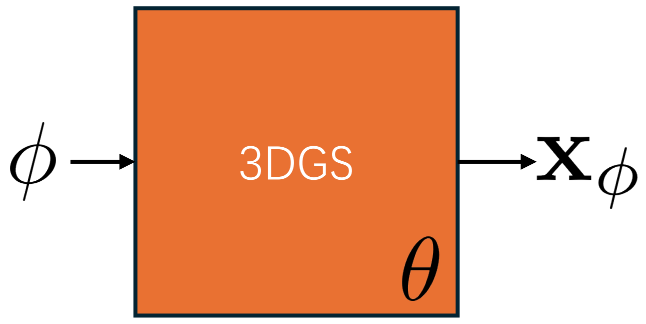
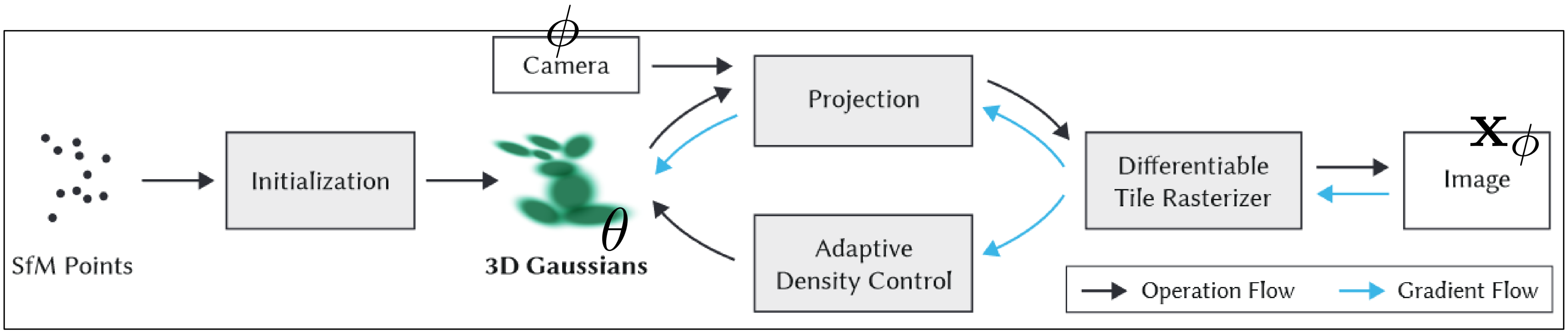
Novel-View Synthesis

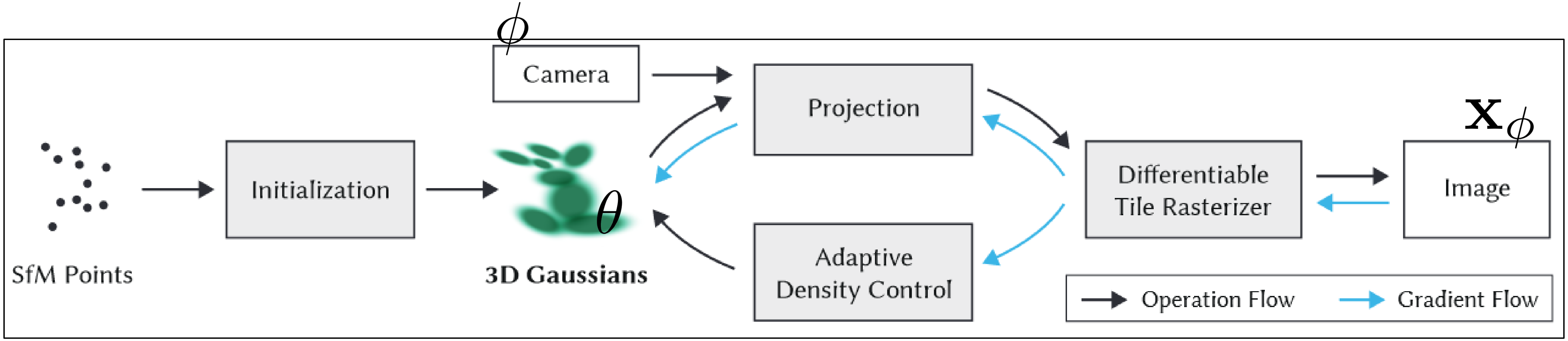
3D Gaussian Splatting, Siggraph'23



Dataset	Mip-NeRF360					
Method|Metric	*SSIM*↑	*PSNR*↑	*LPIPS*↓	Train	FPS	Mem
Plenoxels	0.626	23.08	0.463	25m49s	6.79	2.1GB
INGP-Base	0.671	25.30	0.371	5m37s	11.7	13MB
INGP-Big	0.699	25.59	0.331	7m30s	9.43	48MB
M-NeRF360	0.792†	27.69†	0.237†	48h	0.06	8.6MB
Ours-7K	0.770	25.60	0.279	6m25s	160	523MB
Ours-30K	0.815	27.21	0.214	41m33s	134	734MB



Novel View Synthesis Gaussian Splatting, Siggraph'23



θ : a set of $[\ \underbrace{\mu_i}_{\text{position}\in\mathcal{R}^3},\ \underbrace{\sigma_i}_{\text{opacity}\in\mathcal{R}^1},\ \underbrace{\Sigma_i=(r_i,s_i)}_{\text{covariance}\in\mathcal{R}^7},\ \underbrace{\mathbf{c}_i}_{\text{spherical harmonics}\in\mathcal{R}^{(k+1)^2+1}}\ ]$

Projection:

