

4C4D: 4 Camera 4D Gaussian Splatting

Junsheng Zhou^{1*} Zhifan Yang^{1*} Liang Han¹ Wenyuan Zhang¹

Kanle Shi² Shenkun Xu² Yu-Shen Liu^{1†}

¹School of Software, Tsinghua University, Beijing, China ²Kuaishou Technology

{zhou-js24, yangzf22, han123, zhangwen21}@mails.tsinghua.edu.cn

{shikanle, xushenkun}@kuaishou.com, liuyushen@tsinghua.edu.cn



Figure 1. **Left:** Our capture system records dynamic scenes using only four cameras. **Right:** Dynamic novel views rendered from the reconstructed 4D representations produced by 4DGS, Ex4DGS, and our method.

Abstract

*This paper tackles the challenge of recovering 4D dynamic scenes from videos captured by as few as four portable cameras. Learning to model scene dynamics for temporally consistent novel-view rendering is a foundational task in computer graphics, where previous works often require dense multi-view captures using camera arrays of dozens or even hundreds of views. We propose **4C4D**, a novel framework that enables high-fidelity 4D Gaussian Splatting from video captures of extremely sparse cameras. Our key insight lies that the geometric learning under sparse settings is substantially more difficult than modeling appearance. Driven by this observation, we introduce a Neural Decaying Function on Gaussian opacities for enhancing the geometric modeling capability of 4D Gaussians. This design mitigates the inherent imbalance between geometry and appearance modeling in 4DGS by encouraging the*

*Equal contribution. † Corresponding author.

4DGS gradients to focus more on geometric learning. Extensive experiments across sparse-view datasets with varying camera overlaps show that 4C4D achieves superior performance over prior art. Project page at: <https://junshengzhou.github.io/4C4D>.

1. Introduction

Novel view synthesis of dynamic scenes is a vital task towards achieving 4D intelligence in various applications, including virtual reality, game design and film production. Prior works [41, 44] reconstruct dynamic scenes using complex systems of dense camera arrays with hundreds of synchronized devices, which confines 4D modeling to research labs or large companies rather than everyday users. In this paper, we address the extremely challenging problem of reconstructing 4D dynamic scenes from sparse camera setups. This technique provides a promising pathway to bring 4D

intelligence into daily life, enabling high-quality dynamic scene reconstruction using only a few mobile phones or portable cameras such as GoPros.

Recent advances [15, 25] in novel view synthesis of 3D static scenes have demonstrated remarkable performance in both rendering quality and efficiency. 3D Gaussian Splatting (3DGS) [15] represents scenes with explicit graphic primitives, enabling real-time photorealistic rendering through rasterization. Extending beyond static observations, 4D Gaussian Splatting [44] and deformation-based 3DGS approaches [41, 45] further advance dynamic scene representation by integrating temporal modeling into Gaussian Splatting. Nevertheless, these methods remain applicable primarily to 4D reconstruction from videos captured by dense camera arrays, which limits their usability in sparse or consumer-level settings.

Through a deep dive into the optimization in 4D Gaussian Splatting under sparse-view settings, we observe that the failure of existing 4D frameworks can largely be attributed to a biased optimization process. Specifically, fitting dynamic appearance remains relatively easy for 4DGS even with limited views, whereas recovering accurate and noise-free geometry is significantly more challenging due to insufficient spatial supervision. The current Gaussian formulation fails to properly balance appearance and geometry learning, resulting in biased optimization that overfits to the easier appearance cues while neglecting the underlying geometric consistency.

Driven by this observation, we propose 4C4D, a novel 4D Gaussian Splatting framework for reconstructing high-fidelity dynamic scenes from as few as four camera views. 4C4D addresses the inherent imbalance between geometry and appearance learning in sparse-view 4DGS by introducing an adaptive learning scheme that grants more flexibility to geometric optimization, leading to a more balanced overall training process. To achieve this, we propose the Neural Decaying Function, a novel mechanism implemented with neural networks that adaptively controls the decay of Gaussian opacities. The Neural Decaying Function takes key attributes of each 4D Gaussian primitive (e.g. center, opacity, rotation) as input and predicts a decay factor that neurally modulates Gaussian opacity. By introducing neural control over the opacity, which plays a crucial role in guiding 4DGS geometric learning, 4C4D directs the optimization gradients to focus more effectively on the learning of scene geometry, thereby achieving a better balance between geometric fidelity and appearance modeling.

Simply adopting the Neural Decaying Function across all 4D Gaussians leads to suboptimal training. The reason is that the gradients only exist in the visible regions of 4DGS that contribute to Gaussian rendering, while applying the same decay function to the opacities of invisible regions distorts the optimization process and leads to per-

formance degradation. To address this issue, we introduce a visibility detection strategy that operates across both spatial and temporal domains to identify visible 4D Gaussians at each view and timestep. We then apply separate decay strategies for visible and invisible regions, ensuring that the Gaussian gradients correctly update the Neural Decaying Function during optimization. We evaluate 4C4D on dynamic scene reconstruction from sparse-view inputs across multiple datasets, including Neural3DV, ENeRF-Outdoor, Mobile-Stage and our self-captured Dyn4Cam, which cover both large and small camera overlaps. Experimental results demonstrate that 4C4D achieves superior and temporally consistent novel-view synthesis compared to existing state-of-the-art methods. Our main contributions can be summarized as follows.

