S2CamP: Smart View-Frustum Sampling for Camera-Parameter Preconditioning ⋆

Jiachen Zhou *
Yifei Zhang * **Harald Kruggel-Emden** ** **Uwe D. Hanebeck** *

* *Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany (e-mail: jiachen.zhou@kit.edu; yifei.zhang@student.kit.edu; uwe.hanebeck@kit.edu).*
** *Chair of Mechanical Process Engineering and Solids Processing, Technische Universität Berlin, 10587 Berlin, Germany (e-mail: kruggel-emden@tu-berlin.de)*

Abstract: This paper is concerned with the joint optimization of camera parameters and Neural Radiance Fields (NeRFs). We analyze the Jacobian of the camera projection function using proxy 3D points sampled deterministically within a normalized camera view frustum. A Frolov–Fibonacci lattice combined with spherical inverse transforms yields low-dispersion samples and a sensitivity Gram matrix. From this matrix, we derive a linear reparameterization of the camera parameters that serves as a preconditioning transform, approximately whitening their sensitivities and reducing cross-parameter couplings. On a real-world benchmark, the proposed method improves camera-parameter accuracy and rendering quality while using only 10% of the samples required by a random-sampling baseline.

Keywords: Neural radiance fields, camera parameter estimation, robot perception and sensing, deterministic sampling, preconditioning in optimization.

1. INTRODUCTION

Neural Radiance Fields (NeRFs) (Mildenhall et al. (2021)) have emerged as a powerful paradigm for high-fidelity 3D reconstruction and novel-view synthesis from multi-view image collections, enabling photorealistic rendering of complex objects and large-scale environments. They have been successfully applied to tasks such as virtual and augmented reality rendering (Deng et al. (2022)) and the reconstruction of outdoor scenes for robotics and autonomous systems (Rudnev et al. (2022)). A particularly relevant application domain is advanced particle measurement, where irregular particulate solids must be reconstructed with high geometric fidelity to capture fine surface details and to derive accurate size and shape descriptors. Such morphological information strongly affects the mechanical and chemical behavior of particulate solids in downstream processes (Jia et al. (2024)).

The quality of NeRF-based reconstruction depends critically on the accuracy of the camera parameters governing the projection from 3D world points to 2D image pixels. In practical settings, however, camera poses and intrinsics obtained from Structure from Motion (SfM)-based initialization or external calibration are often noisy. Jointly optimizing the camera parameters together with the radiance field is therefore a principled and essential strategy to address this issue.

State of the Art A broad range of methods jointly refine camera parameters and the NeRF scene representation by exploiting NeRF's differentiable rendering, which enables both to be optimized within a unified framework. However, such joint optimization is often prone to local minima because errors in camera poses and the radiance field are strongly coupled. BARF (Lin et al. (2021)) addresses this issue with a coarse-to-fine schedule that initially deactivates high-frequency positional encodings, smoothing pose gradients and reducing early convergence to high-frequency local minima. GARF (Chng et al. (2022)) follows a similar principle but removes positional encoding entirely, instead relying on Gaussian activations to provide more stable convergence. GNeRF (Meng et al. (2021)) explores a different strategy by combining NeRF with Generative Adversarial Networks (GANs). A coarse radiance field and approximate camera poses are first optimized adversarially and subsequently refined. In contrast, SCNeRF (Jeong et al. (2021)) is among the earliest frameworks for full camera-parameter refinement, jointly estimating both intrinsic and extrinsic parameters via a dedicated self-calibration module.

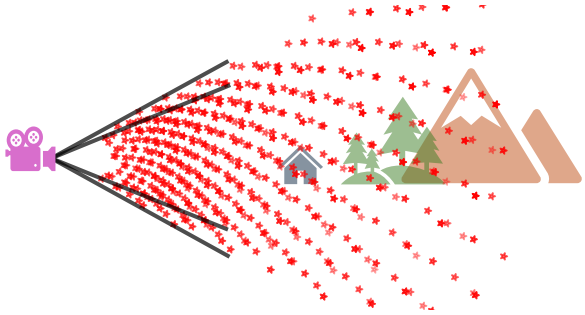


Fig. 1. The proposed deterministic samples (red stars) are shown in the (unnormalized) camera field of view.

⋆ The IGF project 01IF22901N of the research association Forschungs-Gesellschaft Verfahrenstechnik e.V. (GVT) is supported in a program to promote the Industrial Collective Research and Development (IGF) by the Federal Ministry of Economic Affairs and Climate Action on the basis of a decision of the German Bundestag.

A closely related method is CamP Park et al. (2023), which is among the first to explicitly analyze how parameter coupling and unequal sensitivities influence the camera projection function. It shows that even small perturbations in certain camera parameters can induce large changes in image coordinates, leading to ill-conditioned optimization.

Contributions In this work, we consider learning NeRF representations for high-fidelity novel-view synthesis of objects and large-scale environments from multi-view imagery. In particular, we focus on the joint optimization of the full set of initially inaccurate camera parameters, including both intrinsics and extrinsics.

Building on the Jacobian-based sensitivity analysis of the camera projection function proposed in Park et al. (2023), we show that the choice of 3D points used to evaluate the projection Jacobian is critical for improving the conditioning of the joint optimization. To address this issue, we introduce a deterministic sampling framework that constructs a low-dispersion proxy point set within a normalized camera view frustum. Compared with random sampling, the proposed scheme provides more homogeneous coverage with substantially fewer points, yields reproducible sensitivity estimates, and applies to a broad class of normalized frusta with separable polar or spherical parameterizations.