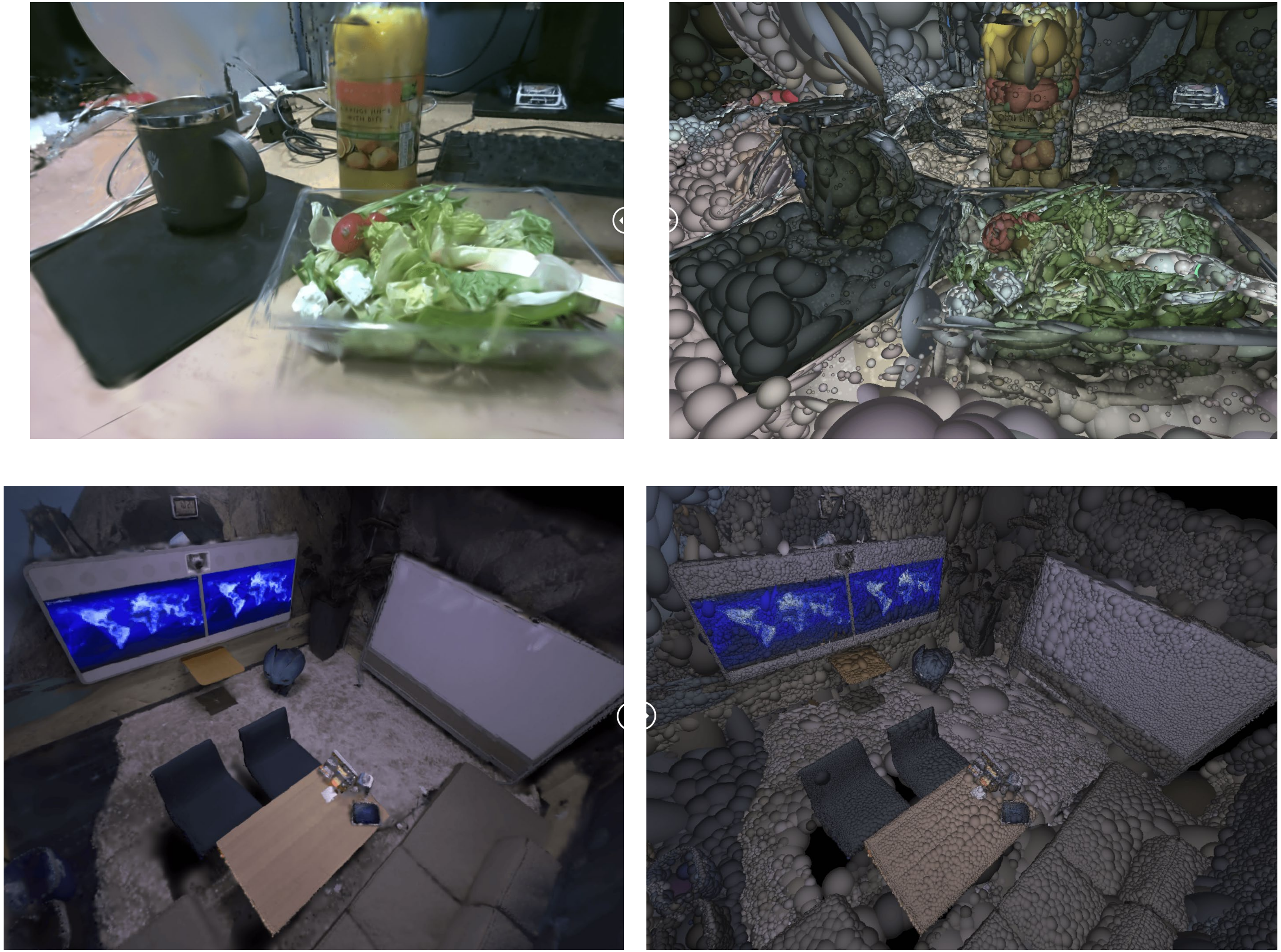
$$\mu_i^{2D}=(K((W\mu_i)/(W\mu_i)_z))_{1:2},$$
$$\Sigma_i^{2D}=(JW\Sigma_iW^TJ^T)_{1:2,1:2},$$

approximation of perspective transform W, K

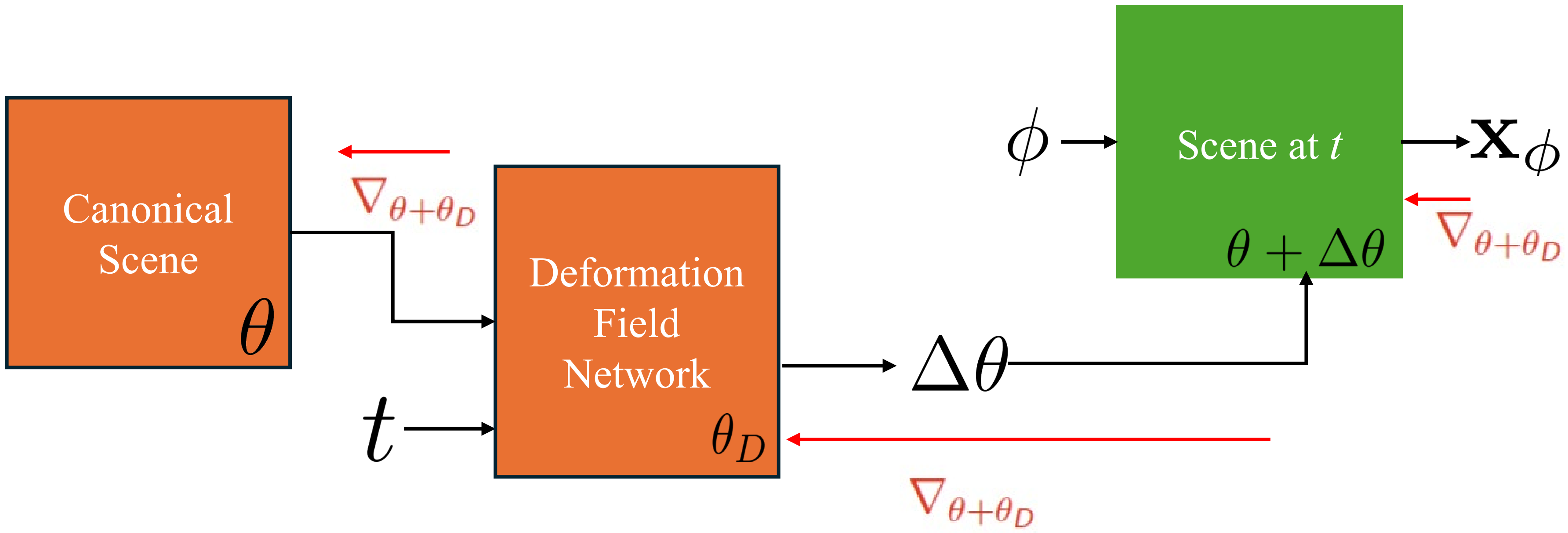
Rasterization:

