirectional images with editable capabilities. In *CVPR*, 2025. 4

[18] Youming Deng, Wenqi Xian, Guandao Yang, Leonidas Guibas, Gordon Wetzstein, Steve Marschner, and Paul Debevec. Self-calibrating gaussian splatting for large field-of-view reconstruction. In *ICCV*, 2025. 2, 3

[19] Xian Ge, Yuling Pan, Yuhang Zhang, Xiang Li, Weijun Zhang, Dizhe Zhang, Zhaoliang Wan, Xin Lin, Xiangkai Zhang, Juntao Liang, et al. Airsim360: A panoramic simulation platform within drone view. *arXiv preprint arXiv:2512.02009*, 2025. 2

[20] Xin Lin, Meixi Song, Dizhe Zhang, Wenxuan Lu, Haodong Li, Bo Du, Ming-Hsuan Yang, Truong Nguyen, and Lu Qi. Depth any panoramas: A foundation model for panoramic depth estimation. *arXiv preprint arXiv:2512.16913*, 2025.

[21] Xiangkai Zhang, Dizhe Zhang, WenZhuo Cao, Zhaoliang Wan, Yingjie Niu, Lu Qi, Xu Yang, and Zhiyong Liu. Fly360: Omnidirectional obstacle avoidance within drone view. *arXiv preprint arXiv:2603.06573*, 2026.

[22] Haoran Feng, Dizhe Zhang, Xiangtai Li, Bo Du, and Lu Qi. Dit360: High-fidelity panoramic image generation via hybrid training. *arXiv preprint arXiv:2510.11712*, 2025.

[23] Yuheng Liu, Xin Lin, Xinke Li, Baihan Yang, Chen Wang, Kalyan Sunkavalli, Yannick Hold-Geoffroy, Hao Tan, Kai Zhang, Xiaohui Xie, Zifan Shi, and Yiwei Hu. Omniroam: World wandering via long-horizon panoramic video generation. *arXiv preprint arXiv:2603.30045*, 2026.

[24] Xin Lin, Xian Ge, Dizhe Zhang, Zhaoliang Wan, Xianshun Wang, Xiangtai Li, Wenjie Jiang, Bo Du, Dacheng Tao, Ming-Hsuan Yang, et al. One flight over the gap: A survey from perspective to panoramic vision. *arXiv preprint arXiv:2509.04444*, 2025.

[25] Changpeng Wang, Xin Lin, Junhan Liu, Yuheng Liu, Zhen Wang, Donglian Qi, Yunfeng Yan, and Xi Chen. PanoWorld: Towards spatial supersensing in 360° panorama world. *arXiv preprint arXiv:2605.13169*, 2026. 2

[26] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022. 3

[27] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *ICCV*, 2023. 3

[28] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P. Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, 2021. 3

[29] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *ICCV*, 2021. 3

[30] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *CVPR*, 2025. 3

[31] Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, Dahua Lin, and Bo Dai. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *arXiv preprint arXiv:2505.23716*, 2025. 3

[32] Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jisang Han, Jiaolong Yang, Chong Luo, and Seungryong Kim. Pf3plat: Pose-free feed-forward 3d gaussian splatting for novel view synthesis. In *ICML*, 2025.

[33] Anpei Chen, Haofei Xu, Stefano Esposito, Siyu Tang, and Andreas Geiger. Lara: Efficient large-baseline radiance fields. In *ECCV*, 2024.

[34] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *ECCV*, 2024. 3

[35] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, 2021. 3

[36] Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *CVPR*, 2020. 3

[37] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *CVPR*, 2020. 3

[38] Varun Jampani, Huiwen Chang, Kyle Sargent, Abhishek Kar, Richard Tucker, Michael Krainin, Dominik Kaeser, William T. Freeman, David Salesin, Brian Curless, Noah Snavely, and Ce Liu. Slide: Single image 3d photography with soft layering and depth-aware inpainting. In *ICCV*, 2021.

[39] Yuxuan Han, Ruicheng Wang, and Jiaolong Yang. Single-view view synthesis in the wild with learned adaptive multiplane images. *ACM Transactions on Graphics*, 41(4), 2022.

