MORE-STEM: Long-Short MemOry REcall and Spatio-TEmporal Consistency Model for Query-Driven 3D/4D Point Cloud Segmentation

Supplementary Material

In this Supplementary Material, we first introduce the construction and characteristics of the InstructKITTI benchmark in Sec. I. We then provide additional dataset details in Sec. II, followed by more implementation details, including configuration settings and training specifications, in Sec. III. Furthermore, Sec. IV presents extended experimental results. Specifically, Sec. IV.1 reports additional experiments on the InstructKITTI 3D/4D benchmark, Sec. IV.2 evaluates temporal consistency, Sec. IV.3 analyzes runtime performance, Sec. IV.4 provides additional qualitative results on ScanRefer, Sec. IV.5 presents quantitative results on multi-scan semantic segmentation for SemanticKITTI, and Sec. IV.6 includes further ablation studies.

I. InstructKITTI Benchmark

We present a scalable and fully automated data generation pipeline built on the SemanticKITTI dataset to construct a novel 4D spatio-temporal and 3D-grounded benchmark for instruction-guided segmentation. The pipeline conducts instance-level temporal reasoning, computes 3D centroids and motion relations, and performs accurate 3D-to-2D projection via calibrated LiDAR-camera transformations. It then synthesizes diverse linguistic instructions through a dual-mode strategy that generates both dynamic and static queries. To enhance benchmark reliability, an additional human verification stage is incorporated to refine the automatically produced queries, ensuring linguistic fluency, removing ambiguous expressions, and guaranteeing faithful grounding between the textual instructions and their corresponding 3D masks.

We select sequences 00-10 from SemanticKITTI because they are the only sequences with publicly available ground-truth labels; the remaining sequences do not provide official annotations. Following this constraint, we annotate sequences 00-06 to form the training set, sequence 07 to form the validation set, and sequences 08-10 to form the test set. The final benchmark contains more than 15K high-quality (Query, 3D-Mask) pairs and provides a linguistically rich and geometrically precise resource for evaluating 3D and 4D instruction-based reasoning.

Our pipeline introduces a Dual-Mode Query Generation strategy, designed to disentangle complex spatio-temporal motion reasoning (Dynamic Mode) from attribute-based object identification (Static Mode). The process unfolds in three hierarchical stages:

Spatio-Temporal Geometric Analysis. The pipeline begins with an instance-level spatio-temporal analysis. For each instance i at frame t, it computes its 3D centroid C_i^t in the global coordinate system P_i^t using the ego-pose $T_{global \leftarrow ego}^t$. For the dynamic mode, this analysis extends to tracking P_i^t across consecutive frames, enabling computation of both the instance's motion vector $\vec{v}_{inst}$ and the ego-vehicle's motion vector $\vec{v}_{ego}$. It then classifies the dynamic motion relationship M_{geom} (e.g., "Same Direction", "Opposite Direction", "Different Direction") by computing the dot product of their 2D ground-plane projections, $d = \vec{v}_{inst,xy} \cdot \vec{v}_{ego,xy}$.

Cross-Modal Projection and Filtering. To bridge the 3D point cloud and 2D image domains, the pipeline projects the 3D instance mask M_{3D} onto the image plane via the camera intrinsic matrix P_I and the LiDAR-to-camera extrinsic transformation $T_{cam \leftarrow velo}$. This projection involves a filtering process: the pipeline first discards all points behind the camera plane ($z_{cam} \leq 0$), then selects the subset of projected points (u, v) that lie within valid image boundaries ($0 \leq u < W, 0 \leq v < H$). A tight 2D bounding box is computed from these valid points, producing a high-fidelity RGB crop I_{crop} that serves as the input to the vision-language model (VLM).

VLM-Driven Semantics and Fusion. The final stage leverages Qwen3VL-7B [43] to generate the query. The model first extracts open-vocabulary attributes A_{vis} (e.g., "a red sedan") from I_{crop}. Then for different modes, we use different strategies:

- **For Dynamic Mode:** To mitigate geometric noise, the pipeline introduces a Geometric-Visual Fusion mechanism. The VLM infers a visual motion cue M_{vis} (e.g., "judging by the car's rear, it is moving in the Same Direction") directly from I_{crop}. This is fused with the geometric prior to yield a robust motion label: $M_{fused} = \text{Fuse}(M_{geom}, M_{vis})$. The final query $Q_{dynamic}$ combines M_{fused} with A_{vis} (e.g., "Identify the moving red sedan traveling in the same direction").
- **For Static Mode:** This mode bypasses motion analysis, composing a simpler query Q_{static} from A_{vis} and scene context (e.g., "Identify the red sedan on the left").