$$\mathbf{I}=\sum_{i\in\mathcal{N}}\mathbf{c}_i\alpha_i^{2D}\prod_{j=1}^{i-1}(1-\alpha_j^{2D})$$

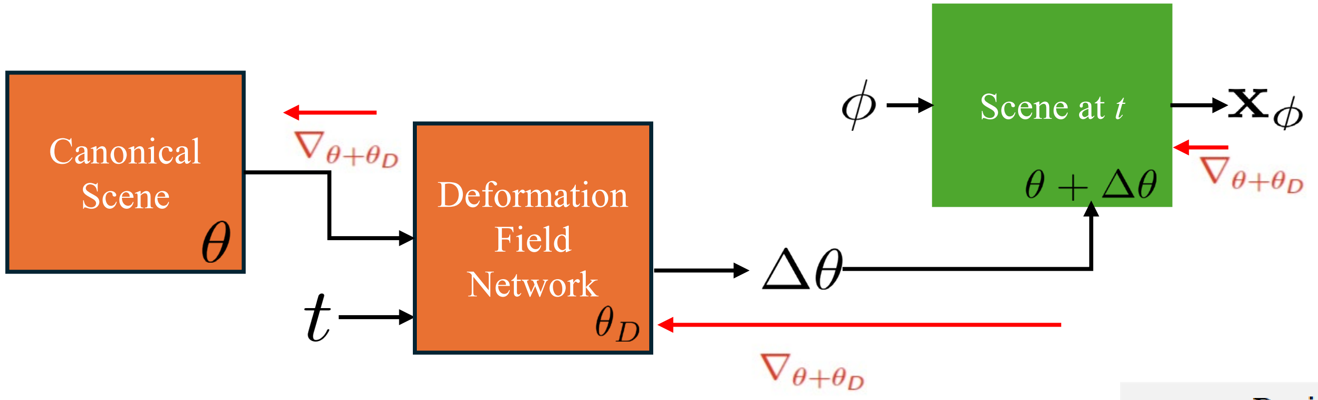
point-based blending of ordered points
v.s. sampling rays in NeRF