[40] Numair Khan, Eric Penner, Douglas Lanman, and Lei Xiao. Tiled multiplane images for practical 3d photography. In *ICCV*, 2023. 3, 7, 8, 9

[41] Yuan Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 3

[42] Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. In *ICLR*, 2025. 3, 7, 8, 9

[43] Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *CVPR*, 2024. 3

[44] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric R. Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, and Jiajun Wu. Zeronvs: Zero-shot 360-degree view synthesis from a single real image. In *CVPR*, 2024. 3

[45] Hanwen Liang, Junli Cao, Vidit Goel, Guocheng Qian, Sergei Korolev, Demetri Terzopoulos, Konstantinos N. Plataniotis, Sergey Tulyakov, and Jian Ren. Wonderland: Navigating 3d scenes from a single image. In *CVPR*, 2025.

[46] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Mueller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *CVPR*, 2025. 3

[47] Luigi Piccinelli, Christos Sakaridis, Mattia Segu, Yung-Hsu Yang, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unik3d: Universal camera monocular 3d estimation. In *CVPR*, 2025. 3, 4, 6, 14

[48] Zimu Liao, Siyan Chen, Rong Fu, Yi Wang, Zhongling Su, Hao Luo, Li Ma, Linning Xu, Bo Dai, Hengjie Li, Zhilin Pei, and Xingcheng Zhang. Fisheye-gs: Lightweight and extensible gaussian splatting module for fisheye cameras. *arXiv preprint arXiv:2409.04751*, 2024. 3

[49] Huajian Huang, Yingshu Chen, Longwei Li, Hui Cheng, Tristan Braud, Yajie Zhao, and Sai-Kit Yeung. Sc-omnigs: Self-calibrating omnidirectional gaussian splatting. In *ICLR*, 2025. 3

[50] Zhengxian Yang, Fei Xie, Xutao Xue, Rui Zhang, Taicheng Huang, Yang Liu, Mengqi Ji, and Tao Yu. Directfisheye-gs: Enabling native fisheye input in gaussian splatting with cross-view joint optimization. *arXiv preprint arXiv:2604.00648*, 2026. 3

[51] Zheng Chen, Yan-Pei Cao, Yuan-Chen Guo, Chen Wang, Ying Shan, and Song-Hai Zhang. Panogrf: Generalizable spherical radiance fields for wide-baseline panoramas. In *NeurIPS*, 2023. 4

[52] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, Xuanmao Li, Xingpeng Sun, Rohan Ashok, Aniruddha Mukherjee, Hao Kang, Xiangrui Kong, Gang Hua, Tianyi Zhang, Bedrich Benes, and Aniket Bera. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, 2024. 6, 7

[53] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ACM Transactions on Graphics*, 37 (4), 2018. 6, 7

[54] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017. 6, 9

[55] Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd objects in the wild: Scaling real-world 3d object learning from rgb-d videos. In *CVPR*, 2024. 6, 7, 10, 14

[56] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Niessner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023. 6, 9

[57] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 6, 8

[58] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M. Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X. Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *NeurIPS Datasets and Benchmarks Track*, 2021. 6, 8, 10, 14

[59] Xian Ge, Yuling Pan, Yuhang Zhang, Xiang Li, Weijun Zhang, Dizhe Zhang, Zhaoliang Wan, Xin Lin, Xiangkai Zhang, Juntao Liang, et al. Airsim360: A panoramic simulation platform within drone view. *arXiv preprint arXiv:2512.02009*, 2025. 6, 7

[60] Avinash Paliwal, Xilong Zhou, Andrii Tsarov, and Nima Khademi Kalantari. Panodreamer: Optimization-based single image to 360 3d scene with diffusion. In *SIGGRAPH Asia Conference Papers*, 2025. 8, 9, 16

[61] Yuanxun Lu, Jingyang Zhang, Tian Fang, Jean-Daniel Nahmias, Yanghai Tsin, Long Quan, Xun Cao, Yao Yao, and Shiwei Li. Matrix3d: Large photogrammetry model all-in-one. In *CVPR*, 2025. 8, 9, 16

[62] Zixun Huang, Cho-Ying Wu, Yuliang Guo, Xinyu Huang, and Liu Ren. 3dgeer: 3d gaussian rendering made exact and efficient for generic cameras. *arXiv preprint arXiv:2505.24053*, 2026. 14

