Appendix

A Data flow

To clarify the decoupling of appearance learning and physical learning, we provide a simpler layour of our pipeline, as shown in Fig. 8.

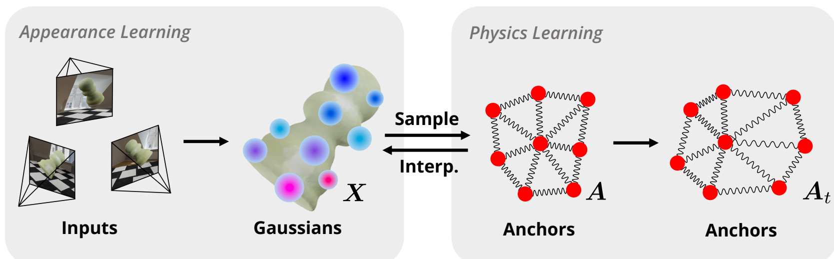

Fig. 8: Simpler layout of the pipeline. In appearance learning, we optimize Gaussian kernels for static image rendering. We sample anchors $\boldsymbol{A}$ from Gaussians' center $\boldsymbol{X}$. In physics learning, we optimize physical parameters among anchors $\boldsymbol{A}$. After simulation, at time t, the updated Gaussians' center $\boldsymbol{X}_t$ could be interpolated from $\boldsymbol{A}_t$.

B More Implementation Details

We use a single NVIDIA RTX 3090 GPU for reconstruction. For static scene reconstruction, we follow the configuration prescribed by 3D Gaussian Splatting [14], which takes approximately 10 minutes to optimize for the synthetic data. The dynamic reconstruction takes 300 iterations. We employ 2048 anchor points for our Spring-Mass model, with each anchor linked to $n_k = 256$ neighbors through springs. We set $n_b = 16$ and $n_c = 16$. For each sequence, we assume mass $m_0 = 1$ and damping factor $\zeta_0 = 0.1$. The weighting coefficient for the D-SSIM term λ_{d-ssim} is set to 0.2 for static reconstruction and 0.05 for dynamic reconstruction. In our experiments, we use a nonlinear spring force, setting $p_k = 0.5$, and for Inverse Distance Weighting (IDW) interpolation, we arbitrarily choose $p_b = 0.5$.

Following the practice from PAC-NeRF [18], we first independently optimize the initial velocity vector $\boldsymbol{v}_0$, utilizing only a few frames captured before the object interacts with the environment.

In terms of the Gaussian kernels' parameters, we optimize all of them during static scene reconstruction while maintaining a constant scaling scalar s_0 for all kernels. We have found that uniform scaling across all kernels in static scene reconstruction results in a more evenly distributed point cloud and anchor points. This consistency markedly improves the dynamic model's simulation capabilities by making the kernels' spatial distribution more uniform. It is important to note

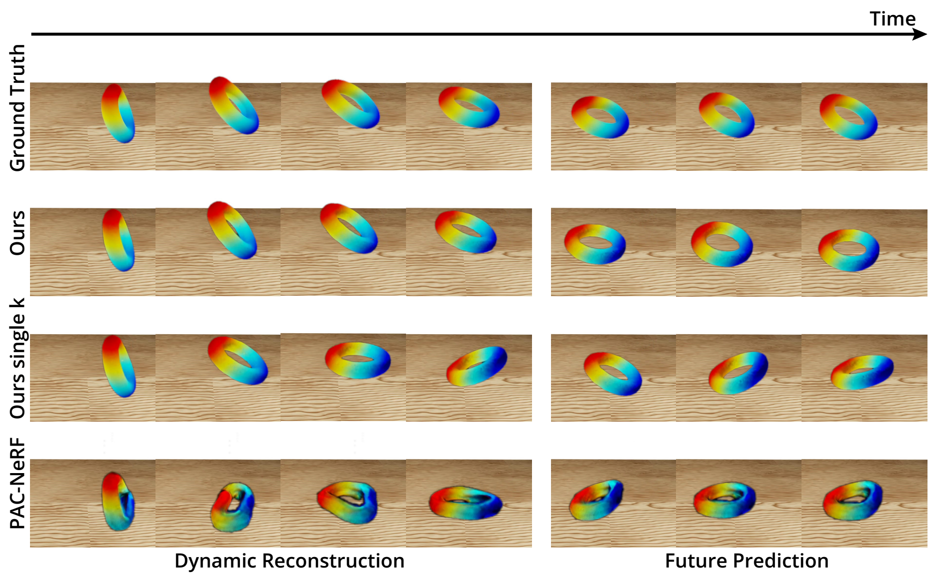

Fig. 9: Ablation study of the effectiveness of optimizing physical parameters for each particle rather than optimizing a single global parameter, on a heterogeneity object. The results shows that optimizing a single global parameter is not able to accurately model objects with complex physical properties.

that we do *not* preserve the scaling scalar as constant s_0 during this refinement phase. Instead, we assign a unique scaling scalar s_i to each Gaussian kernel associated with each anchor point.

C Ablation Study

	Dynamic Reconstruction			Future Prediction		
	CD↓	PSNR↑	SSIM↑	CD↓	PSNR↑	SSIM↑
Spring-Gaus (ours)	**0.18**	**27.08**	**0.967**	**2.04**	**17.63**	**0.927**
Spring-Gaus w/o soft vector η	0.56	25.36	0.959	13.28	13.91	0.881
Spring-Gaus, single k	3.22	23.02	0.940	6.56	14.45	0.892
PAC-NeRF [18]	8.66	19.87	0.916	5.70	15.65	0.894

Table 3: Ablation study. We demonstrate the importance of optimizing parameters for each anchor point individually as well as using a soft vector η. Optimizing parameters for each anchor point allows Spring-Gaus to have a higher degree of freedom in modeling physics, and the soft vector η gives a more flexible formulation.

In our approach, we employ a soft vector η to dynamically regulate both the quantity and intensity of springs linked to the anchor points. This strategy is illustrated in Tab. 3, showcasing its effectiveness. Our method's capability to simulate using very sparse anchors allows for the individual optimization of physical parameters for each anchor point. This contrasts with PAC-NeRF, which utilizes tens of thousands of particles, making it challenging to optimize the physical parameters for each particle infeasible. Consequently, PAC-NeRF faces limitations in accurately modeling objects composed of heterogeneous materials. In contrast, our methodology is adept at handling such complexities. As depicted in Fig. 9 and Tab. 3, we present the outcomes on a heterogeneous object that is segmented into various sections, each with distinct physical properties, thereby demonstrating our model's superior adaptability in capturing the nuanced dynamics of objects with variable material composition.

D Limitations and Future Work

Currently, Spring-Gaus is constrained to modeling elastic objects due to fixed spring lengths in our formulation; these lengths are constants established at the onset of dynamic simulation. Future work should aim to incorporate plastic deformation into the framework. This would involve developing a method to dynamically adjust the original lengths of the springs, also can make and break spring relationships during the simulation, allowing for the accurate modeling of materials that exhibit both elastic and plastic behavior.

Besides, our method focuses on simulating a single object colliding with the ground surface, while multi-object interaction is a fascinating topic but requires new model design (e.g., establishing new springs) and a more thorough evaluation under various challenging scenarios, which we identify as an excellent direction to expand our method. Other directions include considering more complicated boundary conditions and external actions.

Lastly, for better evaluation, a more comprehensive real world dataset with high spatial and temporal resolutions should be collected, including more diverse objects, materials, and interactions. This dataset should also include more challenging scenarios, such as occlusions, lighting changes, and camera motion. This kind of dataset will help to evaluate the robustness and generalization of related methods.