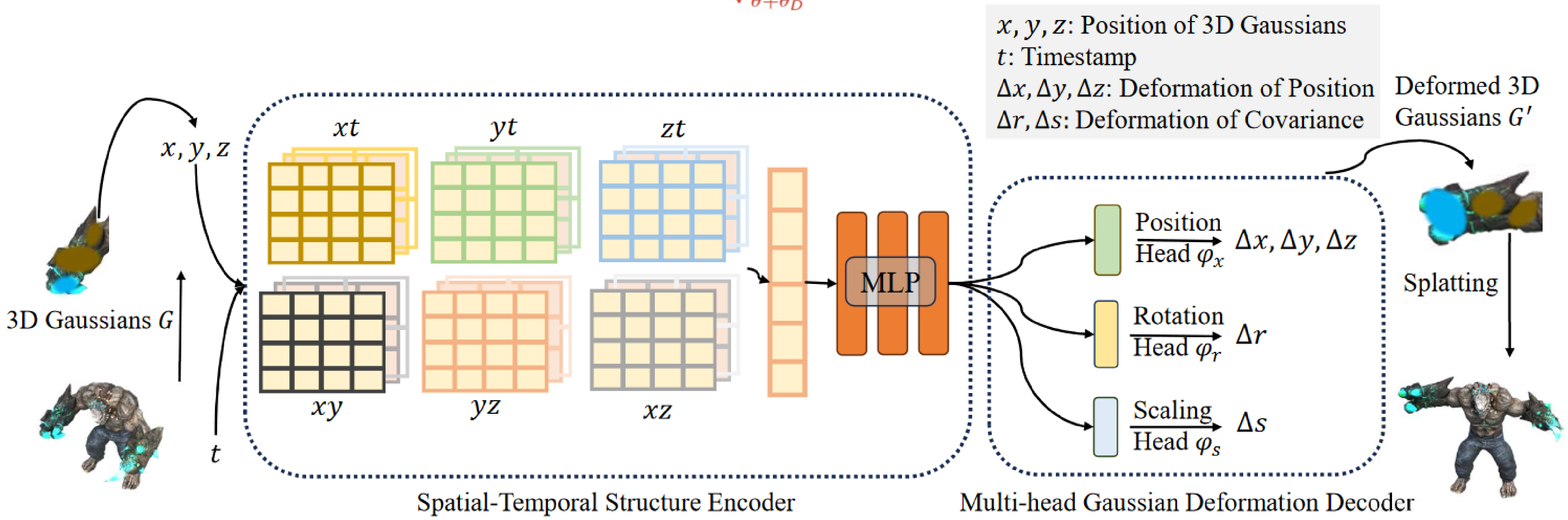


Dynamic Novel-View Synthesis 4D Gaussian Splatting, CVPR'24



Dynamic Novel View Synthesis 4D Gaussian Splatting, CVPR'24

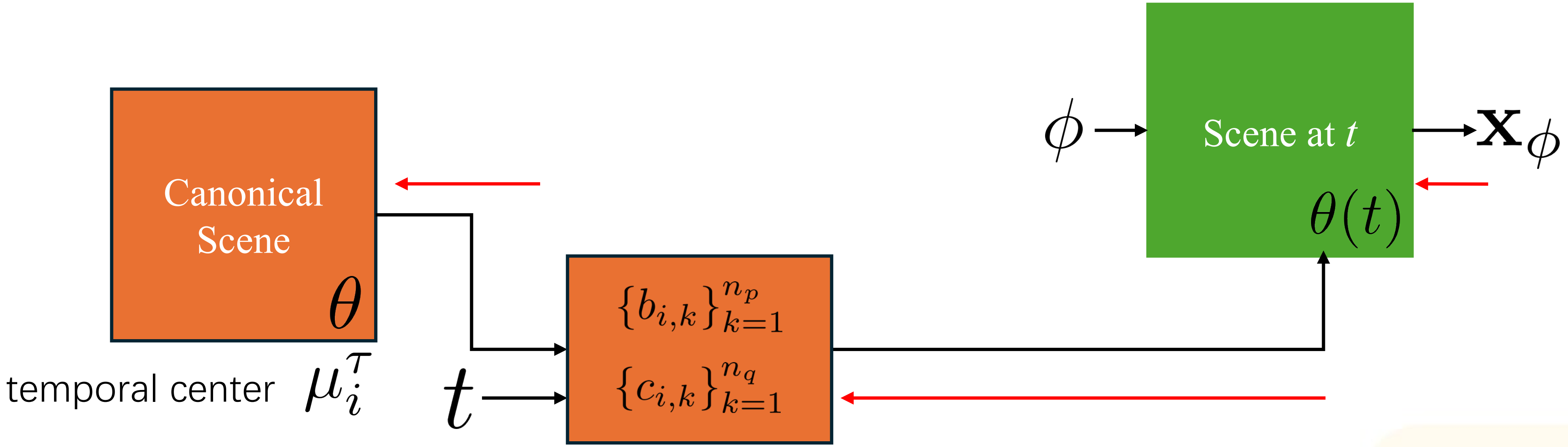




Hexplane Parameter: $R(i,j) \in R^{h \times l \times N_i \times N_j}$

Separate MLP for position, rotation, scaling

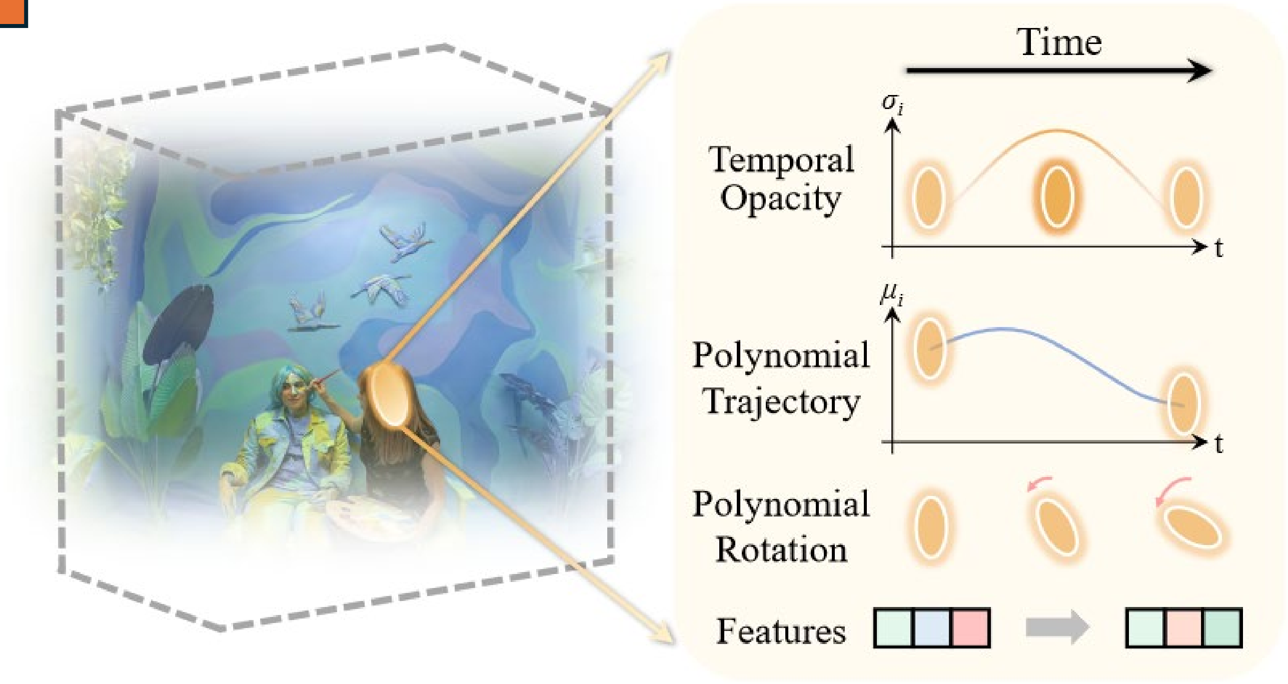
Dynamic Novel View Synthesis Spacetime Gaussian Splatting, CVPR'24