We further evaluate the proposed framework on a real-world benchmark with synthetically perturbed initial camera parameters. The results show that, compared with random-sampling baselines, the proposed method achieves lower volumetric dispersion and, when integrated into a NeRF training pipeline, improves both rendering quality and camera-parameter accuracy while using only 10% of the samples required by the baseline.

2. PROBLEM FORMULATION

Models NeRF represents, for each camera, a light ray as $\underline{r}(t) = \underline{o} + t\,\underline{d}$, which originates from the camera center $\underline{o}$ and extends in direction $\underline{d}$ into the scene, where t denotes the distance along the ray. A radiance field $f_\Theta\colon (\underline{x},\, \underline{d}) \mapsto (\tau,\, \underline{c})$, parameterized by the scene representation parameters $\underline{\Theta}$, predicts for any spatial location $\underline{x}$ and viewing direction $\underline{d}$ the corresponding volume density τ and emitted color $\underline{c}$. A universal photometric squared error loss is formulated by comparing the rendered color $\hat{\underline{c}}(\underline{r})$ along each ray with its ground-truth color $\underline{c}^{\mathrm{GT}}(\underline{r})$

$$\mathcal{L} = \sum\nolimits_{\underline{r}\in\mathcal{R}} \left\| \hat{\underline{c}}(\underline{r}) - \underline{c}^{\mathrm{GT}}(\underline{r}) \right\|_2^2 , \tag{1}$$

where $\mathcal{R}$ is the set of rays across all training images. Minimizing this loss iteratively updates the scene parameters $\underline{\Theta}$.

Image Formation and Camera Parameterization In general, camera parameters define how a camera maps the 3D world onto a 2D image. Various parameterizations can be employed to model this image formation. Section 4 details the parameterization used in this work. A common mapping method from a 3D world point ${}^{W}\underline{p}$ to its 2D pixel coordinates ${}^{I}\underline{u}$ is described by a projection function $\underline{\Pi}_\Phi(\cdot)\colon \mathbb{R}^3 \to \mathbb{R}^2$ parameterized by $\underline{\Phi}$

$$\begin{aligned} (\tilde{u},\, \tilde{v},\, \tilde{w})^\top &= \mathbf{K}_{3\times3}\,[\,\mathbf{R} \mid \underline{t}\,]_{3\times4}\, \left({}^{W}\underline{p},\, 1\right)_{4\times1}^\top , \\ {}^{I}\underline{u} &= \underline{\Pi}_\Phi\left({}^{W}\underline{p}\right) = (\tilde{u}/\tilde{w}\,,\ \tilde{v}/\tilde{w})^\top . \end{aligned} \tag{2}$$

Here, $(\tilde{u},\, \tilde{v},\, \tilde{w})^\top$ denotes the homogeneous image coordinates of the projection of the world point ${}^{W}\underline{p}$, where the tildes indicate quantities expressed in homogeneous coordinates and the last component $\tilde{w} \neq 0$ serves as the projective scale factor. The projection function $\underline{\Pi}_\Phi(\cdot)$ maps a world point ${}^{W}\underline{p}$ to its Euclidean pixel coordinates ${}^{I}\underline{u}$ by normalizing the first two components with $\tilde{w}$, i.e., ${}^{I}\underline{u} = (\tilde{u}/\tilde{w}\,,\ \tilde{v}/\tilde{w})^\top$. $\underline{\Phi}$ comprises both the extrinsic and intrinsic parameters. Extrinsics describe the camera's orientation $\mathbf{R} \in \mathbb{R}^{3\times3}$ and position $\underline{t} \in \mathbb{R}^3$ in the world frame. Compared to extrinsics, intrinsics $\mathbf{K} \in \mathbb{R}^{3\times3}$ are unique and inherent to a given camera. They include focal lengths, principal point offsets, and skew or distortion parameters.

Joint Optimization Challenges In practice, the camera parameters estimated by SfM or external calibration are often noisy, leading to suboptimal photometric supervision. To enhance camera parameter quality, many works adopt joint optimization frameworks (Lin et al. (2021); Bian et al. (2023)), where the photometric loss (1) is backpropagated to both the scene representation $\underline{\Theta}$ and the camera parameters $\underline{\Phi}$. We follow this paradigm and treat $\underline{\Phi}$ as trainable, updating it jointly with the radiance field via (stochastic) gradient descent with learning rate η:

$$\underline{\Phi}_{k+1} = \underline{\Phi}_k + \Delta\underline{\Phi}\,,\ \Delta\underline{\Phi} = -\eta\,\nabla_{\underline{\Phi}}\mathcal{L}\,,\ k = 0,\, 1,\, 2,\, \ldots, \tag{3}$$

where $\nabla_{\underline{\Phi}}\mathcal{L}$ is the loss gradient with respect to the camera parameters. Alternative update strategies are also possible. In Bian et al. (2024), a Multi-Layer Perceptron (MLP) is learned, whose output directly predicts the residual update $\Delta\underline{\Phi}$ between successive camera estimates.

However, jointly optimizing camera parameters remains challenging because they differ in both physical units and numerical scales. A perturbation of the same magnitude applied to each parameter can produce vastly different effects on the 2D projection function (Park et al., 2023, p. 4). For instance, a one-meter translation of the camera center can shift the projected pixel locations by hundreds of pixels, whereas a one-pixel change in focal length affects them only marginally. As these parameters exhibit highly different sensitivities to the projection function, the corresponding gradients differ significantly in magnitude during backpropagation, resulting in an ill-conditioned optimization problem. Thus, training may become unstable or converge slowly, which even advanced optimizers, e.g. Adam (Kingma et al. (2015)), can only partially mitigate.