In both modes, the ground-truth answer remains the original 3D instance mask M_{3D}. A final VLM call, prompted

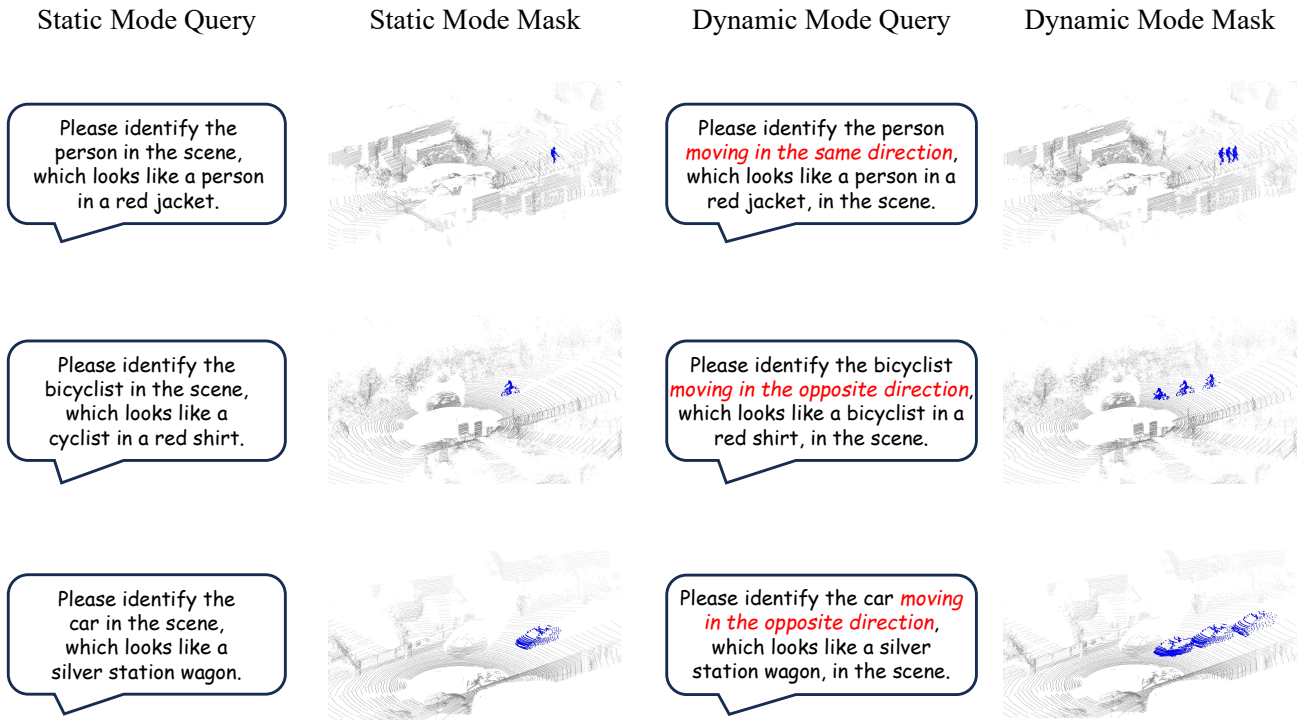

Figure I. Representative examples from the InstructKITTI benchmark. The dataset includes diverse 3D and 4D scenes with annotated objects and dynamic events, illustrating the complexity and variety of scenarios used for evaluation.

with Q and I_{crop}, produces a textual confirmation containing a symbolic [SEG] token (e.g., "The object is at [SEG]."), which links the generated query to its corresponding 3D mask.

Figure I presents a selection of representative examples from our InstructKITTI benchmark, illustrating how the dataset supports both static-mode and dynamic-mode language queries. The static mode focuses on appearance-based descriptions of single-frame LiDAR scenes, while the dynamic mode enriches the instructions with motion-related attributes such as direction consistency or directional opposition across frames. The qualitative samples demonstrate the benchmark's ability to model fine-grained motion reasoning, temporal object disambiguation, and multi-object grounding, enabling more comprehensive evaluation of instruction-driven 3D and 4D scene understanding.