Polynomial Interpolation around Temporal Center

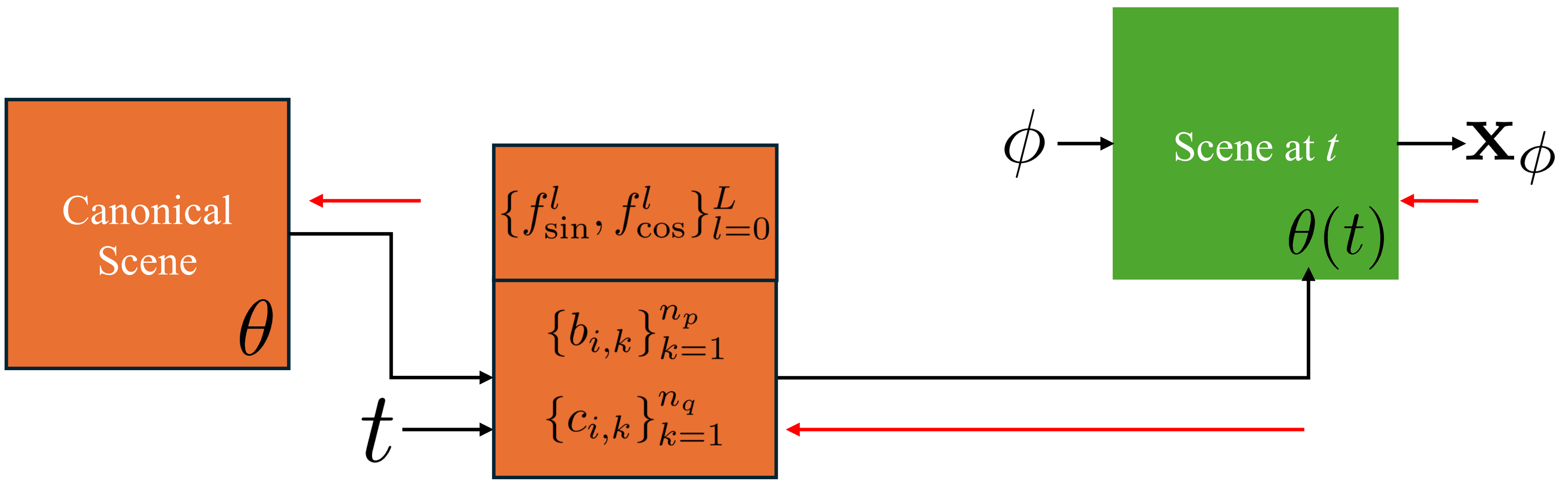
Time-dependent position $$\mu_i(t) = \sum_{k=0}^{n_p} b_{i,k}(t - \mu_i^\tau)^k,$$