- We propose 4C4D, a novel 4D Gaussian Splatting framework that reconstructs high-fidelity dynamic scenes from sparse camera views as few as four, enabling temporally consistent and flexible novel-view rendering across all time steps.
- We introduce the Neural Decaying Function, which adaptively guides Gaussian geometric learning by encouraging 4DGS gradient to focus more on Gaussian opacities. We further design separate decay strategies for visible and invisible regions to ensure stable optimization.
- We evaluate 4C4D on dynamic scene modeling across sparse-view datasets with varying camera overlaps, and show our superiority over the state-of-the-art methods.

2. Related Work

2.1. Novel View Synthesis

Novel view synthesis [2, 4, 15, 25] for static 3D scenes is a long-standing task in 3D computer vision that has drawn significant attention since the introduction of Neural Radiance Fields (NeRF) [25]. NeRF achieves high-quality novel view synthesis by performing volume rendering over view-dependent attributes predicted by a neural network-based function. Subsequent works have extended NeRF to various scenarios, including sparse-view inputs [13, 32, 35], large-scale scenes [33, 36], surface reconstruction [27, 37, 38], and fast learning [4, 6, 26, 50].

Nevertheless, the NeRF family of methods suffers from limited rendering efficiency, which hinders their deployment in real-world applications requiring real-time rendering. Recently, 3D Gaussian Splatting (3DGS) [15] has been proposed, advancing novel view synthesis by enabling real-time rendering through rasterization of explicit Gaussian primitives. Beyond static reconstruction, 3DGS has also been explored for 3D generation [12, 34, 49, 51, 52], reconstruction [5, 11, 54], and material modeling [46–48]. Despite these advances, the above methods remain largely confined to static 3D modeling and lack the ability to cap-

ture and represent 4D dynamics.

2.2. Dynamic Scene Modeling

Modeling dynamic scenes from multi-view videos is a more challenging task that enables novel view synthesis across all time steps. The key difficulty lies in accurately representing the temporal motion of dynamic scenes. Inspired by the success of NeRF, a series of works [7, 17, 28–30] extend Neural Radiance Fields to dynamic scene modeling by incorporating time awareness into the network. DyNeRF [17] introduces temporal conditioning into NeRF representations, enabling dynamic scene synthesis with radiance fields. To improve rendering efficiency, HexPlane [3] and K-Planes [7] advance volumetric representations with compact feature planes, achieving faster dynamic NeRF training.

More recently, researchers have explored the potential of 3D Gaussian Splatting (3DGS) for 4D learning [1, 8–10, 16, 18, 19, 21–24, 31, 40, 53]. Dy3D Gaussian [24] proposes a tracking-based approach that explicitly controls Gaussian motion over time. Works such as Deformable-3DGS [45] and 4DGaussians [41] adopt plane-feature-based architectures similar to dynamic NeRFs to model Gaussian deformation fields. Different from these deformation-based approaches, 4D Gaussian Splatting (4DGS) [44] introduces a more direct dynamic representation by embedding time-aware attributes into Gaussian primitives. ST-GS [18] employs a hybrid scheme that combines temporally aware Gaussian representations with neural network-based appearance decoding. FreeTimeGS [39] further introduces motion attributes to enhance performance under complex dynamic motions. In addition, LongVolCap [43] focuses on reconstructing long 4D video sequences, and Diffuman4D [14] leverages diffusion-based priors for improved human motion modeling.

3. Method

We introduce 4C4D, a novel 4D Gaussian Splatting framework for reconstructing high-fidelity dynamic scenes from as few as four views. The overview of 4C4D is shown in Fig. 2. We begin with previewing 4D Gaussian Splatting in Sec. 3.1. Then, we introduce the methodology and implementation of Neural Decaying Function, as shown in Sec. 3.2. A hyper decay strategy that separately represents the decay function based on visibility detection is proposed in Sec. 3.3.

3.1. Preliminary: 4D Gaussian Splatting

4D Gaussian Splatting (4DGS) represents a dynamic 4D scene using a set of Gaussians, each parameterized by attributes that model the time-varying geometry and appearance of the scene. A 4DGS $G = \{g_i\}_{i=1}^N$, consisting of N 4D Gaussians, represents each Gaussian with key attributes:

position $\sigma = \{x, y, z\}$, rotation r , opacity o , scale s , spherical harmonics coefficients c and the temporal attributes μ_t and s_t , which define the timestep and duration of the 4D Gaussian, respectively.