[63] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 14

A Additional Experiments and Ablations

A.1 Implementation Details

All experiments are conducted on 8 H20 GPUs. UniSHARP uses the feature-only architecture described in Sec. 3, with a UniK3D ViT-L backbone initialized from pretrained weights [47]. For wide-FoV and fisheye rendering, we use the 3DGEER generic-camera Gaussian rasterizer [62]. We optimize the model with Adam [63] for 10^6 iterations using a 10^4-iteration warmup followed by cosine learning-rate decay. The learning rate decays from 1.0×10^{-5} to 1.0×10^{-6} for the depth head, and from 1.2×10^{-4} to 1.6×10^{-5} for the Gaussian decoder and prediction modules. The loss weights are $\lambda_c = 1.0$, $\lambda_a = 1.0$, $\lambda_p = 1.0$, $\lambda_d = 0.5$, $\lambda_{\text{tv}} = 1.0$, $\lambda_g = 0.5$, $\lambda_{\text{gi}} = 0.5$.

A.2 Training Objective Ablation

Table 8: Ablation study of the main training losses on WildRGB-D and HM3D. Each variant removes one loss term from the full objective to measure its contribution to rendering quality.

	WildRGB-D [55]			HM3D [58]		
Variant	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Full objective	**21.56**	**0.674**	**0.143**	**29.24**	**0.895**	**0.065**
w/o perceptual appearance loss	21.18	0.663	0.158	28.69	0.884	0.078
w/o target rendered depth loss	20.42	0.628	0.206	27.12	0.842	0.136
w/o second-layer TV regularization	21.30	0.667	0.151	28.56	0.879	0.084
w/o floater suppression	21.11	0.659	0.162	27.94	0.866	0.153

Training objective. Table 8 analyzes the main loss terms. The full objective achieves the best overall performance, reaching 21.56 PSNR and 0.143 LPIPS on WildRGB-D, and 29.24 PSNR and 0.065 LPIPS on HM3D. Removing target rendered depth supervision causes the largest PSNR drop among the loss ablations, reducing PSNR to 20.42 on WildRGB-D and 27.12 on HM3D, while also increasing LPIPS to 0.206 and 0.136. This shows that source-side depth supervision alone cannot constrain the Gaussian scene after view transformation; supervising the rendered target depth is critical for maintaining cross-view geometry and suppressing view-dependent distortions. The perceptual appearance loss improves visual fidelity, while second-layer TV regularization and floater

suppression stabilize the Gaussian field. Floater suppression is particularly important for panoramic scenes, where removing it increases HM3D LPIPS from 0.065 to 0.153 due to unstable second-layer Gaussians near depth discontinuities.

A.3 Fisheye Dataset Visualization

Figure 5: Visualization of the fisheye validation data used in our benchmark. The samples illustrate the strong radial distortion and wide angular coverage that distinguish native fisheye novel view synthesis from standard perspective evaluation.

A.4 Panoramic Inference via Cubemap Decomposition

As discussed in Sec. 1, SHARP [12] maps every pixel in normalized image space under a pinhole camera assumption and therefore cannot directly ingest equirectangular panoramas or other non-pinhole inputs. A common workaround is to decompose the panorama into six cubemap faces, run SHARP independently on each face, and then fuse the resulting Gaussian predictions before rendering back to the panoramic domain. However, this pipeline inherits the limitations noted in the introduction: each face is processed in isolation under a different local pinhole approximation, so the predicted Gaussian fields are not globally consistent across face boundaries. When the per-face renderings are stitched into a full panorama, the inconsistency manifests as prominent seams and view-dependent discontinuities, especially near geometric edges and depth boundaries.

Figure 6 visualizes this failure mode on a representative panoramic sample. The cubemap-based SHARP baseline produces clearly visible stitching artifacts at face junctions, whereas UniSHARP renders a coherent panoramic target view without such seams. This comparison supports our design choice to avoid pinhole-specific re-projection heuristics and instead predict Gaussians in a camera-unified representation that remains valid across the full 360° field of view.

