

# Supplementary Materials

## A. Overview

In the supplementary material, we present additional implementation details in Sec. B, extended ablation studies in Sec. C, more detailed results in Sec. D, and a discussion of limitations and future work in Sec. E. We also include a supplementary video that visually compares MoRGS with prior methods and further illustrates our reconstruction quality and motion behavior.

## B. More Implementation Details

### B.1. Datasets

Neural 3D Video (N3DV) [6] consists of six indoor dynamic scenes captured as forward-facing multi-view videos with up to 20 cameras at a resolution of  $2704 \times 2028$ . Following prior work, we downsample all videos by a factor of 2 for both training and testing. The Meet Room dataset [5] contains three dynamic scenes recorded with 13 cameras at a resolution of  $1280 \times 720$ . In line with previous methods, we undistort all views using the provided distortion parameters to obtain perspective images and improve reconstruction quality. For all datasets the central view is held-out for test view.

### B.2. Training

We build MoRGS on top of the open-source 3D Gaussian Splatting (3DGS) codebase [4]. For initial-frame training on all datasets, we set the spherical harmonic (SH) degree to 2 and stop 3DGS densification at 8,000 iterations to avoid overfitting in the streaming setting. For N3DV, we compute optical flow [8] using camera views (2, 5, 9, 14), and for Meet Room we use views (2, 5, 8, 11). The per-Gaussian motion confidence is trained only during the last 2,000 iterations of the initial-frame optimization, using a binary motion mask obtained by thresholding optical flow with  $\lambda^{\text{flow}} = 0.5$ . Its learning rate is set to 0.01, matching that of the opacity attribute.

For each subsequent frame, we perform 8 optimization iterations per view to jointly minimize the reconstruction loss  $\mathcal{L}_{\text{recon}}$  and the per-Gaussian motion losses  $\mathcal{L}_{\text{flow}}$  and  $\mathcal{L}_{\text{off}}$ , with  $\lambda_{\text{flow}} = 0.01$  and  $\lambda_{\text{off}} = 0.1$ . The per-Gaussian motion confidence is further refined by applying the mask loss  $\mathcal{L}_{\text{mask}}$  every 5 frames with  $\lambda_{\text{mask}} = 0.1$ . We perform

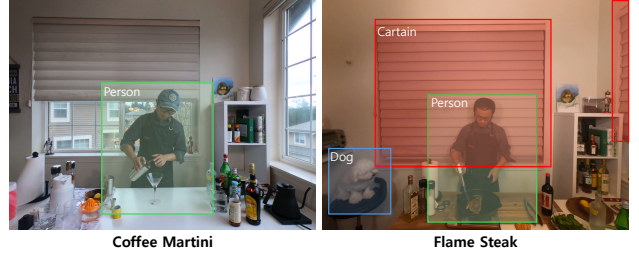


Figure 1. Bounding boxes of the dynamic regions in the coffee martini and flame steak in N3DV dataset.

Method	Cut Roasted Beef				Flame Steak			
	PSNR (dB)↑	GS Num (k)↓	Storage (MB)↓	Train (Sec)↓	PSNR (dB)↑	GS Num (k)↓	Storage (MB)↓	Train (Sec)↓
3DGStream	33.95	294	8	120	33.85	208	8	115
QUEEN	33.90	247	13	59	34.15	249	14	57
<b>Ours</b>	33.86	258	15	79	33.60	242	16	80

Table 1. **Initial 3DGS comparison on N3DV.** Average first-frame reconstruction quality, number of Gaussian, storage and training time of 3DGStream and QUEEN across two scenes under their respective initialization strategies.

densification for subsequent frames up to 80% of the total epochs, using a gradient threshold of 0.002 for both datasets.

### B.3. Total Variation in Static Regions

We manually define bounding boxes around dynamic regions to create masks and measure total variation only in static areas, as shown in Fig. 1.