Quantitative Sensitivity Analysis A common way to characterize the sensitivity of a function to its parameters is to analyze its Jacobian matrix. Given a set of 3D points $\mathcal{P} = \{\underline{p}_i\}_{i=1}^{L}$, we define a stacked projection function $\underline{\Pi}_{\mathcal{P}}(\underline{\Phi})\colon \mathbb{R}^K \to \mathbb{R}^{2L}$, which concatenates the 2D projections of all L points under the K camera parameters $\underline{\Phi} = [\phi_1,\, \ldots,\, \phi_K]^\top$. The Jacobian

$$\mathbf{J}_\Pi(\underline{\Phi};\, \mathcal{P}) = \frac{\partial\,\underline{\Pi}_{\mathcal{P}}(\underline{\Phi})}{\partial\,\underline{\Phi}} \in \mathbb{R}^{2L\times K} \tag{4}$$

quantifies how perturbations in each parameter ϕ_i affect the stacked 2D projections. From this Jacobian, we construct the sensitivity Gram matrix (Lanckriet et al. (2004)) as

$$\mathbf{\Sigma}_\Pi \coloneqq \mathbf{J}_\Pi^\top \mathbf{J}_\Pi \in \mathbb{R}^{K\times K} . \tag{5}$$

The diagonal entries of $\mathbf{\Sigma}_\Pi$ capture the total squared sensitivity of the stacked projections with respect to each parameter, whereas the off-diagonal terms quantify pairwise parameter couplings through the inner products of the corresponding sensitivity vectors. Hence, $\mathbf{\Sigma}_\Pi$ serves as an information matrix that characterizes the joint influence of the camera parameters

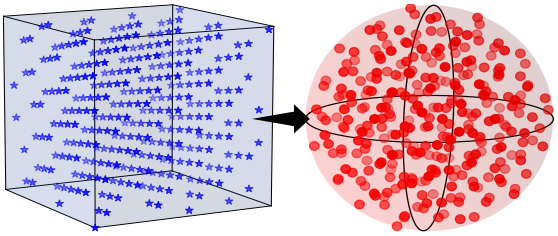


Fig. 2. Illustration of the deterministic low-discrepancy Frolov–Fibonacci lattice in $[0,1]^3$ (left) and its orthogonal inverse transformation onto the unit sphere (right), yielding a uniform coverage of the spherical volume.

on the projection function. It also provides a practical indicator of the conditioning of the joint optimization problem.

Considered Problem The effectiveness of the above sensitivity analysis depends strongly on the choice of the 3D proxy points $\mathcal{P}$ used to evaluate $\mathbf{J}_\Pi$ and the resulting Gram matrix $\mathbf{\Sigma}_\Pi$. In Park et al. (2023), these points are sampled randomly within the entire view frustum. However, random sampling provides no guarantees of uniform angular or depth coverage, often over-represents distant regions, and may lead to spatial clustering. Consequently, the estimated parameter sensitivities and couplings can be unstable, non-reproducible, and reliant on a large number of samples.

To address these limitations, we propose a deterministic sampling strategy that distributes a smaller number of proxy points across the view frustum more evenly. The resulting compact and structured point set enables stable and reproducible estimation of $\mathbf{\Sigma}_\Pi$. Based on this Jacobian-induced metric, we perform targeted parameter updates that balance gradient magnitudes and reduce undesirable parameter couplings, thereby improving the conditioning and stability of the joint optimization process.

3. KEY IDEA AND GROUNDWORK

We briefly review the mathematical preliminaries and notation relevant to this work, following Frisch et al. (2025). For a separable Probability Density Function (PDF) f, a separable inverse transform Q converts a set of uniform low-discrepancy points $\mathcal{P}_L^u$ to deterministic low-dispersion samples $\mathcal{P}_L^f = Q(\mathcal{P}_L^u)$ following f, thereby enabling efficient deterministic sampling in spherical-coordinate domains. In the three-dimensional setting, we generate the point set $\mathcal{P}_L^u$ by means of a Frolov-Fibonacci lattice, a widely used deterministic sampling scheme for three-dimensional applications Frisch et al. (2021, 2023). Its construction in the unit cube for a prescribed number of sample points is described in (Frisch et al., 2025, Sec. III.B, p. 2). Fig. 2 (left) illustrates a deterministic low-discrepancy point set, consisting of 300 samples uniformly distributed in the unit cube $[0,1]^3$.

Orthogonal Inverse Transforms Given a PDF $f(\underline{x})$ on $\mathbb{R}^3$ that is separable in spherical coordinates, i.e., $f(r,\,\theta,\,\varphi) = f_r(r) f_\theta(\theta) f_\varphi(\varphi)$, $r \in [0,1]$, $\theta \in [0,\pi]$, $\varphi \in [0,2\pi)$, with the corresponding volume element $\mathrm{d}V = r^2 \sin(\theta)\,\mathrm{d}r\,\mathrm{d}\theta\,\mathrm{d}\varphi$, the Cumulative Distribution Functions (CDFs) can be expressed as

$$
\begin{aligned}
F(r,\,\theta,\,\varphi) &= \iiint f(r,\,\theta,\,\varphi)\,\mathrm{d}V = F_r(r)\,F_\theta(\theta)\,F_\varphi(\varphi),\\
F_r(r) &= \int_{r'=0}^{r} f_r(r')\,(r')^2\,\mathrm{d}r',\\
F_\theta(\theta) &= \int_{\theta'=0}^{\theta} f_\theta(\theta')\,\sin(\theta')\,\mathrm{d}\theta',\\
F_\varphi(\varphi) &= \int_{\varphi'=0}^{\varphi} f_\varphi(\varphi')\,\mathrm{d}\varphi'.
\end{aligned}
\tag{6}
$$