Similar to 3DGS, 4DGS formulates the 4D covariance matrix as $\Sigma = RSS^T R^T$, which can be factored in to a scale matrix $S \in \mathbb{R}$ and a rotation matrix $R \in \mathbb{R}^4$. The temporal attributes govern the evolution of the geometry and appearance of each 4D Gaussian at different timesteps. Specifically, the position of the 4D Gaussian g at timestep \tilde{t} is given by:

$$\sigma(\tilde{t}) = \sigma + \sum_{1:3,4} \sum_{4,4}^{-1} (\tilde{t} - \mu_t). \quad (1)$$

The opacity o of 4D Gaussian g at timestep \tilde{t} can be formulated as:

$$o(\tilde{t}) = \omega(\tilde{t}) * o, \quad (2)$$

$$\omega(\tilde{t}) = \exp\left(-\frac{1}{2} \frac{(\tilde{t} - \mu_t)^2}{\sum_{4,4}}\right), \quad (3)$$

where $\omega(\tilde{t})$ represents a temporal opacity factor that controls the impact of each 4D Gaussian over time. In addition to the above geometric attributes, 4DGS models the appearances of each Gaussian using 4D spherical harmonics coefficients c . The coefficients model the color as $c(t, \theta, \phi)$, where (θ, ϕ) represents the normalized view direction. The 4D spherical harmonics coefficients are formulated as:

$$c(t, \theta, \phi) = \cos\left(\frac{2\pi n}{T} t\right) Y_{lm}(\theta, \phi), \quad (4)$$

where $Y_{lm}(\theta, \phi)$ is the spherical harmonics basis function with direction (θ, ϕ) , and n donates the order of the Fourier series.

3.2. Neural Decaying Function

We begin by analyzing the core challenge of 4D Gaussian Splatting (4DGS) under sparse-view conditions, which is the biased optimization that tends to overfit to appearance cues while neglecting geometric consistency. We show an example of training 4D Gaussian Splatting using only four cameras in Fig. 3. The figure shows the rendered geometry (depth) and appearance (RGB color) from the learned 4DGS under the training and testing views, respectively. As illustrated, 4DGS accurately reproduces the appearances at the training views but struggles to produce clean and consistent depth geometry due to insufficient spatial supervision, leading to noticeable artifacts and poor generalization to unseen viewpoints.

Driven by this observation, we introduce a neural opacity-decaying control that guides the optimization of 4DGS toward more faithful geometric learning. Specifically, we design a Neural Decaying Function f_θ implemented with a simple neural network. Given key attributes

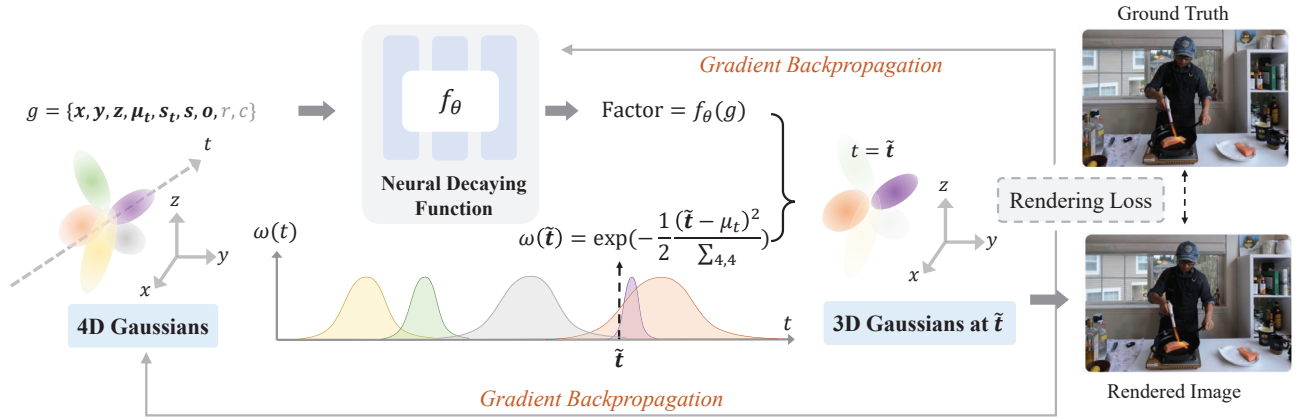


Figure 2. **Overview of 4C4D.** We introduce a Neural Decaying Function f_θ , implemented as a lightweight neural network, to adaptively control the opacity decay of Gaussians. Given key Gaussian attributes as input, f_θ predicts a factor that controls the decay of Gaussian opacities. During training, both the Neural Decaying Function and the 4D Gaussians are jointly optimized via gradient backpropagation under a photometric rendering loss.



Figure 3. Rendered **RGB** and **depth** results from 4DGS trained with and without our Neural Decaying Function, evaluated on both training and unseen views. Without neural decaying, 4DGS can overfit the training views but fails to generalize to unseen viewpoints, due to the learned poor geometry.

of each 4D Gaussian primitive, including its center $\sigma = \{x, y, z\}$, opacity o , and rotation r , the function predicts a decay factor τ for the Gaussian:

$$\tau = f_\theta(x, y, z, o, r). \quad (5)$$

The predicted τ neurally modulates Gaussian opacity, and the final opacity at timestep \tilde{t} in 4C4D can be achieved with:

$$o(\tilde{t}) = \tau * \exp\left(-\frac{1}{2} \frac{(\tilde{t} - \mu_t)^2}{\sum_{4,4}}\right) * o. \quad (6)$$

This neural decaying mechanism redistributes the optimization gradients to emphasize geometric learning, thereby mitigating the overfitting bias and balancing geometry fidelity with appearance modeling. As shown in Figs. 3, 4C4D achieves both geometrically consistent depth reconstructions and visually appealing appearances across training and novel testing views.