## C. More Ablation Studies

### C.1. Initial Frame Reconstruction Comparison

For streaming 3DGS reconstruction frameworks, initial-frame training plays a pivotal role in determining the reconstruction quality of subsequent frames. Tab. 1 compares first-frame reconstruction quality, Gaussian count, storage, and training time across two N3DV scenes. 3DGStream [7] fixes the SH degree to 1 to reduce memory and rendering overhead, whereas QUEEN [2] leverages depth-map-based initialization to place denser Gaussians in empty regions, providing a stronger static prior at the first frame. In con-

# Views	Optical Flow Time (s)	SAM2 Time (s)	Train Time (s)	Total Time(s)	PSNR(dB)
2	0.16	0.45	3.40	3.65	31.93
4	0.42	0.90	3.40	4.00	32.53
8	0.93	1.47	3.40	4.63	32.54
12	2.62	1.77	3.40	6.38	32.57

Table 2. Average runtime per frame for optical flow estimation and MoRGS training, and per keyframe for SAM2-based segmentation, with total average runtime and PSNR, across varying numbers of motion supervision views on N3DV.

Keyframe Interval	1	3	5	8	10
PSNR(dB)	32.51	32.50	32.53	32.48	32.44
Time(s)	4.72	4.12	4.00	3.93	3.91

Table 3. PSNR and per-frame training time under varying keyframe intervals with four motion supervision views on N3DV.

trast, our framework adopts an almost vanilla 3DGS initialization, changing only the SH degree to 2 and leaving all other settings unchanged. Overall, all three methods achieve comparable first-frame PSNR (within 0.55 dB), and the main differences lie in how they trade off storage and optimization cost. 3DGStream [7] attains the smallest storage footprint due to its SH degree 1 representation but requires the longest initial optimization, while QUEEN benefits from depth-based initialization to achieve the fastest first-frame training under a comparable storage budget. Our near-vanilla 3DGS initialization with SH degree 2 lies between these two in terms of Gaussian count and training time, with comparable initial-frame quality, indicating that the gains observed in later frames primarily stem from our motion reasoning rather than from a more favorable static initialization.

## C.2. Training Time for Motion Learning

While Tabs. 4 and 5 in the main paper analyze the accuracy–efficiency trade-off of motion learning, here we focus on the efficiency side and break down the training time overhead of MoRGS. Tab. 2 reports the average per-frame time spent on optical flow estimation, SAM2-based motion segmentation, and 3DGS training for different numbers of motion supervision views. For each frame, optical flow is computed on the selected supervision views, whereas SAM2 is applied only to keyframes at an interval of 5 (60 keyframes out of a total of 300 frames), and its cost is averaged over all frames when computing the per-frame averages training time.

As the number of supervision views increases, the time spent on both optical flow and SAM2 grows, while the core 3DGS training time remains constant at 3.40 s per frame. Increasing the views from 2 to 4 raises the total per-frame time only moderately from 3.65 s to 4.00 s, but further increasing to 8 and 12 views pushes it to 4.63 s and 6.38 s, respectively. This overhead is critical in a streamable set-

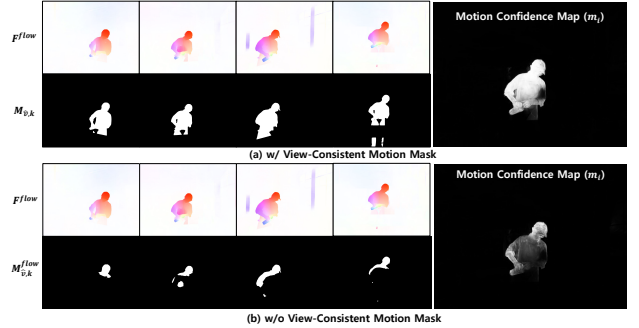


Figure 2. **Effect of View-Consistent Mask** A visualization of per-Gaussian motion confidence map with and without view-consistent motion map.

ting, where per-frame latency directly limits how fast new frames can be reconstructed and transmitted. In practice, we fix the number of supervision views to 4, which provides sufficiently strong motion cues while adding only modest overhead on top of base 3DGS training, enabling effective motion learning without compromising the practicality of online reconstruction.

Tab. 3 further analyzes the effect of varying the keyframe interval while fixing the number of supervision views to 4. As the interval decreases, the per-frame training time increases due to the higher frequency of SAM2-based supervision. Conversely, increasing the interval reduces training overhead, but may miss newly emerging dynamic objects or rapid motion changes, leading to potential degradation in motion estimation. Nevertheless, within a moderate range, PSNR remains largely stable, suggesting that sparse keyframe supervision provides a favorable trade-off between efficiency and performance.