Time-dependent rotation $$q_i(t) = \sum_{k=0}^{n_q} c_{i,k}(t - \mu_i^\tau)^k,$$



(a) Spacetime Gaussians

Dynamic Novel View Synthesis Gaussian-Flow, CVPR'24



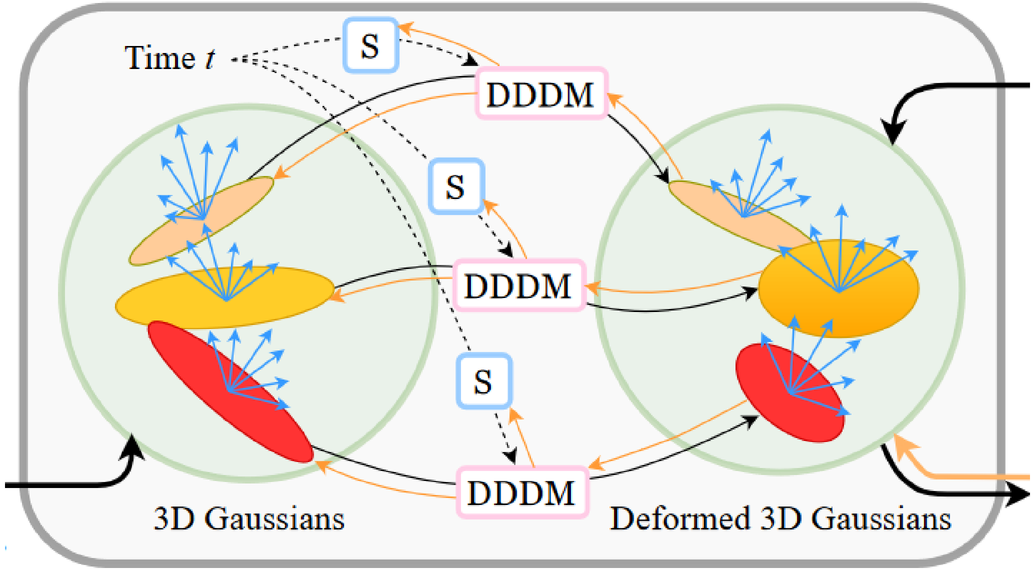
Dual-Domain Deformation Model

polynomial fitting in **time domain**

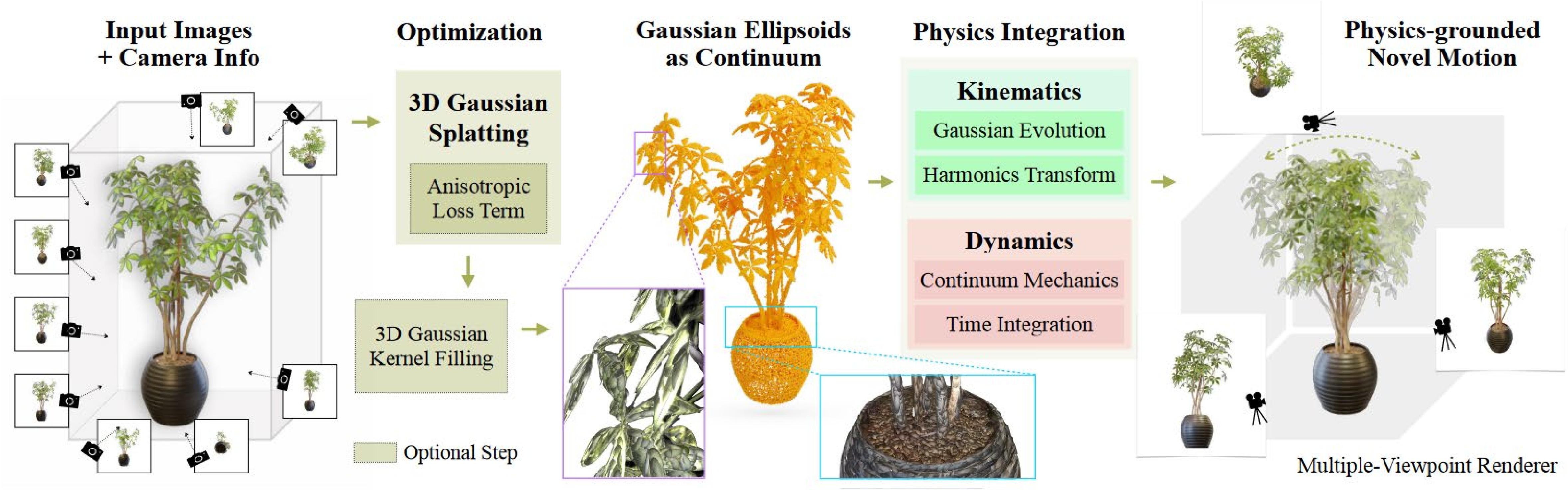
$$P_N(t) = \sum_{n=0}^{N} a_n t^n,$$

Fourier series fitting in **frequency domain**

$$F_L(t) = \sum_{l=1}^{L} \left(f^l_{sin}\cos(lt) + f^l_{cos}\sin(lt)\right)$$



Dynamic Novel View Synthesis PhysGaussian, CVPR'24



Continuum Mechanics

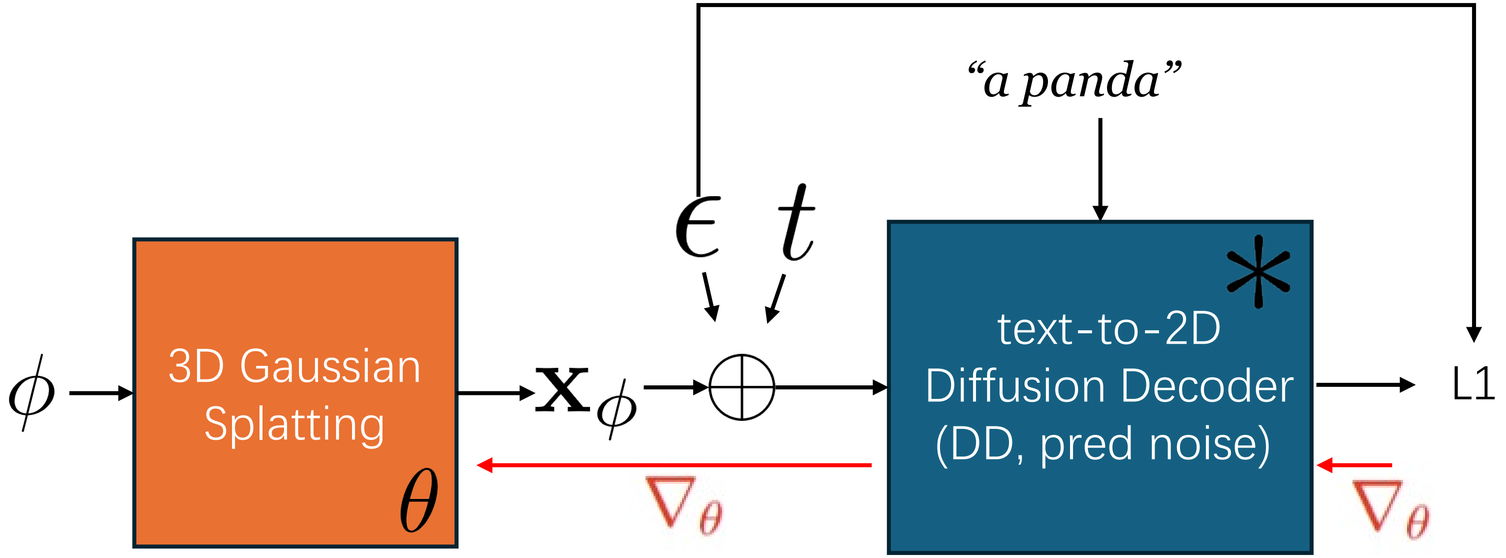
Conservation of mass $$\int_{B_\epsilon^t} \rho(\boldsymbol{x}, t) \equiv \int_{B_\epsilon^0} \rho(\boldsymbol{\phi}^{-1}(\boldsymbol{x}, t), 0),$$

conservation of momentum $$\rho(\boldsymbol{x}, t)\dot{\boldsymbol{v}}(\boldsymbol{x}, t) = \nabla \cdot \boldsymbol{\sigma}(\boldsymbol{x}, t) + \boldsymbol{f}^{\text{ext}},$$