For notational clarity, let $\underline{u} = (u_1,\,u_2,\,u_3)^\top$ denote the vector of normalized inputs within $[0,1]^3$. The corresponding 1D inverse transform functions are then defined as

$$
\begin{aligned}
Q_r(u_1) &= F_r^{-1}(u_1)\,,\\
Q_\theta(u_2) &= F_\theta^{-1}(u_2)\,,\\
Q_\varphi(u_3) &= F_\varphi^{-1}(u_3)\,.
\end{aligned}
\tag{7}
$$

Applying the component-wise inverse transforms to a deterministic low-discrepancy input point set produces samples distributed according to f (expressed in spherical coordinates). This mapping proceeds as follows. Let $\mathcal{P}_L^u = \{\underline{p}_i^u = (p_{i,1}^u,\,p_{i,2}^u,\,p_{i,3}^u)\}_{i=1}^L \subset [0,1]^3$ denote the deterministic low–discrepancy uniform samples, and let the resulting transformed samples be $\mathcal{P}_L^f = \{\underline{p}_i^f = (p_{i,1}^f,\,p_{i,2}^f,\,p_{i,3}^f)\}_{i=1}^L$, which are distributed according to f. For each i, we first apply the component-wise inverse transforms

$$
[r_i,\,\theta_i,\,\varphi_i] \;=\; \left[Q_r(p_{i,1}^u),\;Q_\theta(p_{i,2}^u),\;Q_\varphi(p_{i,3}^u)\right]\,, \tag{8}
$$

and then convert to Cartesian coordinates

$$
\begin{aligned}
\underline{p}_i^{\,f} &= (p_{i,1}^f,\,p_{i,2}^f,\,p_{i,3}^f)\\
&= \big(r_i \sin(\theta_i)\cos(\varphi_i),\;r_i \sin(\theta_i)\sin(\varphi_i),\;r_i \cos(\theta_i)\big)\,.
\end{aligned}
\tag{9}
$$

Deterministic Uniform Sampling within the Unit Ball As a practical example of the transformation framework introduced above, consider the task of generating deterministic samples that are uniformly distributed within a unit solid sphere $\mathbb{B}^3 = \{\underline{x} \in \mathbb{R}^3 \mid \|\underline{x}\|_2 \le 1\}$. Uniformity is understood here with respect to the fact that each volume element $\mathrm{d}V$ is assigned an equal probability mass. In spherical coordinates $(r,\,\theta,\,\varphi)$, the joint density $f(r,\,\theta,\,\varphi)$ associated with a uniform distribution over $\mathbb{B}^3$ is proportional to $r^2 \sin(\theta)$. Fig. 2 (right) shows the transformed Cartesian samples, uniformly covering $\mathbb{B}^3$.

4. METHOD DERIVATION

In this section, we build upon the deterministic sampling framework and adapt it to a specialized camera view frustum, thereby enabling its application to large-scale and unbounded scenes. We also introduce a camera reparameterization model and derive a preconditioning matrix that decouples and normalizes camera parameters, improving the conditioning of the joint NeRF–camera optimization problem.

Normalized Camera View Frustum A fundamental concept in 3D scene representation is the camera view frustum, which defines the region visible to the camera and projected onto the image plane. In large-scale or unbounded scenes, however, a Euclidean frustum is inefficient for NeRF-based representations. Due to perspective projection, distant regions occupy only small image areas while corresponding to large spatial volumes. As a result, NeRF must sample many points along each ray to represent these regions adequately, increasing computational cost and training time.

To address this issue, Barron et al. (2022) introduced a normalized camera view frustum that places points proportionally to disparity, i.e., the inverse of distance. Each Euclidean point $\underline{x} \in \mathbb{R}^3$, defined relative to the camera's optical center as the origin, is mapped to a compact domain via the following transformation

$$\text{normalization}(\underline{x}) = \begin{cases} \underline{x}\,, & \|\underline{x}\|_2 \le 1\,, \\ \left(2 - \frac{1}{\|\underline{x}\|_2}\right) \frac{\underline{x}}{\|\underline{x}\|_2}\,, & \|\underline{x}\|_2 > 1\,. \end{cases} \tag{10}$$

This mapping is the identity for $\|\underline{x}\|_2 \le 1$, thereby preserving the local geometry near the camera, and smoothly compresses all points with $\|\underline{x}\|_2 > 1$ into a bounded ball. Thus, the originally unbounded Euclidean scene is mapped to a finite-radius normalized space while preserving the viewing directions $\underline{x}/\|\underline{x}\|_2$. This mapping is bijective and admits a closed-form inverse, so every normalized point can be mapped back to its unbounded original Euclidean distance, as shown in Fig. 1.