## C.3. View-Consistent Motion Mask

Learning the per-Gaussian motion confidence  $m_i$  crucially depends on providing consistent motion labels across all camera views that observe the same 3D object. As illustrated in Fig. 2(a), only when the binary motion masks are view-consistent can the optimizer reliably assign high confidence to truly moving Gaussians while keeping static ones near zero. In contrast, Fig. 2(b) shows that if a region is labeled as moving in some views but static in others, the conflicting supervision drives the learned confidence toward



Figure 3. **The Visualization of Gaussian Distribution with High Per-Gaussian Motion Confidence** The orange points represent the Gaussian with high motion confidence during optimization.

Method	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	Train (s) $\downarrow$
w/ $M_{\hat{v},k}^{\text{flow}}$	32.35	0.949	0.119	3.82
w/ $M_{\hat{v},k}$	32.53	0.950	0.118	4.0

Table 4. Ablation on view-consistent mask  $M_{\hat{v},k}$  on N3DV.

$\lambda^{\text{flow}}$	0.1	0.5	1.0	1.5
w/ $M_{\hat{v},k}^{\text{flow}}$	31.82	32.35	31.97	30.45
w/ $M_{\hat{v},k}$	32.10	32.53	32.38	31.11

Table 5. Ablation on motion threshold  $\lambda^{\text{flow}}$  on N3DV.

artificially low values. As a result, motion updates on genuinely dynamic Gaussians are heavily down-weighted, so their motion is poorly captured and the model instead relies on per-frame appearance changes to explain the dynamics. In an online, streamable setting where past frames cannot be revisited, such mis-estimated confidence accumulates over time and leads to degraded reconstruction quality.

Tab. 4 compares training with and without view-consistent mask  $M_{\hat{v},k}$ . Compared to using optical-flow-based masks alone, aggregating SAM2-refined masks across views produces a view-consistent motion mask that supervises the per-Gaussian motion confidence  $m_i$ . This consistent supervision enables  $m_i$  to reliably weight Gaussian attribute residuals, assigning high confidence to truly dynamic Gaussians while keeping static ones near zero, which improves both reconstruction quality and temporal stability.

We further analyze the sensitivity to the threshold parameter  $\lambda^{\text{flow}}$  used for motion mask supervision. Tab. 5 compares different  $\lambda^{\text{flow}}$  values on N3DV for supervising the motion confidence  $m_i$  using both the view-dependent mask  $M_{\hat{v},k}^{\text{flow}}$  and the proposed view-consistent mask  $M_{\hat{v},k}$ . The

view-dependent mask exhibits strong sensitivity to  $\lambda^{\text{flow}}$ , where low thresholds introduce noisy motion regions and high thresholds miss dynamic content, leading to degraded performance. In contrast, the view-consistent mask yields stable performance across a wide range of thresholds by enforcing consistent supervision across views. This robustness arises because our mask fusion resolves view-dependent inconsistencies rather than simply improving mask coverage.

We additionally visualize Gaussians with high motion confidence in Fig. 3, where high-confidence Gaussians (orange) concentrate on genuinely moving objects, causing gradient updates to focus on these regions while being strongly suppressed in the background.

## D. More Detailed Results

### D.1. Per-Scene Results

In addition to the average quantitative results over the full N3DV and Meet Room datasets, we also report per-scene results in Tabs. 6 and 7. For each scene, we provide frame-wise metrics, including PSNR, SSIM, LPIPS, and training time.

### D.2. Qualitative Results

Figs. 4 and 5 provide additional visualizations of our per-Gaussian motion map and the confidence map rendered from the test view.

Figs. 6 and 7 present extended qualitative comparisons on the N3DV and Meet Room datasets, comparing our method against 3DGStream [7] and QUEEN [2] and highlighting key differences in dynamic scene reconstruction.

## E. Limitation and Future Works

Despite the improvements brought by MoRGS, our approach still shares several structural limitations with existing online 3DGS pipelines. First, the reconstruction quality remains sensitive to the quality of the initial 3DGS representation built from the first frame. Second, as with other one-pass online methods, MoRGS cannot go back and correct earlier frames, so residual errors may gradually accumulate over long sequences or under drastic scene changes, which can occasionally degrade reconstruction quality. Handling very long videos with large geometric or appearance changes therefore remains challenging and is an interesting direction for future work.