3.3. Separate Decay Strategy based on Visibility

We further justify that simply adopting the Neural Decaying Function to all 4D Gaussians lead to suboptimal training. The reason is that the gradients of 4DGS exist only in the visible regions that contribute to rendering at current view. Applying f_θ to invisible Gaussians distorts the optimization and results in performance degradation. To address this issue, we introduce a visibility detection strategy that operates in both spatial and temporal domains to identify visible 4D Gaussians at the current view and timestep. We then apply f_θ only to the visible regions while adopting a separate decay strategy for invisible 4D Gaussians. This ensures that the Gaussian gradients correctly update both the Neural Decaying Function and the 4D Gaussian primitives during optimization. The visibility detection of 4DGS G at camera view \tilde{v} and timestep \tilde{t} is formulated as:

$$G_m = Z_V(\tilde{v}, \sigma, Z_T(\tilde{t}, s_t, G)), \quad (7)$$

where Z_V and Z_T donate the visibility detection based on view intersection and time intersection, respectively. Z_V considers the 4D Gaussian centers σ and filters out the Gaussians whose centers are not visible at current view. Z_T considers the temporal span s_t of each Gaussian and filters out those Gaussians whose duration does not include \tilde{t} .

Through the visibility detection, we now obtain the visible 4D Gaussians G_m , while the invisible ones that do not



Figure 4. Visual comparisons under Neural3DV dataset.

contributes to Gaussian rendering at the current view and timestep are defined as $G_m^* = G - G_m$. We further find that applying a small constant decay to the opacity of invisible 4D Gaussians stabilizes training, consistent with the observations in [11]. The final separate decay strategy for each 4D Gaussian g at the full set of 4DGS is defined as:

$$\tau(g) = \begin{cases} f_\theta(x, y, z, o, r) & \text{if } g \in G_m, \\ \beta & \text{if } g \in G_m^*, \end{cases} \quad (8)$$

where the decay factor β for invisible Gaussians is set to 0.999.

4. Experiment

4.1. Setup

Dataset. We conduct comprehensive experiments on four datasets to evaluate 4C4D, including Neural3DV [17], ENeRF-Outdoor [20], Mobile-Stage [42], and our self-captured Dyn4Cam dataset. Neural3DV [17] is a widely used multi-view video dataset containing six dynamic scenes. Each scene is recorded with 18–21 cameras at a resolution of 2704×2028 and 30 FPS. Following prior works, we downsample the images by a factor of 0.5 before training, and we use the first 300 frames of each scene for both training and evaluation. For the sparse-view setting, we select four cameras located at the most distant positions as

the training views, while the remaining camera videos are used as test views. ENeRF-Outdoor [20] contains three outdoor dynamic scenes captured by an 18-camera array. The videos are recorded at a resolution of 1920×1080 and 60 FPS. We adopt the same four-camera train/test split strategy as in Neural3DV. Mobile-Stage, released by 4K4D [42], contains a complex scene featuring highly dynamic multi-person dance performances. The scene is captured using 24 cameras at a resolution of 1920×1080. We select four views with large overlap for training, and use the remaining views as test data.

Finally, since all existing datasets are densely captured using camera arrays with around 20 devices, we additionally collect Dyn4Cam, a new dataset captured with only four portable GoPro-style cameras. The videos are recorded at 60 FPS with a resolution of 1920×1080, and each scene contains 300 frames. As Dyn4Cam does not include held-out test views, we only provide qualitative comparisons on this dataset.

Metrics. Following prior works [18, 39, 44], we report PSNR, DSSIM and LPIPS as our evaluation metrics for assessing the visual quality of novel view rendering. PSNR is the most commonly used pixel-wise metric that measures the per-pixel ℓ_2 difference between the rendered image and the ground-truth image. DSSIM evaluates the structural dissimilarity between two images, and we report two variants,