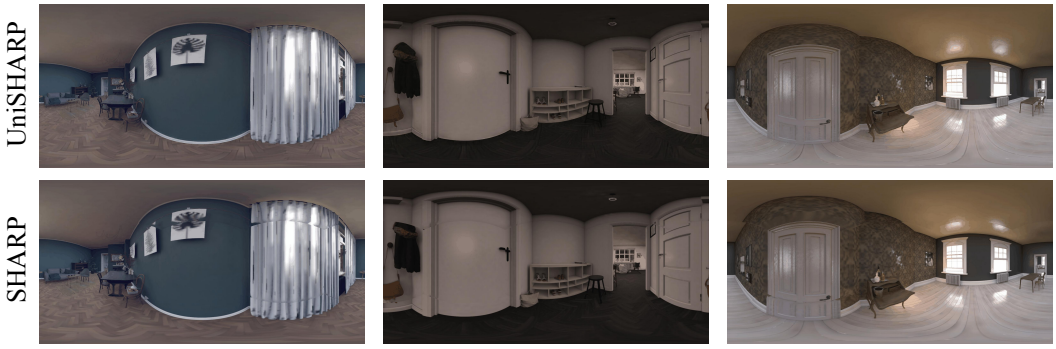

Figure 6: Comparison of panoramic novel view synthesis with a cubemap-based SHARP baseline and UniSHARP. For SHARP, we split the source panorama into cubemap faces, run feedforward inference on each face separately, and stitch the rendered target views back into equirectangular format. The stitched result exhibits large cubemap seams caused by inconsistent cross-face geometry. In contrast, UniSHARP operates directly on the panoramic input in unified ray-distance space and produces a seamless target rendering without cubemap decomposition or post-stitching.

Table 9: Inference time comparison for single-image novel view synthesis. Runtime is measured under the same evaluation setting; relative runtime is reported with respect to UniSHARP.

Method	Runtime ↓	Relative runtime ↓
UniSHARP	**3.1s**	–
PanoDreamer [60]	8.6s	2.8×
Matrix3D [61]	38.8s	12.5×

A.5 Inference Time Comparison

Inference efficiency. Table 9 compares the inference time of UniSHARP with panoramic baselines. UniSHARP completes inference in 3.1 seconds, while PanoDreamer and Matrix3D require 8.6 seconds and 38.8 seconds, respectively. PanoDreamer and Matrix3D are therefore 2.8× and 12.5× slower than UniSHARP. The speed advantage comes from the model design: UniSHARP predicts the complete Gaussian representation with a single feedforward pass and directly renders it, avoiding per-scene optimization, diffusion sampling, or iterative video generation at inference time.

B Limitations

UniSHARP is a feedforward Gaussian prediction model rather than a generative scene completion method. With mixed-camera training, the model can acquire a degree of extrapolation ability and can handle moderate disocclusions around the input view. However, when target views expose large regions that are completely outside the source image, the model has limited evidence for hallucinating unseen content. In such cases, holes or weakly supported structures may appear near outer boundary regions. Improving long-range extrapolation while preserving the efficiency and geometric consistency of feedforward Gaussian prediction remains an interesting direction for future work.

C Societal Impact

Universal-camera monocular novel view synthesis can broaden the use of 3D perception and rendering systems beyond standard perspective imagery. By supporting perspective, wide-FoV, fisheye, and panoramic inputs in a unified model, UniSHARP may benefit applications such as embodied AI, robotics, AR/VR content creation, immersive telepresence, and spatial documentation. The ability to infer renderable 3D structure from a single image can reduce capture requirements and make 3D content generation more accessible for devices with diverse camera systems. The proposed

benchmark and dataset can also support more systematic evaluation of camera-general view synthesis methods, encouraging future research on robust and geometry-aware spatial intelligence.

D Benchmark Details

This section supplements the benchmark description in Sec. 4.1. The main paper introduces the benchmark composition and reports results across perspective, wide-FoV, fisheye, and panoramic camera groups. Here we provide additional details on data splits, pair construction, camera metadata, metric computation, baseline adaptation, and evaluation protocol, so that the benchmark can be reproduced and reused by future work.

D.1 Dataset Splits and Scene Selection

For all datasets, validation samples are constructed in a single-source multi-target format. Each sample contains one source image, one or more target views, the corresponding camera parameters when available, and metadata describing the projection type and effective FoV. Target-view RGB images are used only for evaluation and are never provided to the model during inference.