Furthermore, our framework leverages 2D optical flow as the primary motion cue, so the learned 3D motion can still be affected by the accuracy and robustness of optical flow estimators. In low-texture regions, under motion blur, illumination changes, or near occlusion boundaries, the estimated flow may become noisy or biased, which can lead to imperfect supervision for per-Gaussian motion guidance and confidence. Given that an online approach must update motion and appearance on-the-fly to faithfully capture real scene dynamics, exploring motion cues that further strengthen the coupling between 3D motion estimation and appearance updates is a promising direction for future work.



Method	Coffee Martini			Cook Spinach			Cut Roasted Beef		
	PSNR↑	SSIM↑	Training↓	PSNR↑	SSIM↑	Training↓	PSNR↑	SSIM↑	Training↓
3DGStream [7]	27.75	-	15.40	33.31	-	12.40	33.21	-	12.50
HiCoM [1]	28.04	-	10.50	32.45	-	11.00	32.72	-	11.50
QUEEN-1 [2]	<b>28.38</b>	<b>0.915</b>	<b>2.71</b>	<b>33.40</b>	<b>0.956</b>	<b>3.11</b>	<b>34.01</b>	<b>0.959</b>	<b>2.76</b>
4DGC [3]	27.89	-	59.5	32.81	-	46.8	33.03	-	48.4
<b>Ours</b>	<b>29.01</b>	<b>0.923</b>	<b>3.82</b>	<b>33.90</b>	<b>0.960</b>	<b>4.61</b>	<b>33.66</b>	<b>0.959</b>	<b>3.79</b>

Method	Flame Salmon			Flame Steak			Sear Steak		
	PSNR↑	SSIM↑	Storage↓	PSNR↑	SSIM↑	Storage↓	PSNR↑	SSIM↑	Storage↓
3DGStream [7]	28.42	-	12.10	<b>34.30</b>	-	11.20	33.01	-	14.80
HiCoM [1]	28.37	-	10.80	32.87	-	11.20	32.57	-	11.00
QUEEN-1 [2]	<b>29.25</b>	<b>0.923</b>	<b>3.09</b>	<b>34.17</b>	<b>0.962</b>	<b>3.04</b>	<b>33.93</b>	<b>0.960</b>	<b>3.07</b>
4DGC [3]	28.49	-	44.8	33.58	-	41.5	33.60	-	59
<b>Ours</b>	<b>30.11</b>	<b>0.931</b>	<b>3.99</b>	<b>34.17</b>	<b>0.964</b>	<b>4.01</b>	<b>34.36</b>	<b>0.967</b>	<b>3.78</b>

Table 6. Per-Scene Quantitative Results on the N3DV Dataset.

Method	Discussion			Trimming			Vrheadset		
	PSNR↑	SSIM↑	Storage↓	PSNR↑	SSIM↑	Storage↓	PSNR↑	SSIM↑	Storage↓
3DGStream [7]	30.55	-	6.97	<b>31.87</b>	-	6.97	<b>32.75</b>	-	7.68
HiCoM [1]	26.39	-	10.8	26.38	-	10.9	25.98	-	9.5
QUEEN-1 <sup>†</sup> [2]	<b>30.76</b>	<b>0.951</b>	<b>1.56</b>	29.98	<b>0.947</b>	<b>1.48</b>	27.67	<b>0.942</b>	<b>1.48</b>
<b>Ours</b>	<b>31.37</b>	<b>0.958</b>	<b>2.32</b>	<b>32.36</b>	<b>0.959</b>	<b>2.29</b>	<b>31.65</b>	<b>0.954</b>	<b>2.30</b>

Table 7. Per-Scene Quantitative Results on the Meet Room Dataset. Methods with <sup>†</sup> are reproduced using the official code in the same experimental environment.

