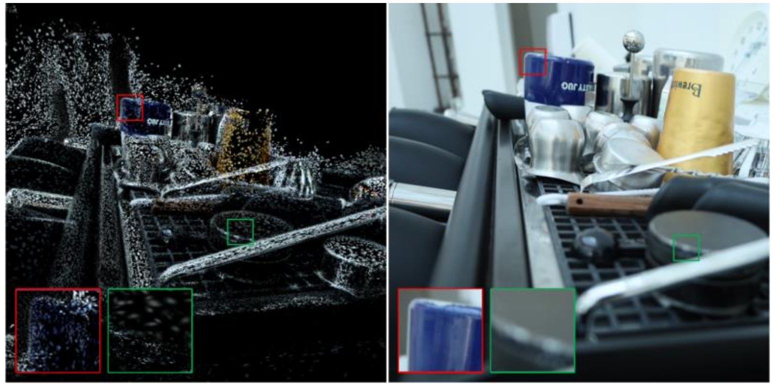
3D Blur Kernel on Gaussian Splatting

Lin Yongchao[1] Su Xiangdong[*2] Yang Yuhan[3]
College of Computer Science Inner Mongolia University
Hohhot, Inner Mongolia, China
[*]Denotes Corresponding Author

Motivation

- 3D Gaussian Splatting cannot reconstruct scenes from images with dynamic blur.
- For areas with blur, 3D Gaussian Splatting fits large and sparse Gaussian points.



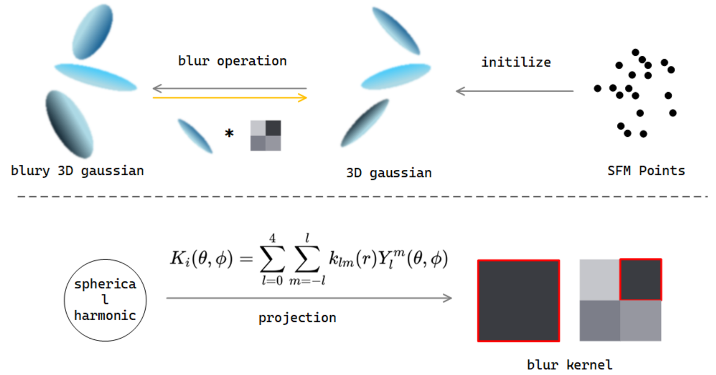
Comparison of Point Density in Regions with Varying Degrees of Blur

Contribution

- We propose a 3D blur kernel to address the training and rendering issues of 3D Gaussian Splatting with blurred images.
- We introduce a point density regularization to enable the model to render blurred areas with denser Gaussian points.

Proposed Method

3D Sparse Blur Kernel



For each Gaussian point, we apply a spherical harmonic. During training, for a given viewpoint, the projection of the spherical harmonic onto a 2D plane forms an N×N-specific viewpoint-dependent blur kernel. This blur kernel acts on the covariance matrix of a 3D Gaussian.

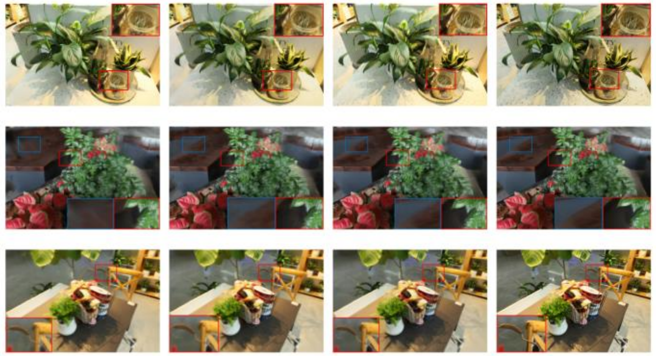
Point Density Regularization

$$\mathcal{L}_d=\sum_i\left(||\Sigma_i^{-1}||_2 e^{-\alpha K_i}\right)$$

This regularization term encourages denser Gaussian distributions and finer variance in areas with more excellent blur, facilitating more accurate reconstruction of details within blurred regions.

Results

Qualitative comparison on real world camera motion blur.



Since our model directly utilizes the 3D Gaussian Splatting pipeline for rendering after training, it achieves rendering speeds of over 100 FPS.

Quantitative results on camera motion scenes.

	cozy2room			factory			pool			tanabata			wine		
camera motion	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NeRF [17]	25.66	.7941	.2288	19.32	.4563	.5304	30.45	.8354	.1932	22.22	.6807	.3653	21.25	.6370	.3633
3D-GS [10]	25.24	.7641	.2262	21.52	.5495	.4641	27.97	.7659	.2833	20.15	.5722	.3968	20.11	.6174	.3790
MPR [25]	27.88	.8502	.1153	21.70	.6153	.3094	30.64	.8385	.1641	22.71	.7199	.2509	22.64	.7141	.2344
Deblur-NeRF [15]	32.08	.9261	.0477	25.60	.7750	.2687	31.61	.9182	.1246	27.11	.8640	.1228	27.45	.8632	.1363
DP-NeRF [11]	31.88	.9175	.0481	28.03	.8628	.1127	31.52	.8946	.1901	26.25	.8517	.0995	25.18	.8067	.1436
BAD-NeRF [27]	30.97	.9014	.0552	31.65	.9037	.1228	31.72	.9280	.1153	23.82	.8311	.1378	28.25	.8727	.0914
Ours	32.13	.9178	.0406	30.95	.9047	.1092	33.08	.9527	.1275	30.36	.8721	.0883	31.25	.9055	.0658

Quantitative results on camera defocus scenes.

	cozy2room			factory			pool			tanabata			wine		
defocus	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NeRF [17]	30.03	.8926	.0885	25.36	.7847	.2351	27.77	.7266	.3340	23.80	.7811	.2142	22.67	.7103	.2799
3D-GS [10]	30.21	.8819	.0897	25.29	.7723	.2365	27.88	.7341	.3298	23.97	.7798	.2176	22.84	.7119	.2763
KPAC [24]	28.15	.8592	.0815	26.40	.8194	.1624	26.69	.6589	.2631	24.81	.8147	.1639	23.42	.7495	.2155
Deblur-NeRF [15]	31.85	.9175	.0481	28.03	.8628	.1127	30.52	.8246	.1901	26.25	.8517	.0995	25.18	.8067	.1436
DP-NeRF [11]	32.06	.9157	.0521	28.37	.8681	.1173	30.79	.8310	.1911	26.42	.8561	.1021	25.48	.8117	.1446
BAD-NeRF [27]	31.24	.9053	.0567	28.67	.8675	.1258	31.01	.8529	.1839	26.18	.8327	.1394	25.57	.8756	.0932
Ours	32.35	.9518	.0478	31.84	.9173	.0934	32.12	.9056	.1756	29.58	.8735	.0867	29.47	.9114	.0821