II. Dataset Details

The proposed MORE-STEM is applicable to a wide range of point cloud segmentation tasks. We evaluate its effectiveness across four downstream sub-tasks: 4D instruction segmentation, 3D instruction segmentation, 3D referring segmentation, 3D semantic segmentation. These evaluations are conducted using the proposed InstructKITTI-3D/4D benchmark, Instruct3D [10], ScanRefer [4], and SemanticKITTI [3].

The data generation pipeline for the proposed InstructKITTI benchmark is elaborated in Suppl. I. This benchmark comprises >15k query-answer pairs from the SemanticKITTI dataset sequences 00-10, featuring textual query statements alongside 3D/4D instruction segmentation masks. Instruct3D [10] is a large-scale indoor benchmark built upon ScanNet++ [44]. It contains over 100,000 instruction-mask pairs across various indoor environments, each accompanied by rich natural language descriptions that refer to target objects and their spatial relationships. ScanRefer [4], constructed on the ScanNet dataset [6], is widely used for 3D referring segmentation. It includes more than 50,000 human-written expressions linked to 3D object instances across multiple indoor scenes, enabling fine-grained text-point correspondence analysis. SemanticKITTI [3] serves as the benchmark for both 3D and 4D semantic segmentation. It consists of sequential LiDAR scans captured from real-world driving scenarios, providing dense per-point semantic annotations across 28 categories and allowing temporal segmentation evaluation in outdoor environments.

In order to evaluate our query-driven 3D/4D segmentation method on 3D/4D semantic segmentation using the SemanticKITTI dataset, a standardized text query generation methodology has been implemented. The approach described here employs the following structured template:

Table I. Comprehensive comparison of recent methods on the InstructKITTI 3D/4D benchmark. Evaluation metrics include mean Intersection over Union (mIoU), Acc@50, and Acc@25, as well as inference speed and GPU memory usage, providing a complete assessment of both accuracy and efficiency. Best results are highlighted in **bold**, and next-best results are underlined.

Method	3D Acc@25	3D Acc@50	3D mIoU	4D Acc@25	4D Acc@50	4D mIoU	Latency(s)	GPU
RefMask3D [9], ACM MM 2024	14.15	12.70	12.50	17.70	16.74	16.24	0.21	16.4
3D-STMN [36], AAAI 2024	16.83	14.70	14.59	20.61	18.93	18.34	0.05	13.2
Chat-Scene [11], NeurIPS 2024	23.07	21.60	21.50	28.53	24.45	23.58	0.35	22.7
LESS [25], NeurIPS 2024	22.80	20.03	17.37	31.22	19.51	23.46	0.05	22.7
IPDN [5], AAAI 2025	20.89	17.35	16.28	25.08	21.96	21.18	0.07	15.6
Reason3D [12], 3DV 2025	17.45	15.81	15.68	21.13	19.82	19.15	0.38	24.5
UniVLG [15], ICML 2025	22.03	18.29	18.12	26.08	22.58	21.76	0.31	20.9
3D-LLaVA [7], CVPR 2025	21.12	20.45	19.94	23.52	20.96	20.23	0.25	22.4
Our MORE-STEM	**39.45**	**38.62**	**37.95**	**44.93**	**42.19**	**40.67**	0.19	28.9

'Please segment the masks of all the categories in this point cloud in order and output separate masks for each category in the following order: {category1}, {category2}, ...'. The sequence is delineated as follows: *car, bicycle, motorcycle, truck, other-vehicle, person, bicyclist, motorcyclist, road, parking, sidewalk, other-ground, building, fence, vegetation, trunk, terrain, pole, traffic-sign, moving-car, moving-bicyclist, moving-person, moving-motorcyclist, moving-truck, moving-other-vehicle*. This sequence demands the processing of 19 static and 6 dynamic outdoor classes. This methodology enables precise semantic segmentation in complex 3D/4D environments through structured commands. These commands activate the network's cross-modal alignment capabilities during segmentation processing, maintaining geometric-semantic consistency. The methodology integrates with our memory module to leverage retrieved visual-text knowledge, resolving categorical ambiguities.

III. More Implementation Details

All experiments on our proposed InstructKITTI 3D/4D benchmark are conducted on the same hardware configuration. The reported results are obtained by re-training and testing each method using the official open-source code released by their authors, averaged over at least three independent runs.