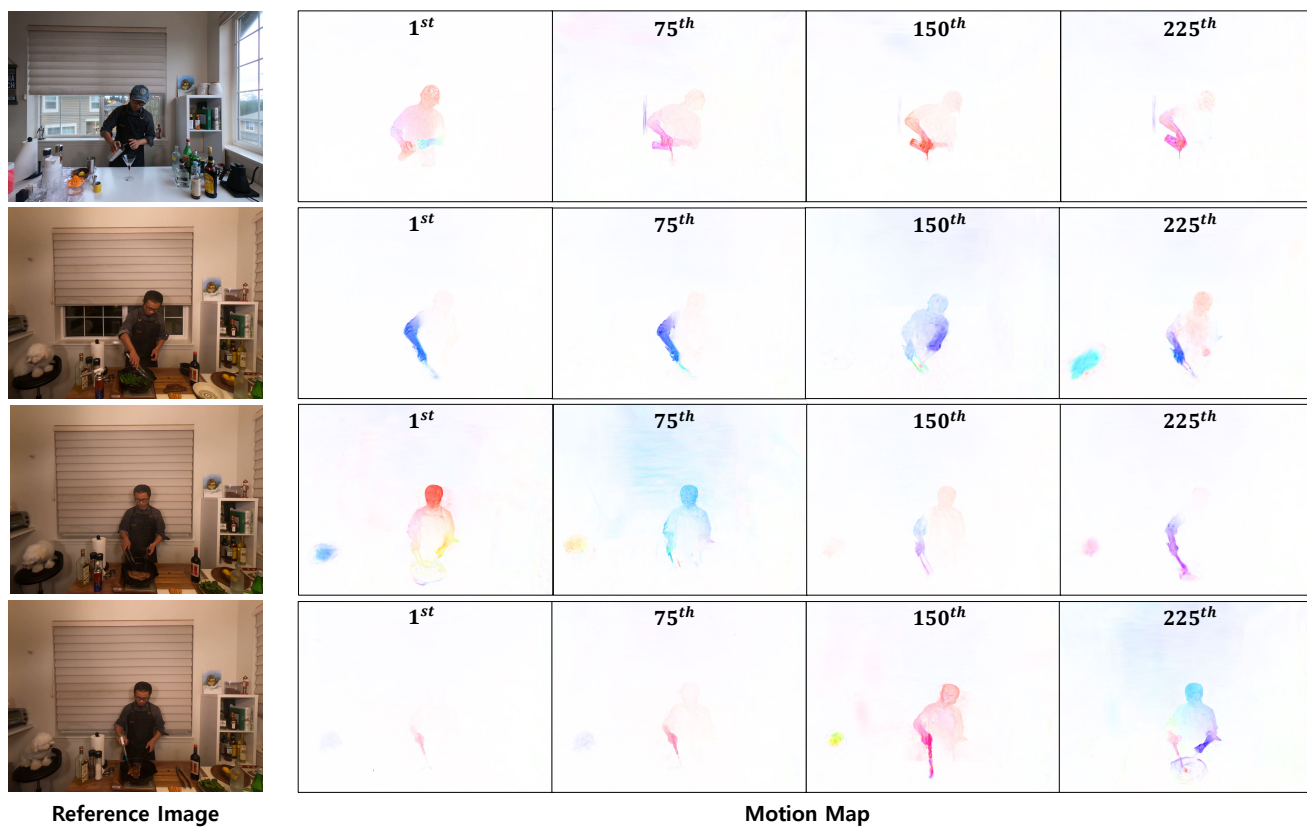


Figure 4. The Visualization of Per-Gaussian Motion Rendered from Test View.