Deterministic View-Frustum Sampling We aim to generate deterministic low-dispersion samples that uniformly and homogeneously cover the entire normalized view frustum of interest. These samples can then serve as an effective and representative proxy for the scene content in the real unnormalized field of view. We begin with the general case, leaving the specific parameter bounds unspecified for now. The 1D marginal CDFs under the spherical coordinate volume measure are

$$\begin{aligned} F_r(r) &= \frac{\int_{r_{\min}}^{r} (r')^2 \,\mathrm{d}r'}{\int_{r_{\min}}^{r_{\max}} (r')^2 \,\mathrm{d}r'} = \frac{r^3 - r_{\min}^3}{r_{\max}^3 - r_{\min}^3}\,, \\ F_\theta(\theta) &= \frac{\int_{\theta_{\min}}^{\theta} \sin(\theta') \,\mathrm{d}\theta'}{\int_{\theta_{\min}}^{\theta_{\max}} \sin(\theta') \,\mathrm{d}\theta'} = \frac{\cos(\theta_{\min}) - \cos(\theta)}{\cos(\theta_{\min}) - \cos(\theta_{\max})}\,, \\ F_\varphi(\varphi) &= \frac{\int_{\varphi_{\min}}^{\varphi} \mathrm{d}\varphi'}{\int_{\varphi_{\min}}^{\varphi_{\max}} \mathrm{d}\varphi'} = \frac{\varphi - \varphi_{\min}}{\varphi_{\max} - \varphi_{\min}}\,. \end{aligned} \tag{11}$$

According to (7), the inverse transforms are calculated as

$$\begin{aligned} Q_r(u_1) &= \left[r_{\min}^3 + u_1(r_{\max}^3 - r_{\min}^3)\right]^{1/3}, \\ Q_\theta(u_2) &= \arccos\left[\cos(\theta_{\min}) - u_2\big(\cos(\theta_{\min}) - \cos(\theta_{\max})\big)\right], \\ Q_\varphi(u_3) &= \varphi_{\min} + u_3(\varphi_{\max} - \varphi_{\min}). \end{aligned} \tag{12}$$

The normalized camera view frustum is parameterized by the image resolution (H, W), the focal length f, and the near and far spherical radii t_{near} and t_{far}. It is bounded by the rays through the four image corners and their intersections with the two spherical surfaces, with radial extent $r \in [t_{\text{near}}, t_{\text{far}}]$ and angular extent determined by f and (H, W). Assuming that the optical axis passes through the image center, the horizontal and vertical half-angles of view are

$$\alpha_x = \arctan(W/2f)\,, \quad \alpha_y = \arctan(H/2f)\,. \tag{13}$$

These half-angles define the angular limits of the frustum in spherical coordinates as

$$\begin{aligned} \theta_{\min} &= -\alpha_y, & \theta_{\max} &= \alpha_y, \\ \varphi_{\min} &= -\alpha_x, & \varphi_{\max} &= \alpha_x. \end{aligned} \tag{14}$$

Adjusting the interval $[t_{\text{near}}, t_{\text{far}}]$ directly controls the spatial extent of the reconstructed scene. For instance, a point at a Euclidean distance of 10 m from the camera maps to a normalized radius of 1.9, while choosing $t_{\text{far}} = 1.99$ covers distances of roughly up to 100 m.

We map a locally stored deterministic low-discrepancy Frolov–Fibonacci lattice through the inverse transform in (12),

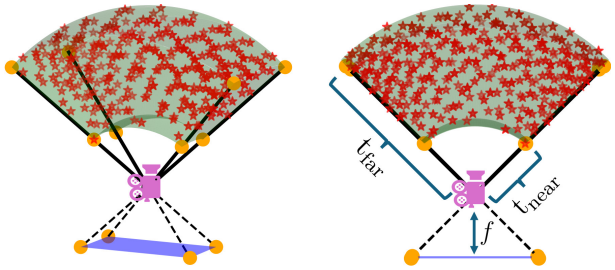


Fig. 3. Example illustration of the normalized camera view frustum (green) and the deterministic samples (red).

yielding low-dispersion samples that uniformly cover the normalized camera view frustum. An example of the resulting frustum and its deterministic samples is shown in Fig. 3.

Camera Parameterization As introduced in Section 2, the camera parameters $\underline{\Phi}$ define the projection from 3D points to 2D pixels. We represent the extrinsics in the $\mathfrak{se}(3)$ screw-axis form by a 6D pose vector, following Lin et al. (2021). This $\mathfrak{se}(3)$ parameterization enables smooth, physically interpretable pose updates during optimization (Gallo et al. (2022)). The intrinsics consist of the focal length, principal point offsets, skew, and two radial distortion coefficients. A residual-based scheme is used for parameter refinement, with updated parameters $\underline{\Phi}'$ obtained from $\underline{\Phi}$ and $\Delta\underline{\Phi}$, i.e., $\underline{\Phi}' = \underline{\Phi} + \Delta\underline{\Phi}$.

Preconditioning Building on the above formulation, we use uniform deterministic sampling within the normalized camera view frustum to construct a structured proxy point set $\mathcal{P}^*$. On this basis, we introduce a preconditioning scheme for the camera parameters that normalizes per-parameter sensitivities of the projection function and mitigates cross-parameter correlations. Given the projection $\underline{\Pi}_\Phi(\cdot)$ and its Jacobian $\mathbf{J}_\Pi(\underline{\Phi}; \mathcal{P}^*)$ using $\mathcal{P}^* = \{\underline{p}_\ell^*\}_{\ell=1}^L$, we construct the covariance-like sensitivity matrix $\mathbf{\Sigma}_\Pi$ in (5). Large off-diagonal values indicate strong correlations and hence poor conditioning.