text to 3D

DreamFusion: Score Distillation Sampling, ICLR'23

Intuition: Novel-View Image = Noised Hidden Diffusion Step

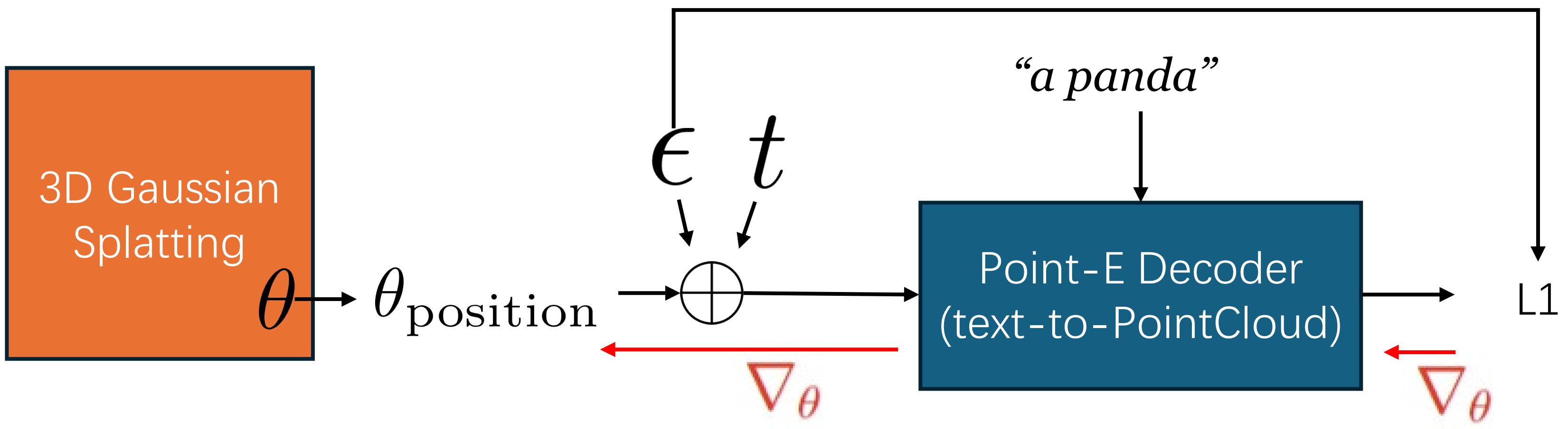


$$L_{\text{SDS}} = E_{t,\epsilon}\left[\left(DD(t, \epsilon, \mathbf{L}, \mathbf{x}_\phi) - \epsilon\right)\frac{\partial \mathbf{x}_\phi}{\partial \theta}\right]$$

text to 3D

GSGEN, CIVPR'24

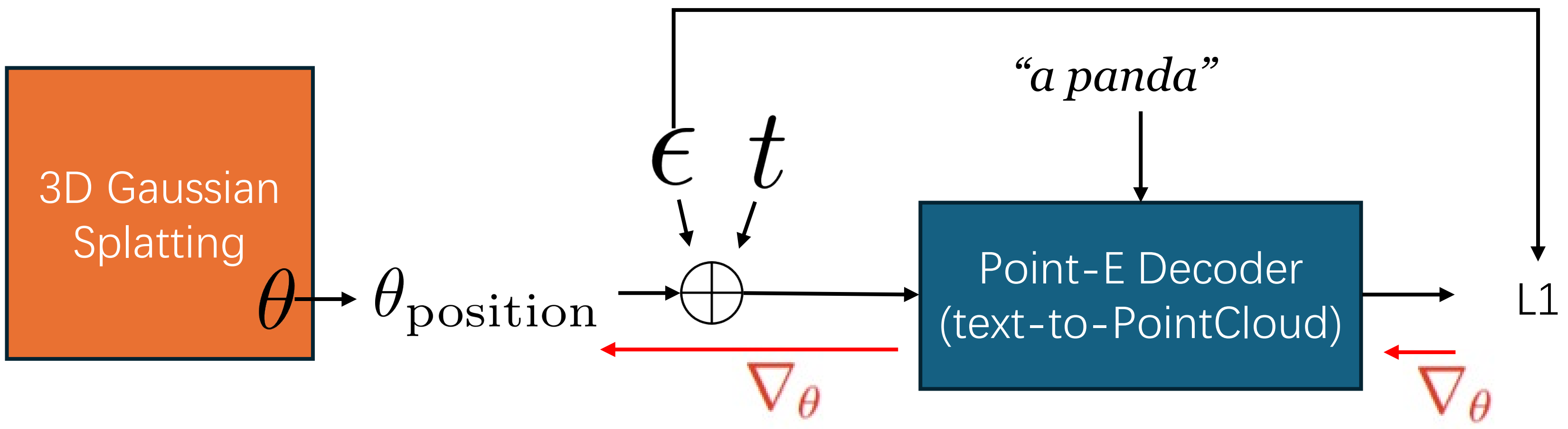
Intuition: Different from NeRF (implicit), GS are Explicit Point-based Representation



$$L_{\text{PCD}} = E_{t,\epsilon}[(DD(t, \epsilon, \mathbf{L}, \theta_{\text{position}}) - \epsilon)]$$

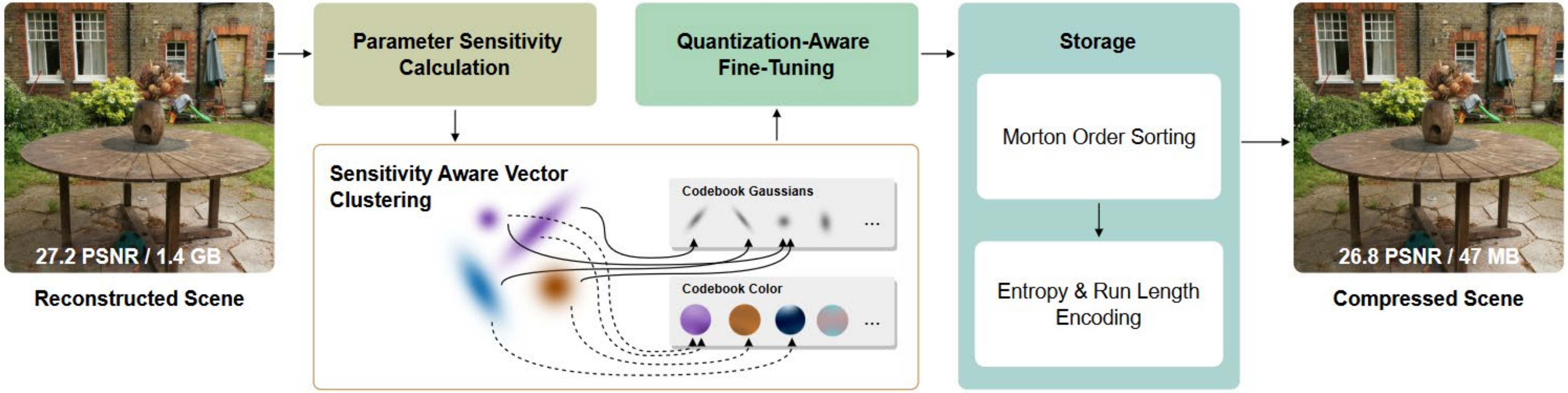
sampling as optimization GSGEN, CVPR'24

Intuition: Different from NeRF (implicit), GS are Explicit Point-based Representation