For text query encoding, we employ a pretrained LLaMA2-7B model [32], where extracted text features are converted from `float32` to `float16` precision to effectively reduce GPU memory usage. The proposed long-short memory recall module maintains a compact memory footprint, with stored content controlled within 100 MB. All reported results across different datasets represent the average of at least three test runs, with standard deviations kept within 0.5%, ensuring experimental reliability and statistical stability.

IV. More Experiments

IV.1. Experiments on our InstructKITTI 3D/4D

We conducted a comprehensive and fair comparison between our method and several recent open-source approaches from the past two years on our proposed InstructKITTI 3D/4D benchmark. All methods were evaluated using three accuracy metrics: mean Intersection over Union (mIoU), Acc@50, and Acc@25. Acc@50 and Acc@25 define a sample as correctly identified if its IoU exceeds 0.5 or 0.25, respectively, and calculate accuracy as the proportion of correctly identified samples among all samples. In addition, we compared the inference speed and GPU memory usage of all methods to provide a complete assessment of their efficiency.

Table I provides a comprehensive comparison of recent state-of-the-art methods on the InstructKITTI 3D/4D benchmark, evaluating both segmentation accuracy and computational efficiency. Overall, our method significantly outperforms all existing approaches across all metrics in both 3D and 4D settings. In 3D instruction-guided segmentation, Chat-Scene [11] previously achieved the strongest reported performance with 23.07 Acc@25, 21.60 Acc@50, and 21.50 mIoU. In contrast, our method attains 39.45 Acc@25, 38.62 Acc@50, and 37.95 mIoU, surpassing the leading baseline by substantial margins of +16.38 Acc@25, +17.02 Acc@50, and +16.45 mIoU. These improvements highlight the effectiveness of our proposed cross-modal alignment strategy and memory-guided reasoning mechanism in handling complex 3D grounding queries.

In the more challenging 4D spatio-temporal setting, our model also establishes a new state of the art. While

Table II. Temporal consistency evaluation on proposed InstructKITTI 4D.

Method	$\mathrm{IoU}_{\mathrm{Stab}} \uparrow$	$\mathrm{Var}_{\mathrm{IoU}} \downarrow$
Chat-Scene [11]	0.684	0.015
3D-LLaVA [7]	0.658	0.013
Our MORE-STEM	0.859	0.006

Chat-Scene reaches 28.53 Acc@25, 24.45 Acc@50, and 23.58 mIoU, our approach achieves 44.93 Acc@25, 42.19 Acc@50, and 40.67 mIoU, showing consistent relative gains of over 17 points across all metrics. This performance demonstrates the model's ability to maintain coherent temporal grounding and stable multi-modal reasoning under dynamic scene variations. Regarding efficiency, our method achieves 0.19 s latency with a GPU memory footprint of 28.9 GB. Although slightly heavier than lightweight architectures such as 3D-STMN [36], which runs at 0.05 s and 13.2 GB of GPU memory, our approach provides a significantly better accuracy-efficiency balance than prior large multi-modal models such as Chat-Scene [11] and 3D-LLaVA [7], both of which consume comparable memory but deliver substantially lower accuracy. The results verify that the proposed architectural components introduce meaningful accuracy gains without incurring prohibitive computational overhead.

Together, these findings confirm that our model provides a new state of the art for both 3D and 4D instruction-driven segmentation, delivering superior multi-modal alignment, stronger text-vision reasoning, and robust temporal grounding capabilities.

IV.2. Temporal Consistency Evaluation

In addition to the evaluation metrics presented in the main paper, we further propose two metrics to assess temporal consistency: (1) IoU stability, defined as $\mathrm{IoU}_{\mathrm{Stab}} = \frac{1}{T-1}\sum_{t=1}^{T-1}\mathrm{IoU}(\mathrm{Mask}_t, \mathrm{Mask}_{t+1})$, which measures the mask consistency of static objects between consecutive frames; and (2) IoU variance, defined as $\mathrm{Var}_{\mathrm{IoU}} = \mathrm{Var}\big(\mathrm{IoU}_1, \mathrm{IoU}_2, \ldots, \mathrm{IoU}_T\big)$, which quantifies the variation of IoU for the same object across the sequence.

Table II presents the evaluation results on temporal consistency. Our method achieves an $\mathrm{IoU}_{\mathrm{Stab}}$ of 0.859 and a $\mathrm{Var}_{\mathrm{IoU}}$ of 0.006, significantly outperforming Chat-Scene [11] (0.684, 0.015) and 3D-LLaVA [7] (0.658, 0.013). These results indicate that our method produces substantially more stable mask predictions across consecutive frames while maintaining low temporal fluctuations.