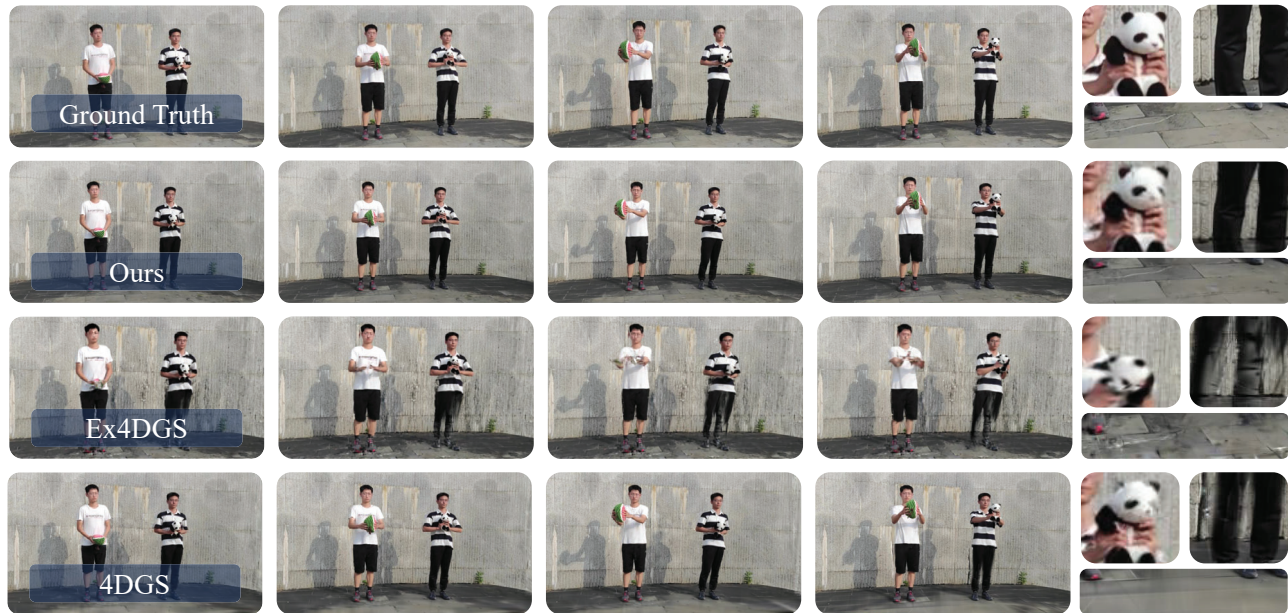


Figure 5. Visual comparisons under ENeRF-Outdoor dataset.

Table 1. Quantitative comparison under Neural3DV dataset.

Method	PSNR \uparrow	DSSIM ₁ \downarrow	DSSIM ₂ \downarrow	LPIPS \downarrow
4DGaussians	20.82	0.117	0.077	0.190
STGS	17.70	0.158	0.107	0.325
4DGS	20.60	0.143	0.094	0.244
Ex4DGS	19.33	0.149	0.099	0.239
Ours	22.29	0.098	0.062	0.146

DSSIM₁ and DSSIM₂, computed with different data-range settings of 1.0 and 2.0. LPIPS measures perceptual similarity between images, and lower values indicate better perceptual quality.

Baselines. We compare 4C4D with state-of-the-art Gaussian-based baselines including 4DGaussians [41], Ex4DGS [16], ST-GS [18], and 4DGS [44]. All baselines follow the same experimental setup as 4C4D to ensure a fair comparison.

4.2. Results of 4-Camera Novel View Synthesis

Neural3DV. We conduct comprehensive evaluations on the Neural3DV dataset. As shown in Tab. 1, 4C4D consistently outperforms all state-of-the-art baselines across all quantitative metrics, demonstrating clear advantages in both geometric accuracy and photometric quality. Visual comparisons in Fig. 4 further highlight the superiority of our reconstructions. Under the challenging sparse-view setting with supervision from only four cameras, most baselines exhibit unstable 4D reconstruction behaviors, including noisy point distributions, drifting primitives, and noticeable Gaussian artifacts.

Table 2. Quantitative comparisons under ENeRF-Outdoor dataset.

Method	PSNR \uparrow	DSSIM ₁ \downarrow	DSSIM ₂ \downarrow	LPIPS \downarrow
4DGaussians	18.21	0.300	0.183	0.456
STGS	15.51	0.315	0.186	0.688
4DGS	23.52	0.183	0.112	0.151
Ex4DGS	21.89	0.224	0.138	0.263
Ours	24.32	0.162	0.097	0.121

In contrast, 4C4D produces clean and temporally coherent 4D Gaussian structures, yielding high-fidelity appearance and robust geometry even in regions with limited viewpoint coverage. This enables visually compelling novel view synthesis across a wide range of timesteps and viewpoints. The visualizations in Fig. 4 follow a circular camera trajectory over time, showing the reconstructed scenes at multiple timestamps and perspectives, and clearly revealing improvements in temporal stability, geometric completeness, and rendering consistency achieved by our approach.

ENeRF-Outdoor. We further evaluate 4C4D on the ENeRF-Outdoor dataset, which contains real-world scenes characterized by complex object motions, uncontrolled lighting conditions, and significant background clutter. These outdoor environments introduce strong illumination variations, shadows, and reflections, making dynamic 4D reconstruction especially challenging. Quantitative results in Tab. 2 demonstrate that 4C4D substantially outperforms all baseline methods, achieving consistently superior performance across all evaluation metrics.

Detailed visualizations in Fig. 5 reveal that 4C4D reconstructs cleaner and more detailed 4D scenes, resulting

