4D LangSplat: 4D Language Gaussian Splatting via Multimodal Large Language Models

Supplementary Material

A. Datasets

Since there are no publicly available ground truth segmentation mask labels for the HyperNeRF [7] and Neu3D [6] datasets, nor annotations tailored for time-sensitive querying, we adopt the annotation pipeline outlined in Segment Any 4D Gaussians [3] and manually annotate the mask labels ourselves. Specifically, we leverage the Roboflow platform alongside the SAM (Segment Anything Model) framework for interactive annotation.

For the HyperNeRF dataset, where data is captured with a monocular camera, we select one frame every four frames as the training set. From the remaining data, we annotate a subset as the test set to ensure no overlap between the two sets. For the Neu3D dataset with 21 camera views, one is reserved for testing, and the remaining 20 are used for training, aligning with the 4D-GS [10] setting. To evaluate on the Neu3D dataset, we annotate every 20 frames from the test views.

B. Implementation Details

Multimodal Object-Wise Video Prompting. For Multimodal Object-Wise Video Prompting, we utilize the largest SAM-defined semantic levels as mask inputs for the Multimodal Large Language Models (MLLMs). The prompting process is outlined in Table 1, which provides the specific prompts used for MLLM prompting. For visual prompting, we employ a red contour line with a radius of 2 to delineate object boundaries. Additionally, we apply Gaussian blur with a radius of 10 and convert the images to grayscale mode to achieve gray-level augmentation. These techniques enhance the effectiveness of the visual input during the prompting process.

Autoencoder. Following LangSplat [8], we employ two autoencoders to compress the high-dimensional CLIP feature (512-dimension) and LLM feature (4096-dimension) separately. Specifically, two MLPs are used to compress 512-dimensional CLIP features and 4096-dimensional video features to 3 and 6 dimensions, respectively. The autoencoders are optimized with L2 loss. To enhance stability, a cosine similarity loss is also included as a regularization.

Training Details. Our training pipeline is structured into four stages, progressively refining the model for robust performance in dynamic 4D language field construction. 1) In the initial stage, we train a static Gaussian field to reconstruct the RGB channel of static scenes. This provides a foundation for modeling the visual appearance of the scene. 2) Next, we incorporate semantic information into the static Gaussian field without introducing deformable networks. Semantic features are embedded into the scene by minimizing an L_1 loss, ensuring accurate representations of the static scene's semantics. 3) In the third stage, we extend the model to dynamic RGB scenes by introducing non-semantic deformation fields. Leveraging the approach of 4D-GS [10], we employ deformable networks to learn temporal and motion-based deformations that capture spatial and temporal dynamics for RGB scenes. 4) For time-agnostic semantic rendering, we refine the semantic features

Video prompts	Image prompts
I highlighted the objects I want you to describe in red outline and blurred the objects that don't need you to describe. First please determine the object highlighted in red line in the video. Then briefly summarize the transformation process of this object.	You have an understanding of the overall transformation process of the object: {video prompt}. Now, I have provided you with images extracted from this process. Please describe the specific state of the object(s) in the given image, without referring to the entire video process. Avoid describing states that you can't infer directly from the picture. Avoid repeating descriptions in context. For example, if the context suggests the object is moving up and down but the image shows it is just moving down, explicitly only state that the object is in a moving down state. If the context suggests the object is breaking but the image shows it is complete right now, explicitly only state that the object appears to be complete. If context tells you something changes from green to blue, but it's blue in this image, just state that the object is blue.

Table 1. Details of Text prompts

Method	FPS
Gaussian Grouping [11]	1.47
Ours-agnostic	**5.24**
Ours-sensitive	4.05

Table 2. Query Performance Comparison.

Method	americano		chickchicken		split-cookie	
	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)
Feature-3DGS [12]	34.65	62.96	47.21	87.22	47.03	68.25
Gaussian Grouping [11]	61.77	71.31	34.65	75.52	72.71	96.56
LangSplat [8]	72.08	97.61	75.98	97.86	76.54	97.32
Ours	**83.48**	**98.77**	**86.50**	**98.81**	**90.04**	**98.67**
Method	espresso		keyboard		torchocolate	
	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)
Feature-3DGS [12]	24.04	80.13	42.14	80.98	24.71	64.58
Gaussian Grouping [11]	32.45	82.46	42.44	74.15	58.95	85.52
LangSplat [8]	82.93	**98.66**	72.42	**96.75**	69.55	98.09
Ours	**83.52**	97.95	**79.53**	95.71	**71.79**	**98.10**

Table 3. Comparison of mean IoU and mean Accuracy for various methods on the HyperNeRF [7] datasets.