The improvement in $\mathrm{IoU}_{\mathrm{Stab}}$ suggests that the proposed model effectively preserves spatial coherence for static objects under dynamic scene changes, reducing inconsistency caused by viewpoint variation or partial occlusion. Meanwhile, the notably lower $\mathrm{Var}_{\mathrm{IoU}}$ demonstrates that the model yields more consistent segmentation quality over time, avoiding abrupt performance degradation across frames. Compared with existing methods, which exhibit larger temporal variance and less stable inter-frame alignment, our approach achieves a better balance between spatial accuracy and temporal smoothness. This can be attributed to the proposed spatiotemporal modeling strategy, which enforces more consistent feature representation across frames and mitigates noise accumulation in sequential predictions. Overall, the results further validate the robustness and reliability of our method for dynamic scene understanding.

Table III. Component level runtime evaluation on proposed InstructKITTI.

Module	Latency (s)	GPU (GB)
CFTVA	$\sim$0.07	$\sim$7.2
STEM	$\sim$0.04	$\sim$4.3
LTM & STM	$\sim$0.03	$\sim$10.1
Other Components	$\sim$0.06	$\sim$7.3
Full MORE-STEM	0.19	28.9

IV.3. Runtime Evaluation

From a component-wise efficiency perspective, as shown in Table III, the cross-frame text-visual alignment module (CFTVA), the spatio-temporal consistency module (STEM), the long- and short-term memory recall modules (LTM & STM), and other computational components incur approximately 0.07s, 0.04s, 0.03s, and 0.06s latency per query, respectively, resulting in an overall latency of 0.19s. In terms of GPU memory consumption, these components require 7.2 GB, 4.3 GB, 10.1 GB, and 7.3 GB, respectively, leading to a total memory footprint of 28.9 GB.

These results demonstrate that the proposed modules introduce only moderate computational overhead while delivering substantial performance gains. In particular, the memory modules (LTM & STM) account for the largest portion of GPU usage, which is expected due to their role in maintaining temporal context across frames. However, their latency contribution remains relatively small, indicating an efficient design that balances memory usage and computational cost. Furthermore, the latency distribution across modules suggests that no single component dominates the inference time, reflecting a well-balanced pipeline. The CFTVA module, while slightly more time-consuming, plays a critical role in cross-modal alignment, which is essential for improving segmentation accuracy under multi-modal queries. Overall, the proposed framework maintains a fa-