The key idea is to linearly reparameterize the camera parameters $\underline{\Phi}$ so that the resulting Gram matrix is (approximately) whitened, i.e., $\mathbf{\Sigma}_{\hat{\Pi}} \coloneqq \mathbf{J}_{\hat{\Pi}}^\top \mathbf{J}_{\hat{\Pi}} \overset{!}{=} \mathbf{I}$, thereby decorrelating and normalizing all parameter sensitivities. To this end, we introduce a preconditioning matrix $\mathbf{P}$ and define the transformed parameter vector as $\hat{\underline{\Phi}} = \mathbf{P}\,\underline{\Phi}$. We then reparameterize the projection function by replacing $\underline{\Phi}$ with $\mathbf{P}^{-1}\hat{\underline{\Phi}}$, yielding $\hat{\underline{\Pi}}(\cdot) \coloneqq \underline{\Pi}_{\mathbf{P}^{-1}\hat{\underline{\Phi}}}(\cdot)$ in (2). According to (4), its Jacobian is

$$\begin{aligned} \mathbf{J}_{\hat{\Pi}}(\hat{\underline{\Phi}};\, \mathcal{P}^*) &= \frac{\mathrm{d}\,\hat{\underline{\Pi}}_{\mathcal{P}^*}(\underline{\Phi})}{\mathrm{d}\,\hat{\underline{\Phi}}} = \frac{\mathrm{d}\,\underline{\Pi}_{\mathcal{P}^*}(\mathbf{P}^{-1}\hat{\underline{\Phi}})}{\mathrm{d}\,\hat{\underline{\Phi}}} \\ &= \frac{\mathrm{d}\,\underline{\Pi}_{\mathcal{P}^*}(\mathbf{P}^{-1}\hat{\underline{\Phi}})}{\mathrm{d}\,\underline{\Phi}} \frac{\mathrm{d}\,\underline{\Phi}}{\mathrm{d}\,\hat{\underline{\Phi}}} = \mathbf{J}_\Pi\,\mathbf{P}^{-1}\,. \end{aligned} \tag{15}$$

Its Gram matrix is

$$\mathbf{\Sigma}_{\hat{\Pi}} \coloneqq \mathbf{J}_{\hat{\Pi}}^\top \mathbf{J}_{\hat{\Pi}} = \mathbf{P}^{-T}\,\mathbf{J}_\Pi^\top \mathbf{J}_\Pi\,\mathbf{P}^{-1} \overset{!}{=} \mathbf{I}\,. \tag{16}$$

The corresponding whitening condition can be rewritten as

$$\begin{aligned} \mathbf{P}^{-T}\,\mathbf{J}_\Pi^\top \mathbf{J}_\Pi\,\mathbf{P}^{-1}\,\mathbf{P}^{-T} &= \mathbf{P}^{-T}\,, \\ \mathbf{J}_\Pi^\top \mathbf{J}_\Pi\,\mathbf{P}^{-1}\,\mathbf{P}^{-T} &= \mathbf{I}\,, \\ \mathbf{P}^T\,\mathbf{P} &= \mathbf{J}_\Pi^\top \mathbf{J}_\Pi = \mathbf{\Sigma}_\Pi\,. \end{aligned} \tag{17}$$

For any real, invertible matrix $\mathbf{P}$ we can apply the polar decomposition and write $\mathbf{P} = \mathbf{Q}\,\mathbf{S}$, where $\mathbf{Q}$ is an orthogonal

matrix with $\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}$ and $\mathbf{S}$ is a symmetric positive-definite matrix. Substituting $\mathbf{P} = \mathbf{Q}\,\mathbf{S}$ into (17) yields

$$(\mathbf{Q}\,\mathbf{S})^\top (\mathbf{Q}\,\mathbf{S}) = \mathbf{S}^\top \mathbf{Q}^\top \mathbf{Q}\,\mathbf{S} = \mathbf{S}^\top \mathbf{S} = \mathbf{S}^2 = \mathbf{\Sigma}_\Pi\,. \tag{18}$$

Hence $\mathbf{S}$ is the unique symmetric positive-definite matrix square root of $\mathbf{\Sigma}_\Pi$, denoted $\mathbf{S} = \mathbf{\Sigma}_\Pi^{1/2}$. Consequently, every admissible preconditioner $\mathbf{P}$ can be written as

$$\mathbf{P} = \mathbf{Q}\,\mathbf{S} = \mathbf{Q}\,\mathbf{\Sigma}_\Pi^{1/2}\,, \tag{19}$$

with $\mathbf{Q}$ an arbitrary orthogonal matrix. This implies that the whitening condition does not determine a unique $\mathbf{P}$, but in fact admits infinitely many solutions.

To obtain a unique and practically convenient preconditioner, we eliminate this rotational ambiguity by adopting the Mahalanobis whitening transform Bell et al. (1997) and setting $\mathbf{Q} = \mathbf{I}$. Among the whitening schemes reviewed in Kessy et al. (2018), this choice corresponds to the transform without any additional orthogonal rotation in parameter space. Thus, it introduces no extra mixing of camera parameters beyond what is strictly required for whitening. Moreover, in the sense of the expected squared Euclidean distance, the resulting preconditioned parameters remain as close as possible to the original ones Eldar et al. (2003). Hence, the desired preconditioner is

$$\mathbf{P} = \mathbf{Q}\,\mathbf{S} = \mathbf{\Sigma}_\Pi^{1/2} = \left[\mathbf{J}_\Pi^\top(\underline{\Phi};\,\mathcal{P}^*)\,\mathbf{J}_\Pi(\underline{\Phi};\,\mathcal{P}^*)\right]^{1/2}\,. \tag{20}$$

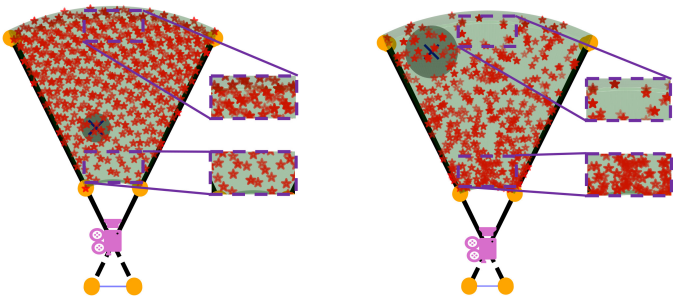
During joint optimization, all gradient updates are carried out in the whitened parameter space $\hat{\underline{\Phi}}$. For efficiency, the preconditioner $\mathbf{P}$ is refreshed periodically as the camera parameters evolve. At each iteration, the current camera parameters used for rendering are obtained by mapping back according to $\underline{\Phi} = \mathbf{P}^{-1}\hat{\underline{\Phi}}$.