Method	coffee martini		cook spinach		cut roasted beef	
	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)
Feature-3DGS [12]	30.23	84.74	41.50	95.59	31.66	91.07
Gaussian Grouping [11]	71.37	97.34	46.45	93.79	54.70	93.25
LangSplat [8]	67.97	98.47	78.29	98.60	36.53	97.04
Ours	**85.16**	**99.23**	**85.09**	**99.38**	**85.32**	**99.28**
Method	flame salmon		flame steak		sear steak	
	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)	mIoU(%)	mAcc(%)
Feature-3DGS [12]	54.33	77.13	27.27	88.23	24.78	85.94
Gaussian Grouping [11]	35.72	**94.69**	36.92	95.96	54.44	95.27
LangSplat [8]	66.01	82.16	64.05	97.77	**78.29**	98.60
Ours	**89.88**	94.35	**88.44**	**98.27**	76.78	**99.38**

Table 4. Comparison of mean IoU and mean Accuracy for various methods on the Neu3D [6] dataset.

from the second stage while keeping the deformable network parameters fixed. For time-sensitive semantic rendering, we jointly train the status deformable network and the state prototype features to refine and model dynamic semantics effectively. For all datasets, the iterations for four stages are 3000, 1000, 10000, and 10000. The learning rates for the deformable network and the state prototype features are set to 1.6×10^{-4} and 2.5×10^{-3}, respectively. Other training parameters remain consistent with those used in 4D-GS.

C. More Quantitative Results

In Table 3 and Table 4, we present a detailed evaluation of time-agnostic querying performance on the HyperNeRF and Neu3D datasets, respectively. Our method achieves a mean IoU exceeding 85% across all scenarios, outperforming the baseline methods in most scenes for both mean IoU and mean accuracy. These results underscore the robustness of our approach, demonstrating its ability to deliver superior segmentation accuracy and reliability compared to existing methods, even in dynamic scenes.

Table 2 further compares the runtime efficiency of our method with the baseline on the HyperNeRF dataset. The comparison encompasses the total time required for rendering semantic features and conducting open-vocabulary queries. Our method demonstrates significant advantages over the Gaussian Grouping approach, achieving faster runtime for both time-agnostic and time-sensitive queries. These findings validate our method as an efficient and scalable solution for handling open-vocabulary queries in dynamic 4D scenes.

D. More Visualization Results

Figure 1 illustrates visualization results for time-agnostic querying. As depicted, our method demonstrates superior accuracy in capturing objects that correspond to semantic descriptions, compared to other methods. Furthermore,

Figure 1. Visualization of time-agnostic querying results on HyperNeRF [7] and Neu3D [6] datasets.

it effectively tracks the spatial dynamics of these objects across different temporal steps, showcasing its effectiveness in handling dynamic scenarios.

E. MLLM-based Embeddings

Since our method utilizes MLLMs to generate captions, the feature representation capability of the obtained embeddings is inherently limited by the capacity of the MLLMs, which constitutes a limitation of our approach. To verify that our MLLM-based embeddings indeed encode spatial-temporal information, we directly apply the MLLM-based embeddings, without any fine-tuning, to video classification and spatial-temporal action localization tasks using 2D videos. As shown in Tables 5 and 6, our results demonstrate that, even in a zero-shot setting, the MLLM-based embeddings achieve competitive performance compared to state-of-the-art (SOTA) methods specifically designed for these tasks. This indicates that MLLM-based embeddings inherently capture some spatial-temporal information. However, we also acknowledge that the performance of our approach is ultimately constrained by the representational capacity of the MLLMs.

Method	HMDB51 [5]	UCF101 [9]	Kinetics400 [4]
MLLM	58.34	78.97	55.14
IMP [1]	59.1	91.5	77.0

Table 5. Accuracy Results (%) on the Video Classification task.

Method	VmAP@0.1	VmAP@0.2	VmAP@0.5
MLLM	78.13	75.78	64.38
HIT [2]	86.1	88.8	74.3

Table 6. Spatial-Temporal Action Localization Results (%) on UCF101 [9].

References

[1] Hassan Akbari, Dan Kondratyuk, Yin Cui, Rachel Hornung, Huisheng Wang, and Hartwig Adam. Alternating gradient descent and mixture-of-experts for integrated multimodal perception. *NeurIPS*, 2023. 3

[2] Gueter Josmy Faure, Min-Hung Chen, and Shang-Hong Lai. Holistic interaction transformer network for action detection. In *WACV*, 2023. 3

[3] Shengxiang Ji, Guanjun Wu, Jiemin Fang, Jiazhong Cen, Taoran Yi, Wenyu Liu, Qi Tian, and Xinggang Wang. Segment any 4d gaussians. *arXiv preprint arXiv:2407.04504*, 2024. 1

[4] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 3

[5] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pages 2556–2563. IEEE, 2011. 3

[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*, pages 5521–5531, 2022. 1, 2, 3

[7] 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. 1, 2, 3

[8] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20051–20060, 2024. 1, 2

[9] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 3

[10] 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

[11] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3d scenes. In *European Conference on Computer Vision*, pages 162–179. Springer, 2025. 1, 2

[12] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21676–21685, 2024. 2