For existing datasets, we follow the official validation or test splits whenever they are available. When a dataset does not provide a standard split for monocular novel view synthesis, we construct a held-out validation split at the sequence or scene level to avoid source-target leakage across training and evaluation. The evaluation samples are fixed before testing, which makes the benchmark deterministic and avoids dependence on dataloader randomness.

Perspective datasets. For RealEstate10K, DL3DV, Tanks and Temples, and WildRGB-D, we use the perspective camera metadata provided by the original datasets. RealEstate10K follows its official test split, while the remaining datasets use held-out scene or sequence subsets constructed for monocular novel view synthesis. Tanks and Temples is used only as a held-out perspective evaluation set to measure out-of-domain generalization.

Wide-FoV dataset. OmniRooms-Wide is derived from OmniRooms by projecting equirectangular panoramas into wide-FoV views. We use a 130° equidistant projection at 1024×1024 resolution. Source and target views within the same local group share the same camera orientation, so the benchmark isolates translation-induced view synthesis rather than mixing translation and rotation. The metadata records the projection type, FoV, valid image radius, and camera-to-world transform for each rendered view.

Fisheye dataset. For ScanNet++ Fisheye, we preserve the native fisheye camera model and use the provided camera calibration when available. Frames without valid calibration or depth are skipped before pair construction.

Panoramic datasets. For HM3D, Replica, and OmniRooms, images are evaluated in equirectangular projection. Replica is used as an out-of-domain panoramic validation set, while HM3D and OmniRooms evaluate panoramic rendering quality under real-scanned and simulated indoor scenes, respectively. The OmniRooms panoramic split is generated from fixed simulated trajectories, and all source-target groups are defined before evaluation. The complete scene and frame identifiers are provided with the released benchmark metadata.

D.2 OmniRooms Construction

OmniRooms is a simulated indoor equirectangular panorama dataset built to provide dense local camera trajectories for monocular panoramic novel view synthesis. For each valid anchor location, we render one central source camera and multiple nearby target cameras within a local neighborhood. All cameras associated with the same anchor share the same orientation, so the benchmark isolates translation-induced view synthesis rather than mixing translation and rotation. Each panorama is rendered at 1024×2048 resolution.

Anchor sampling. Anchor locations are sampled on a 0.5m voxel grid in navigable indoor regions before local camera expansion. We retain only anchor centers whose height coordinate satisfies

$60 \leq Z \leq 180$ cm, which removes floor-level, ceiling-level, and otherwise implausible camera centers. Collision, navigability, and degenerate-rendering checks are applied during simulation and data filtering. A rendered sample is discarded if its RGB image, depth map, camera metadata, or source-target visibility overlap does not satisfy the benchmark requirements.

Target sampling. For each retained anchor, target cameras are sampled in a local axis-aligned cube of edge length 30 cm around the source camera. Each anchor produces 30 camera positions: the original center and 29 randomly perturbed target centers. This local sampling range evaluates nearby-view synthesis while still introducing meaningful parallax and disocclusions.

Rendering settings. Each OmniRooms sample contains an RGB panorama, aligned depth, and camera metadata. Panoramic RGB images are rendered at 1024×2048 resolution. Depth and surface geometry are used for supervision, visibility filtering, and analysis, but are not provided as model input at test time. RGB values are decoded and normalized to $[0, 1]$ before evaluation.

D.3 Source-Target Pair Filtering

The benchmark focuses on local monocular novel view synthesis. We therefore filter source-target pairs to avoid evaluating unconstrained long-range hallucination. A pair is retained only if it satisfies three constraints: source-target overlap at least 60%, camera-center distance smaller than 0.5m, and image-index gap at most 10. These constraints focus the evaluation on geometry and disocclusion reasoning under meaningful local motion.

Camera distance. Camera-center distance is computed as the Euclidean distance between the source and target camera centers in the dataset coordinate system:

$$d(s, t) = \|\mathbf{c}_s - \mathbf{c}_t\|_2. \tag{9}$$

The pair is valid if $d(s, t) < 0.5$m.

Frame-index gap. For sequence-based datasets, the index gap is computed using the original frame order. For real-world video or image-sequence datasets, frames follow the temporal or acquisition order provided by the original data. For simulated panoramic data, frames are ordered according to the generated local camera groups. For datasets without native video order, source and target indices are fixed by the benchmark metadata.