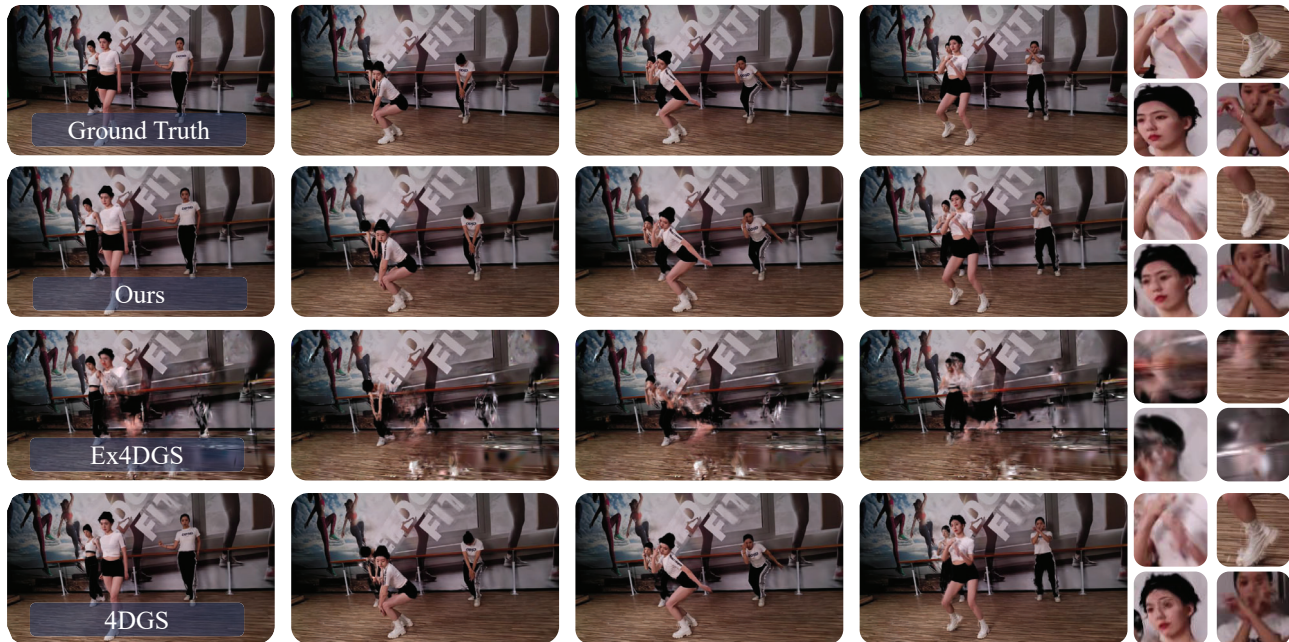


Figure 6. Visual comparisons under Mobile-Stage dataset.

Table 3. Quantitative comparison under Mobile-Stage dataset.

Method	PSNR \uparrow	DSSIM ₁ \downarrow	DSSIM ₂ \downarrow	LPIPS \downarrow
4DGaussians	20.15	0.218	0.144	0.226
STGS	14.30	0.321	0.229	0.658
4DGS	22.15	0.199	0.135	0.180
Ex4DGS	17.85	0.272	0.193	0.260
Ours	22.36	0.194	0.129	0.121

in noticeably higher rendering quality. In particular, 4C4D exhibits strong robustness in modeling rapid motions from sparse-view videos, successfully capturing the dynamics of objects such as the doll and the hands. Moreover, 4C4D consistently improves the rendering quality at the static regions like the grounds.

Mobile-Stage. We further evaluate 4C4D on the Mobile-Stage dataset, which contains large-scale outdoor scenes with multiple actors performing highly dynamic and coordinated dance motions. This dataset presents substantial challenges for 4D reconstruction, including rapid limb movements, complex multi-person interactions, frequent self-occlusions, and cluttered backgrounds. Qualitative and quantitative comparisons are provided in Fig. 6 and Tab. 3.

As shown in Tab. 3, our method achieves the strongest performance across all evaluation metrics. The visualizations in Fig. 6 further highlight that 4C4D captures both high-fidelity geometry and consistent appearance throughout the entire dynamic sequence. Importantly, while all competing baselines struggle to reconstruct the subtle and fast motion details in the three-person dance performance,

4C4D successfully recovers these fine-scale dynamics from only four input cameras. This result demonstrates the robustness of 4C4D in challenging real-world multi-person scenarios and its ability to generalize effectively to large-scale dynamic scenes.

Dyn4Cam. To further demonstrate the effectiveness of 4C4D on multi-view videos captured with sparse and fully portable cameras, we additionally evaluate our method on Dyn4Cam, a self-captured dynamic scene dataset. Dyn4Cam is recorded using only four GoPro-style consumer cameras that are inexpensive and easy for everyday users to deploy, with the entire capture setup costing less than 1,500 USD.

As shown in Fig. 7, 4C4D generates more temporally consistent 4D dynamics and produces significantly more faithful and stable dynamic novel-view renderings compared to previous methods. Because Dyn4Cam is collected using only four cameras and does not contain any extra viewpoints for numerical evaluation, we do not report quantitative metrics on this dataset. The visualizations in Fig. 7 follow a circular camera trajectory over time, illustrating the reconstructed scenes at various timesteps and viewpoints and highlighting the differences in temporal stability and geometric fidelity across methods.

4.3. Ablation Study

Framework Designs. To evaluate the effectiveness of the major components in our method, we conduct a series of ablation studies on the representative data in Neural3DV dataset and report the quantitative performance in Tab. 4.



Figure 7. Visual comparisons under the self-captured Dyn4Cam dataset.

Table 4. Ablation study on the framework designs.

Method	PSNR \uparrow	DSSIM ₁ \downarrow	DSSIM ₂ \downarrow	LPIPS \downarrow
w/o Neural Decay	22.60	0.097	0.062	0.147
w/o Vis Detect	24.49	0.075	0.045	0.127
Full	24.68	0.070	0.042	0.115

Table 5. Ablation study on decaying functions.

Method	PSNR \uparrow	DSSIM ₁ \downarrow	DSSIM ₂ \downarrow	LPIPS \downarrow
None	22.60	0.097	0.062	0.147
Constant	24.31	0.075	0.046	0.125
Pow Decay	24.35	0.074	0.045	0.124
Exp Decay	24.32	0.077	0.047	0.135
Ours	24.68	0.070	0.042	0.115

We first assess the contribution of the proposed Neural Decaying Function by removing the learnable decay module and replacing it with a fixed decay scheme, reported as “w/o Neural Decay”. The result demonstrates that an optimal opacity decay that helps the geometric learning in 4DGS plays a crucial role in improving dynamic modeling capabilities. We then examine the importance of the visibility detection strategy by disabling the visibility-aware masking and applying the same decay function uniformly to all regions, reported as “w/o Vis Detect”. The ablation results consistently validate the effectiveness of our design choices. Both components contribute notably to improving geometric stability and photometric consistency, and their combined effect leads to substantial gains in dynamic novel view synthesis quality.