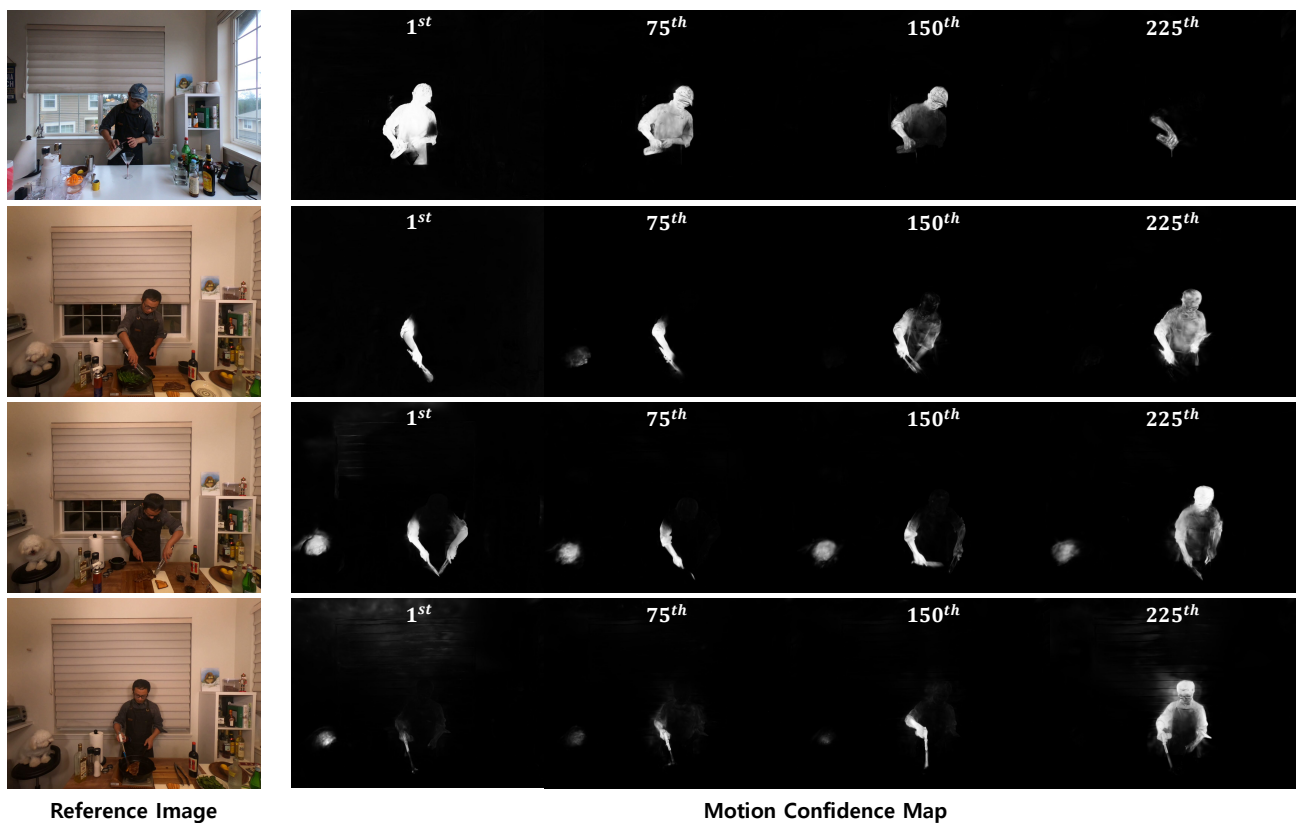


Figure 5. The Visualization of Per-Gaussian Motion Confidence Rendered from Test View.