Source-target overlap. Overlap is measured as the fraction of target-view pixels whose corresponding visible 3D points are also visible in the source view. When ground-truth depth is available, we compute overlap by back-projecting target pixels into 3D and re-projecting them into the source camera. A projected point is counted as overlapping if it lies inside the source image domain and passes a visibility check. For panoramic data, overlap is computed with circular horizontal wrap-around. For fisheye data, the native fisheye valid mask is applied before counting pixels. We use dataset-provided depth, camera poses, or meshes for this filtering.

A generic overlap computation can be written as:

$$\mathrm{Overlap}(s, t) = \frac{1}{|\Omega_t|} \sum_{\mathbf{p} \in \Omega_t} \mathbf{1}\left[\Pi_s\left(\Pi_t^{-1}(\mathbf{p}, D_t(\mathbf{p}))\right) \in \Omega_s\right], \tag{10}$$

where Ω_s and Ω_t denote source and target image domains, D_t is the target depth map, and Π_s, Π_t are the corresponding camera projection functions.

D.4 Camera Metadata and Projection Models

Each benchmark sample is associated with camera metadata. We store camera parameters in a unified format while preserving the native projection model of each dataset. Extrinsics are represented as camera-to-world transforms after converting dataset-specific coordinate conventions to the common training convention used by UniSHARP.

Perspective cameras. Perspective samples use pinhole intrinsics:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{11}$$

Wide-FoV cameras. OmniRooms-Wide uses an equidistant wide-FoV projection:

$$r = f\theta, \tag{12}$$

where θ is the angle between the optical axis and the viewing ray, and r is the radial distance from the image center. For the 130° rendered views, the metadata records the FoV, image size, valid radius, and per-frame camera transform.

Fisheye cameras. For fisheye datasets, we preserve the native fisheye calibration provided by the dataset whenever available. The benchmark stores the corresponding fisheye projection parameters and converts them to a renderer-compatible representation during evaluation. For simulated fisheye views, we use an equidistant fisheye projection with a 130° FoV, 1024×1024 resolution, and a valid radius of 512 pixels.

Panoramic cameras. Panoramic samples use equirectangular projection. For a pixel coordinate (u, v) in an image of width W and height H, longitude and latitude are computed as:

$$\phi = 2\pi \left(\frac{u + 0.5}{W} \right) - \pi, \quad \theta = \frac{\pi}{2} - \pi \left(\frac{v + 0.5}{H} \right). \tag{13}$$

The corresponding unit ray is:

$$\mathbf{r} = \begin{bmatrix} \cos\theta \sin\phi \\ \sin\theta \\ \cos\theta \cos\phi \end{bmatrix}. \tag{14}$$

D.5 Evaluation Protocol

The benchmark follows a single-source multi-target protocol. For each sequence, the first frame in a predefined evaluation group is used as the source view, and target views are selected from subsequent valid frames whose frame-index gap is at most 10. The model receives only the source image and, in the calibrated setting, the source and target camera parameters. No target-view RGB information is used during inference.

For each valid target view, the method renders an image in the target camera projection. The rendered image is compared with the ground-truth target image using PSNR, SSIM, and LPIPS. Metrics are first averaged over all valid target views of a source sequence, then averaged over sequences within each dataset. We report dataset-level scores in the main paper.

Resolution. All methods are evaluated at the ground-truth target resolution used by the corresponding validation sample, unless a fixed validation resizing multiple is specified. In that case, both predictions and targets are resized consistently before metric computation. OmniRooms equirectangular panoramas use 1024×2048 resolution. Perspective datasets are evaluated at the image resolution produced by the corresponding dataset loader after the same deterministic resizing rule.

Valid masks. If a dataset contains invalid pixels, missing depth, black borders, or undefined fisheye regions, metrics are computed only on valid pixels. Fisheye views use the valid fisheye mask, and wide-FoV equidistant views use the valid rendered image domain recorded by the camera metadata. For no-extrapolation rendering modes, border-connected black invalid regions are excluded from metric computation.

Panoramic boundary handling. For equirectangular images, horizontal coordinates are circular. During rendering and any geometric filtering, longitude wrap-around is handled circularly. For image-quality metrics, predictions and ground-truth images are compared in the same equirectangular coordinate system.