Variants Decaying Functions. To further demonstrate that our proposed Neural Decaying Function, which is learned during training, provides an optimal opacity decay scheme, we conduct additional ablation studies by replacing it with several alternative designs. The results are shown in Tab. 5. The most naive baseline is a constant opacity decay, as used in [11]. We adopt the same decay coefficient of 0.9 and report the result as “Constant”. We also explore several manually designed mathematical functions that compute the decay coefficient based on the current opacity. For example, (1) we define a hand-tuned exponential function and report the result as “Exp Decay”, and (2) we define a hand-tuned power function and report the result as “Pow Decay”. The ablation results show that the learnable Neural Decaying Function consistently outperforms all manually crafted alternatives, even those requiring extensive effort to tune, demonstrating its ability to automatically learn an optimal opacity decay strategy.

5. Conclusion

We introduce 4C4D, a novel framework that achieves high-fidelity 4D Gaussian Splatting from videos captured by as few as four portable cameras. The core idea of 4C4D is a Neural Decaying Function that improves geometric modeling by learning to decay Gaussian opacities adaptively. By directing the 4DGS gradients to focus more on geometry, this mechanism alleviates the inherent imbalance between geometry and appearance learning in 4DGS and results in less biased optimization. We conduct comprehensive evaluations on diverse sparse-view dynamic scene datasets with varying camera overlaps and demonstrate that 4C4D consistently outperforms state-of-the-art methods.

6. Acknowledgment

This work was partially supported by Deep Earth Probe and Mineral Resources Exploration – National Science and Technology Major Project (2024ZD1003405), and the National Natural Science Foundation of China (62272263, 625B2102), and in part by Kuaishou. Junsheng Zhou is also partially funded by China Association for Science and Technology Young Talents Support Project (Doctoral Program) and Baidu Scholarship.

References

- [1] Jeongmin Bae, Seoha Kim, Youngsik Yun, Hahyun Lee, Gun Bang, and Youngjung Uh. Per-gaussian embedding-based deformation for deformable 3d gaussian splatting. In *European Conference on Computer Vision*, pages 321–335. Springer, 2024. 3
- [2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 2
- [3] Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2023. 3
- [4] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*, pages 333–350. Springer, 2022. 2
- [5] 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 *European Conference on Computer Vision*, pages 370–386. Springer, 2024. 2
- [6] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5501–5510, 2022. 2
- [7] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023. 3
- [8] Quankai Gao, Qiangeng Xu, Zhe Cao, Ben Mildenhall, Wenchao Ma, Le Chen, Danhang Tang, and Ulrich Neumann. Gaussianflow: Splatting gaussian dynamics for 4d content creation. *arXiv preprint arXiv:2403.12365*, 2024. 3
- [9] Zhongpai Gao, Benjamin Planche, Meng Zheng, Anwesa Choudhuri, Terrence Chen, and Ziyan Wu. 7dgs: Unified spatial-temporal-angular gaussian splatting. *arXiv preprint arXiv:2503.07946*, 2025.
- [10] Zhiyang Guo, Wengang Zhou, Li Li, Min Wang, and Houqiang Li. Motion-aware 3d gaussian splatting for efficient dynamic scene reconstruction. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 3
- [11] Liang Han, Junsheng Zhou, Yu-Shen Liu, and Zhizhong Han. Binocular-guided 3d gaussian splatting with view consistency for sparse view synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2, 5, 8
- [12] Xianglong He, Junyi Chen, Sida Peng, Di Huang, Yangguang Li, Xiaoshui Huang, Chun Yuan, Wanli Ouyang, and Tong He. Gvgen: Text-to-3d generation with volumetric representation. *arXiv preprint arXiv:2403.12957*, 2024. 2
- [13] Han Huang, Yulun Wu, Junsheng Zhou, Ge Gao, Ming Gu, and Yu-Shen Liu. NeuSurf: On-surface priors for neural surface reconstruction from sparse input views. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. 2
- [14] Yudong Jin, Sida Peng, Xuan Wang, Tao Xie, Zhen Xu, Yifan Yang, Yujun Shen, Hujun Bao, and Xiaowei Zhou. Difuman4d: 4d consistent human view synthesis from sparse-view videos with spatio-temporal diffusion models. *arXiv preprint arXiv:2507.13344*, 2025. 3
- [15] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023. 2
- [16] Junoh Lee, ChangYeon Won, Hyunjun Jung, Inhwan Bae, and Hae-Gon Jeon. Fully explicit dynamic gaussian splatting. *Advances in Neural Information Processing Systems*, 37:5384–5409, 2024. 3, 6
- [17] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5521–5531, 2022. 3, 5
- [18] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8508–8520, 2024. 3, 5, 6
- [19] Yiqing Liang, Numair Khan, Zhengqin Li, Thu Nguyen-Phuoc, Douglas Lanman, James Tompkin, and Lei Xiao. Gafre: Gaussian deformation fields for real-time dynamic novel view synthesis. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2642–2652. IEEE, 2025. 3
- [20] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 5
- [21] Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21136–21145, 2024. 3
- [22] Qingming Liu, Yuan Liu, Jiepeng Wang, Xianqiang Lv, Peng Wang, Wenping Wang, and Junhui Hou. Modgs: Dynamic gaussian splatting from causally-captured monocular videos. *arXiv e-prints*, pages arXiv–2406, 2024.
- [23] Zhicheng Lu, Xiang Guo, Le Hui, Tianrui Chen, Min Yang, Xiao Tang, Feng Zhu, and Yuchao Dai. 3d geometry-aware