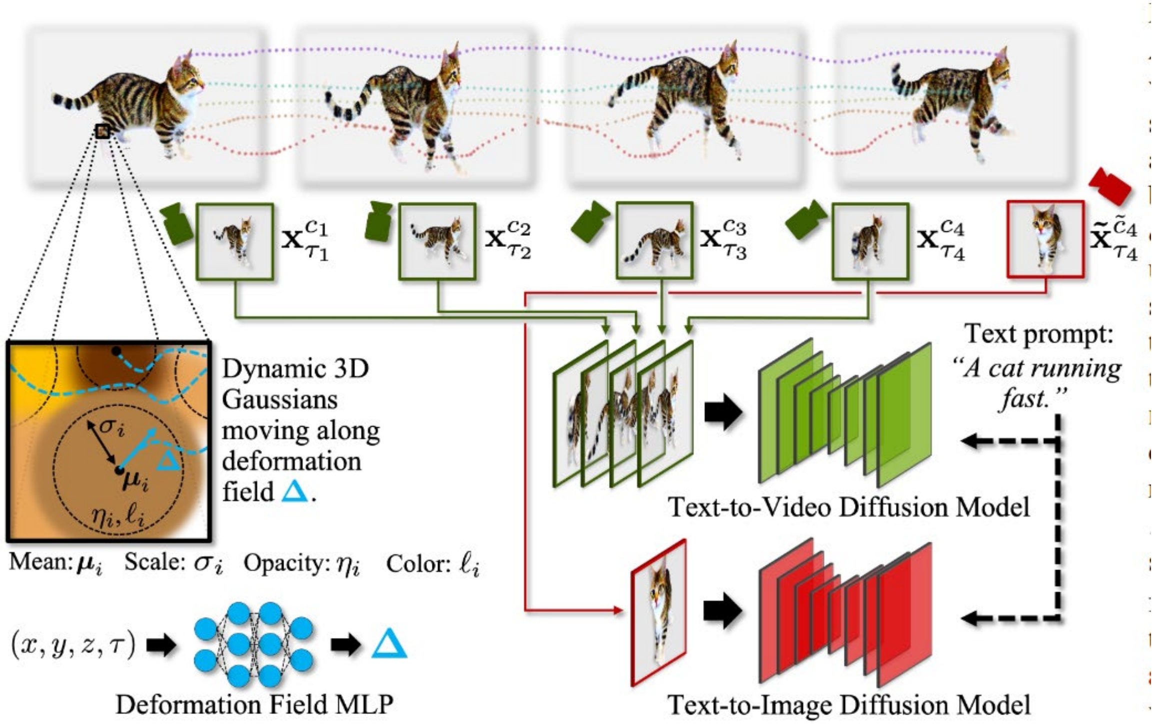


$$L_{\text{PCD}} = E_{t,\epsilon}[(DD(t, \epsilon, \mathbf{L}, \theta_{\text{position}}) - \epsilon)]$$

Efficient Gaussian Splatting Compressed 3D Gaussian, CVPR'24

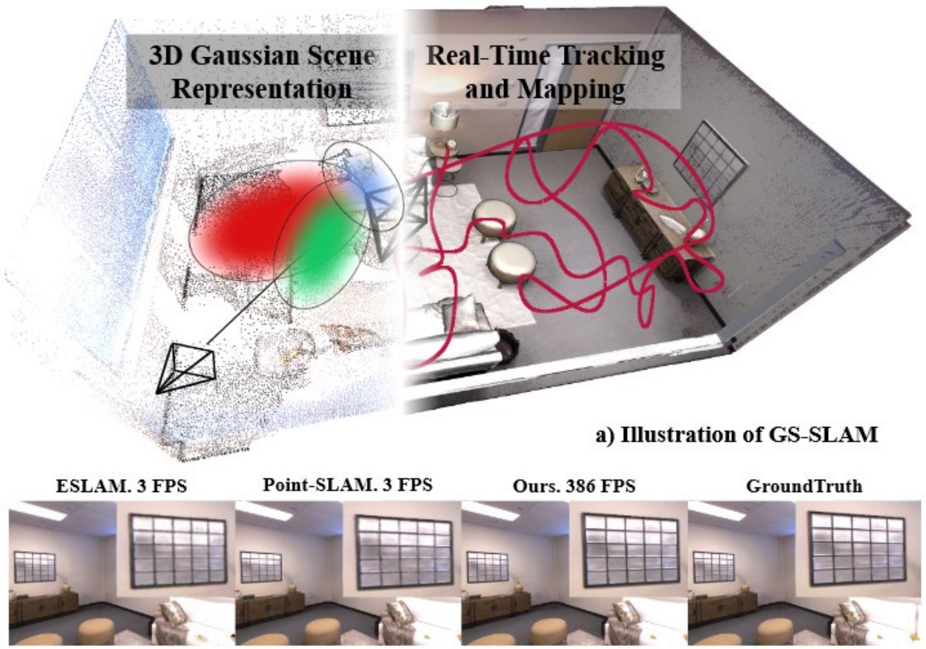


text to 4D Align Your Gaussians, CVPR'24



Simultaneous Localization and Mapping

GS-SLAM, CVPR'24



b) The Rendering Performance of GS-SLAM

Gaussian Splatting SLAM, CVPR'24

incremental 3D reconstruction using a single moving monocular or RGB-D camera