5. EVALUATION

We evaluate our method on the mip-NeRF 360 dataset from Barron et al. (2022). The underlying architecture and training procedure follow the Zip-NeRF framework proposed in Barron et al. (2023). Training is performed for 50.000 iterations on 2 NVIDIA A100 GPUs. The initial camera parameters for joint optimization are obtained from an SfM module. To thoroughly evaluate the robustness of our approach with respect to camera pose errors, we follow the setup of prior work (see CamP Park et al. (2023)), and introduce synthetic perturbations to these SfM-estimated parameters before training.

Sampling Uniformity and Dispersion Analysis We begin by comparing and visualizing our deterministic sampling strategy and the random sampling method proposed in Park et al. (2023) within the same normalized camera view frustum. The objective is to assess whether the two strategies provide homogeneous coverage of the 3D scene, that is, whether all regions are sampled sufficiently uniformly. This property directly influences the subsequent computation of the preconditioning matrix. Fig. 4 visually compares the two sampling strategies, where the ground-truth camera parameters of the *room* scene are used to illustrate the normalized view frustum.

Furthermore, we quantitatively compare the two sampling strategies using the dispersion metric. Dispersion, originally defined for a point set within the d-dimensional unit cube $[0,1]^d$ (Hinrichs et al. (2020)), measures the volume of the largest axis-aligned empty region and serves as an effective indicator of spatial homogeneity. Using the camera intrinsics and image resolutions provided in the datasets, we compute the radius



(a) Deterministic sampling (b) Random sampling

Fig. 4. Comparison of (a) deterministic and (b) random sampling within the normalized camera view frustum, each using 500 samples (red stars).

Table 1. Dispersion radii across different sample counts, with the left and right entries corresponding to deterministic and random sampling, respectively. Lower dispersion values are underlined.

Type	Scenes	Number of Samples		
		100	**500**	**5000**
Indoor	Room	<u>0.19</u> / 0.32	<u>0.11</u> / 0.22	<u>0.05</u> / 0.12
	Bonsai	<u>0.19</u> / 0.27	<u>0.11</u> / 0.20	<u>0.05</u> / 0.11
	Counter	<u>0.19</u> / 0.25	<u>0.11</u> / 0.19	<u>0.05</u> / 0.11
	Kitchen	<u>0.19</u> / 0.26	<u>0.11</u> / 0.20	<u>0.05</u> / 0.09
Outdoor	Flowers	<u>0.22</u> / 0.31	<u>0.13</u> / 0.22	<u>0.06</u> / 0.12
	Bicycle	<u>0.21</u> / 0.38	<u>0.12</u> / 0.22	<u>0.06</u> / 0.11
	Garden	<u>0.23</u> / 0.33	<u>0.14</u> / 0.25	<u>0.05</u> / 0.09
	Stump	<u>0.21</u> / 0.28	<u>0.13</u> / 0.22	<u>0.06</u> / 0.11

Table 2. Time and space complexity of the two strategies for sampling L points.

Method	**Time Compl.**	**Space Compl.**
Random	$\mathcal{O}(L)$	$\mathcal{O}(L)$
Deterministic	$\mathcal{O}(L)$	$\mathcal{O}(L)$

of the largest empty sphere within the frustum that contains no sample points. The resulting dispersion radii are reported in Table 1. Across all scenes and sample sizes, our S2CamP consistently achieves more uniform and homogeneous coverage of the normalized frustum for a fixed number of samples. To complement the dispersion analysis, Table 2 summarizes the time and space complexity of the two sampling strategies with respect to the number of proxy points L. Both methods scale linearly in time and memory. In the proposed deterministic pipeline, the precomputed lattice and periodic preconditioner updates keep overhead low while improving spatial coverage for the same sample budget.

Rendering Quality Comparison We evaluate our method using both camera-parameter accuracy and novel-view synthesis quality. For extrinsics, we report rotation and translation errors relative to the ground-truth poses, measured as the angular difference in orientation (radians) and the Euclidean distance between camera centers (meters), respectively. For intrinsics, we report the absolute focal-length error in pixels. Rendering quality is assessed using three standard metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS). Higher PSNR and SSIM, and lower LPIPS, indicate better quality.

Table 3. Quantitative rendering quality comparison, with the left and right values corresponding to deterministic and random sampling, respectively. For comparison, the better score is underlined for each metric.

Type	Scenes	Image Quality Metrics		
		PSNR ↑	SSIM ↑	LPIPS ↓
Indoor	Room	29.09 / 28.38	0.86 / 0.84	0.23 / 0.24
	Bonsai	28.23 / 28.19	0.87 / 0.88	0.21 / 0.21
Outdoor	Garden	23.23 / 23.12	0.65 / 0.64	0.27 / 0.28
	Stump	23.10 / 23.35	0.63 / 0.63	0.34 / 0.31

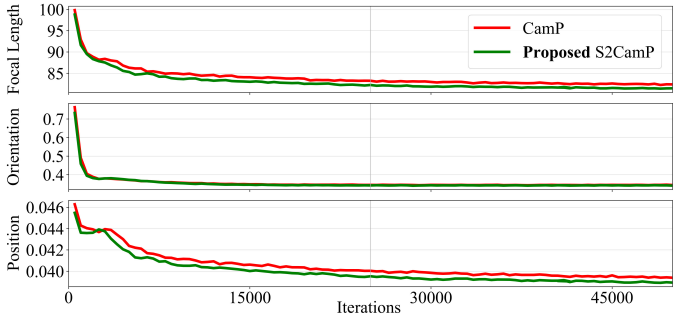


Fig. 5. Training evolution of focal-length, orientation, and position errors for the random-sampling-based CamP-baseline and the proposed deterministic method.