- deformable gaussian splatting for dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8900–8910, 2024.
- [24] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pages 800–809. IEEE, 2024. 3
- [25] B Mildenhall, PP Srinivasan, M Tancik, JT Barron, R Ramamoorthi, and R Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, 2020. 2
- [26] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multi-resolution hash encoding. *ACM Transactions on Graphics (TOG)*, 41(4):1–15, 2022. 2
- [27] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 2
- [28] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5865–5874, 2021. 3
- [29] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021.
- [30] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10318–10327, 2021. 3
- [31] Richard Shaw, Michal Nazarczuk, Jifei Song, Arthur Moreau, Sibi Catley-Chandar, Helisa Dharmo, and Eduardo Pérez-Pellitero. Swings: sliding windows for dynamic 3d gaussian splatting. In *European Conference on Computer Vision*, pages 37–54. Springer, 2024. 3
- [32] Ruoxi Shi, Xinyue Wei, Cheng Wang, and Hao Su. Zerorf: Fast sparse view 360deg reconstruction with zero pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21114–21124, 2024. 2
- [33] Matthew Tancik, Vincent Casser, Xinchun Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-NeRF: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8258, 2022. 2
- [34] Jiayang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 2
- [35] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4190–4200, 2023. 2
- [36] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12922–12931, 2022. 2
- [37] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems*, 34:27171–27183, 2021. 2
- [38] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. HF-NeuS: Improved surface reconstruction using high-frequency details. *Advances in Neural Information Processing Systems*, 2022. 2
- [39] Yifan Wang, Peishan Yang, Zhen Xu, Jiaming Sun, Zhanhua Zhang, Yong Chen, Hujun Bao, Sida Peng, and Xiaowei Zhou. Freetimeg: Free gaussian primitives at anytime anywhere for dynamic scene reconstruction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21750–21760, 2025. 3, 5
- [40] Zihan Wang, Jeff Tan, Tarasha Khurana, Neehar Peri, and Deva Ramanan. Monofusion: Sparse-view 4d reconstruction via monocular fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8252–8263, 2025. 3
- [41] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20310–20320, 2024. 1, 2, 3, 6
- [42] Zhen Xu, Sida Peng, Haotong Lin, Guangzhao He, Jiaming Sun, Yujun Shen, Hujun Bao, and Xiaowei Zhou. 4k4d: Real-time 4d view synthesis at 4k resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20029–20040, 2024. 5
- [43] Zhen Xu, Yinghao Xu, Zhiyuan Yu, Sida Peng, Jiaming Sun, Hujun Bao, and Xiaowei Zhou. Representing long volumetric video with temporal gaussian hierarchy. *ACM Transactions on Graphics (TOG)*, 43(6):1–18, 2024. 3
- [44] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023. 1, 2, 3, 5, 6
- [45] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20331–20341, 2024. 2, 3
- [46] Yuxuan Yao, Zixuan Zeng, Chun Gu, Xiatian Zhu, and Li Zhang. Reflective gaussian splatting. *arXiv preprint arXiv:2412.19282*, 2024. 2
- [47] Keyang Ye, Qiming Hou, and Kun Zhou. 3d gaussian splatting with deferred reflection. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–10, 2024.

- [48] Wenyuan Zhang, Jimin Tang, Weiqi Zhang, Yi Fang, Yu-Shen Liu, and Zhizhong Han. Materialrefgs: Reflective gaussian splatting with multi-view consistent material inference. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. [2](#)
- [49] Weiqi Zhang, Junsheng Zhou, Haotian Geng, Wenyuan Zhang, and Yu-Shen Liu. Gap: Gaussianize any point clouds with text guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 25627–25638, 2025. [2](#)
- [50] Junsheng Zhou, Baorui Ma, and Yu-Shen Liu. Fast learning of signed distance functions from noisy point clouds via noise to noise mapping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. [2](#)
- [51] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3D: Exploring unified 3d representation at scale. *International Conference on Learning Representations*, 2024. [2](#)
- [52] Junsheng Zhou, Weiqi Zhang, and Yu-Shen Liu. Diffgs: Functional gaussian splatting diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. [2](#)
- [53] Ruijie Zhu, Yanzhe Liang, Hanzhi Chang, Jiacheng Deng, Jiahao Lu, Wenfei Yang, Tianzhu Zhang, and Yongdong Zhang. Motiongs: Exploring explicit motion guidance for deformable 3d gaussian splatting. *Advances in Neural Information Processing Systems*, 37:101790–101817, 2024. [3](#)
- [54] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *European conference on computer vision*, pages 145–163. Springer, 2024. [2](#)