D.6 Metric Implementation

We use PSNR, SSIM, and LPIPS as the benchmark metrics. PSNR measures pixel-level reconstruction fidelity, SSIM measures structural similarity, and LPIPS measures perceptual similarity.

PSNR. PSNR is computed from the mean squared error over valid pixels:

$$\mathrm{PSNR} = -10 \log_{10}(\mathrm{MSE}), \tag{15}$$

assuming RGB values are normalized to $[0, 1]$.

SSIM. SSIM is computed on RGB images using a Gaussian window of size 11 and standard deviation 1.5. We use constants $C_1 = 0.01^2$ and $C_2 = 0.03^2$ on normalized RGB values. Images are not converted to luminance; the RGB-channel SSIM values are averaged.

LPIPS. LPIPS is computed using the official `lpips` implementation with the AlexNet backbone, i.e., `LPIPS(net="alex")`. All benchmark numbers reported in the paper use this LPIPS-Alex setting.

Aggregation. For each dataset, metrics are averaged across target views and then across source sequences. Camera-group averages are computed as unweighted averages over dataset-level scores in the same camera group, so large validation sets do not dominate the group score.

D.7 Baseline Evaluation Details

We evaluate all baselines on the same source-target pairs as UniSHARP. For each baseline, we use official checkpoints and inference code when available. If a method supports only a subset of camera models, we use the closest compatible input representation and keep the target-view evaluation protocol unchanged.

Perspective baselines. SHARP, Flash3D, LVSM, and TMPI are evaluated on perspective datasets. These methods are designed primarily for perspective monocular view synthesis, so we use the original perspective camera parameters and render target views under the corresponding pinhole cameras. Inputs are resized using the official preprocessing of each method, and outputs are resized back to the target resolution before metric computation.

Wide-FoV, fisheye, and panoramic baselines. PanoDreamer and Matrix3D are evaluated on non-perspective datasets because they support broader view synthesis or panoramic generation settings. For each method, we convert the benchmark input into the representation expected by the official implementation and then render or sample the corresponding target views. Camera trajectories are always taken from the benchmark metadata. We use the official default number of diffusion steps, optimization iterations, and post-processing settings unless the method cannot produce the required projection directly, in which case the output is first rendered in the method's native projection and then reprojected to the benchmark target camera.

Runtime measurement. For runtime comparisons, all methods are evaluated under the same hardware setting and using the same number of rendered target views. Timing excludes dataset loading and image decoding, includes model forward and target-view rendering, and is measured after warm-up runs. Batch size, image resolution, GPU type, and the number of warm-up and measured iterations are fixed across methods.

D.8 Pose-Free Evaluation Details

The pose-free setting evaluates whether a method can operate without manually provided camera intrinsics. In this setting, the model receives only the source RGB image. UniSHARP predicts a ray field, infers the camera model, and recovers rendering geometry before synthesizing target views.

We evaluate pose-free rendering on WildRGB-D. The calibrated setting uses the available camera parameters, while the pose-free setting replaces the source camera calibration with the recovered camera geometry. Target camera parameters are still provided to define the evaluation views and

to make the rendered images comparable with ground-truth target frames. Thus the pose-free setting removes source-camera calibration from the model input, but does not change the target-view definitions used for metric computation.

D.9 Quality Control

We apply the filtering criteria introduced in Sec. 4.1 and further remove samples with missing source or target images, invalid camera metadata, or missing/invalid depth when depth is required for overlap computation. Fisheye samples are evaluated only inside the valid fisheye mask. In no-extrapolation modes, connected black border regions introduced by rendering are excluded from the valid metric mask.

For simulated OmniRooms data, we additionally remove samples with invalid depth buffers or severe clipping. The camera-position expansion step keeps only centers with $60 \leq Z \leq 180$ cm. Because each generated group starts from a retained center and clips the generated Z coordinate to the same valid range, all released local camera positions remain within the configured height interval.

D.10 Licenses, Ethics, and Privacy

The benchmark combines existing public datasets with the newly constructed OmniRooms and OmniRooms-Wide data. For existing datasets, users should follow the licenses and usage terms of the original datasets. OmniRooms and OmniRooms-Wide are released under the CC BY-NC 4.0 license for research and non-commercial use. The released metadata will not include private user information.