For comparison, we adopt CamP (Park et al. (2023)) as the primary baseline. Following its original setup, the projection Jacobian is evaluated using 5000 random samples within the normalized view frustum. To ensure a fair comparison, we repeat the random-sampling baseline five times with different random seeds and report the averaged results. In contrast, our S2CamP evaluates the same Jacobian using only 500 deterministic samples, i.e., only 10% of the baseline's budget. The quantitative rendering results are summarized in Table 3. Across all four scenes, S2CamP achieves performance comparable to the baseline. In addition, Fig. 5 shows the evolution of the focal-length, orientation, and position errors for the *room* scene during training. The red and green curves correspond to the CamP-baseline and the proposed S2CamP, respectively. Whereas both methods exhibit similar convergence in orientation error, S2CamP reaches lower focal-length and position errors earlier and attains better final values, indicating a more stable and data-efficient joint optimization of camera parameters.

6. CONCLUSION

We presented S2CamP, a deterministic sampling-based preconditioning framework for the joint optimization of camera parameters and NeRF representations. Using Frolov–Fibonacci lattices and separable inverse transforms in spherical coordinates, S2CamP constructs low-dispersion proxy points that uniformly cover the normalized camera view frustum, enabling stable and reproducible estimation of the projection Jacobian and its sensitivity Gram matrix. Based on this matrix, we derive a linear reparameterization that approximately whitens camera-parameter sensitivities and alleviates cross-parameter couplings. Experiments demonstrate that S2CamP improves or matches the rendering quality of a strong random-sampling baseline with only 10% of the sample budget, while also producing more accurate camera parameter estimates. Beyond NeRF, the versatility of the proposed framework makes it directly applicable to a wide variety of 3D reconstruction pipelines, including methods such as 3D Gaussian Splatting.

REFERENCES

Barron, J.T. et al. (2022). Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5470–5479.

Barron, J.T. et al. (2023). Zip-NeRF: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19697–19705.

Bell, A.J. et al. (1997). The "independent components" of natural scenes are edge filters. *Vision Research*, 37(23), 3327–3338.

Bian, J.W. et al. (2024). PoRF: Pose residual field for accurate neural surface reconstruction. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*.

Bian, W. et al. (2023). NoPe-NeRF: Optimising neural radiance field with no pose prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4160–4169.

Chng, S.F. et al. (2022). Gaussian activated neural radiance fields for high fidelity reconstruction and pose estimation. In *The 17th European Conference on Computer Vision (ECCV2022)*, 264–280. Springer-Verlag, Berlin, Heidelberg.

Deng, N. et al. (2022). FoV-NeRF: Foveated neural radiance fields for virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 28(11), 3854–3864.

Eldar, Y.C. et al. (2003). MMSE whitening and subspace whitening. *IEEE Transactions on Information Theory*, 49(7), 1846–1851.

Frisch, D. et al. (2021). Deterministic Gaussian sampling with generalized Fibonacci grids. In *Proceedings of the 24th International Conference on Information Fusion (Fusion 2021)*. Sun City, South Africa. doi:10.23919/FUSION49465.2021.9626975.

Frisch, D. et al. (2023). The generalized Fibonacci grid as low-discrepancy point set for optimal deterministic Gaussian sampling. *Journal of Advances in Information Fusion*, 18(1), 16–34.

Frisch, D. et al. (2025). Deterministic sampling with separation of variables in spherical coordinates. In *Proceedings of the 28th International Conference on Information Fusion (FUSION 2025)*, 1–8. Rio de Janeiro, Brazil.

Gallo, E. (2022). The SO(3) and SE(3) lie algebras of rigid body rotations and motions and their application to discrete integration, gradient descent optimization, and state estimation. *arXiv preprint arXiv:2205.12572*.

Hinrichs, A. et al. (2020). Expected dispersion of uniformly distributed points. *Journal of Complexity*, 61, 101483.

Jeong, Y. et al. (2021). Self-calibrating neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5846–5854.

Jia, X. et al. (2024). 3D-measurement of particles and particulate assemblies–a review of the paradigm shift in describing anisotropic particles. *Powder Technology*, 447, 120109.

Kessy, A. et al. (2018). Optimal whitening and decorrelation. *The American Statistician*, 72(4), 309–314.

Kingma, D.P. et al. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 1–15.

Lanckriet, G.R. et al. (2004). Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5(Jan), 27–72.

Lin, C.H. et al. (2021). BARF: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5741–5751.

Meng, Q. et al. (2021). GNeRF: GAN-based neural radiance field without posed camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6351–6361.

Mildenhall, B. et al. (2021). NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1), 99–106.

Park, K. et al. (2023). CamP: Camera preconditioning for neural radiance fields. *ACM Transactions on Graphics (TOG)*, 42(6), 1–11.

Rudnev, V. et al. (2022). NeRF for outdoor scene relighting. In *European Conference on Computer Vision*, 615–631. Springer.