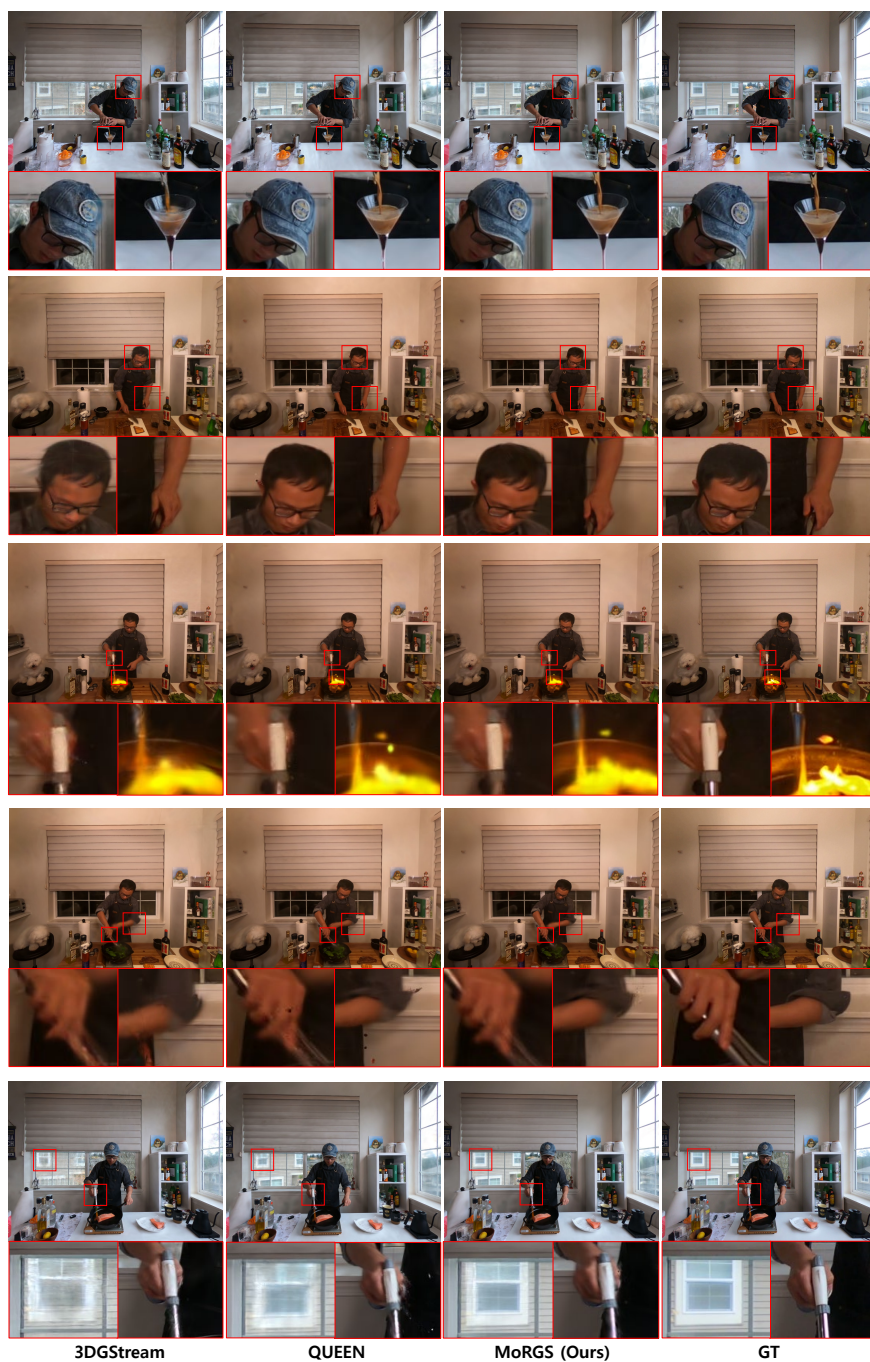


Figure 6. Additional Qualitative Results on N3DV Dataset.





Figure 7. Additional Qualitative Results on Meet Room Dataset.



## References

- [1] Qiankun Gao, Jiarui Meng, Chengxiang Wen, Jie Chen, and Jian Zhang. HiCoM: Hierarchical coherent motion for dynamic streamable scenes with 3d gaussian splatting. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2024. [5](#)
- [2] Sharath Girish, Tianye Li, Amrita Mazumdar, Abhinav Shrivastava, Shalini De Mello, et al. QUEEN: QUantized Efficient ENcoding of Dynamic Gaussians for Streaming Free-viewpoint Videos. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2024. [1](#), [3](#), [5](#)
- [3] Qiang Hu, Zihan Zheng, Houqiang Zhong, Sihua Fu, Li Song, Guangtao Zhai, Yanfeng Wang, et al. 4DGC: Rate-Aware 4D Gaussian Compression for Efficient Streamable Free-Viewpoint Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. [5](#)
- [4] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics (TOG)*, 2023. [1](#)
- [5] Lingzhi Li, Zhen Shen, Zhongshu Wang, Li Shen, and Ping Tan. Streaming Radiance Fields for 3D Video Synthesis. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2022. [1](#)
- [6] 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 (CVPR)*, 2022. [1](#)
- [7] Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 3DGStream: On-the-fly Training of 3D Gaussians for Efficient Streaming of Photo-Realistic Free-Viewpoint Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. [1](#), [2](#), [3](#), [5](#)
- [8] Yihan Wang, Lahav Lipson, and Jia Deng. Sea-raft: Simple, efficient, accurate raft for optical flow. In *European Conference on Computer Vision*, pages 36–54. Springer, 2024. [1](#)