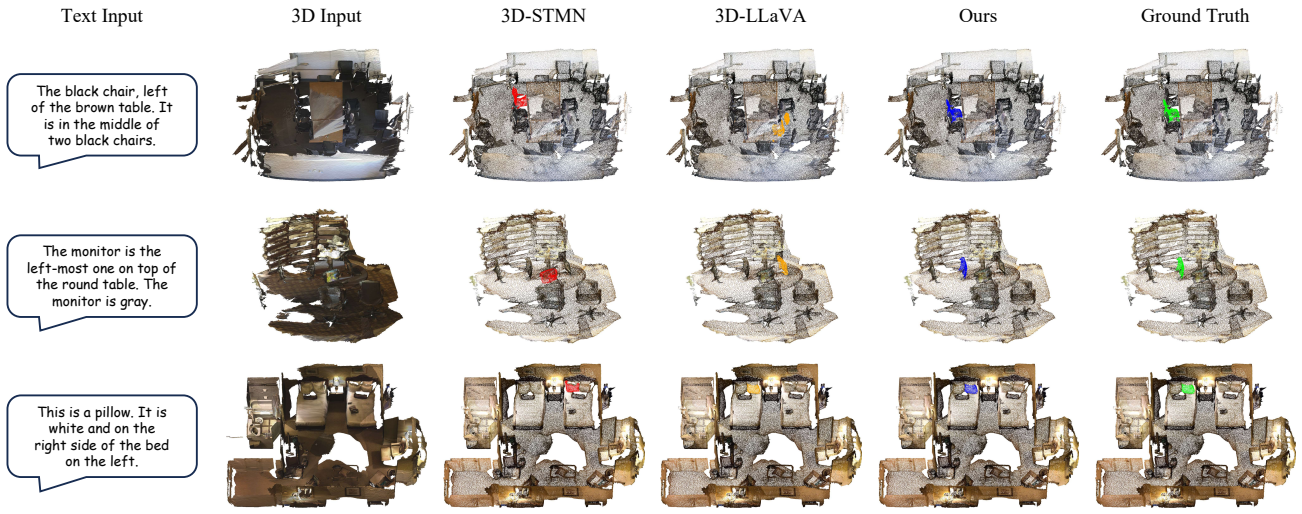

Figure II. Qualitative comparison of 3D instruction segmentation experiment on the ScanRefer [4] dataset. In the visualization results, the blue masks denote the predictions of our proposed MORE-STEM, the red masks and orange masks represent the outputs of 3D-STMN [36] and 3D-LLaVA [7], respectively, and the green masks indicate the ground-truth masks.

vorable trade-off between efficiency and performance, ensuring controllable latency and memory overhead while effectively supporting high-quality multi-modal query-based segmentation in both static and dynamic scenes.

IV.4. Qualitative Experiments on ScanRefer

Figure II presents qualitative comparisons among 3D-STMN [36], 3D-LLaVA [7], and our proposed method on the ScanRefer dataset. Across all examples, our approach demonstrates substantially improved grounding, especially for objects described by fine-grained spatial relations and subtle attribute cues.

In the first row, the instruction specifies a "black chair, left of the brown table and between two black chairs." Both 3D-STMN [36] and 3D-LLaVA [7] struggle to correctly localize the target object, either highlighting partial regions or selecting the wrong chair due to insufficient modeling of relational context. In contrast, our method precisely segments the intended chair, showing a clear understanding of both geometric layout and inter-object spatial dependencies.

The second row involves a description requiring attribute grounding ("the gray monitor") combined with positional specification ("the left-most one on the round table"). Prior methods frequently misinterpret the monitor's position or confuse visually similar objects in the cluttered scene. Our model, however, accurately identifies the correct monitor and produces a segmentation mask that closely matches the ground truth. This result highlights the effectiveness of leveraging direct cross-modal alignment and memory-guided text-vision associations to handle ambiguous or attribute-dependent referring expressions.

The third example further illustrates robustness under challenging indoor layouts. The instruction references a "white pillow on the right side of the bed," a case where similar objects, symmetric room geometry, and strong occlusions lead previous methods to produce incomplete or entirely incorrect masks. Our method not only selects the correct bed but also extracts the precise pillow region indicated by the instruction, demonstrating strong spatial discrimination and improved grounding consistency.

Overall, the qualitative results confirm that our method provides more semantically aligned and spatially coherent segmentations than existing approaches. The improvements are most notable in scenarios requiring fine-grained reasoning over relational constraints, appearance-based attributes, and cluttered 3D environments, validating the model's enhanced multi-modal understanding and its ability to maintain high-quality segmentation across diverse scenes.

IV.5. Experiments on SemanticKITTI Multi-Scan

In the multi-frame 4D semantic segmentation setting, our method achieves 61.3 mIoU on the SemanticKITTI validation set, outperforming representative 4D approaches such as MemorySeg [20] (58.5) and 4D-CS [46] (57.7) by at least 2.8 mIoU. This performance gain demonstrates the effectiveness of our method in modeling spatiotemporal dependencies and maintaining semantic consistency in dynamic 4D scenes.

The improvement can be attributed to the ability of our approach to effectively aggregate multi-frame information and leverage cross-modal memory for more robust feature representation. By incorporating temporal context across consecutive frames, the model reduces ambiguity caused by motion, occlusion, and sparsity variations, leading to more

Table IV. Complete ablation results on the InstructKITTI 3D/4D benchmark. These results further validate the effectiveness of each module and demonstrate their contributions to improving segmentation accuracy and temporal consistency.

Method	3D mIoU	3D Δ	4D mIoU	4D Δ	Latency(s)
Baseline	35.29	+0.00/-2.66	37.18	+0.00/-3.49	0.12
w/o CFTVA	36.11	+0.82/-1.84	39.53	+2.35/-1.14	0.17
w/o STEM	36.87	+1.58/-1.08	39.15	+1.97/-1.52	0.18
w/o LTM	36.91	+1.62/-1.04	39.54	+2.36/-1.13	0.16
w/o STM	37.14	+1.85/-0.81	38.95	+1.77/-1.72	0.18
Replace LLaMA2-7B with 13B	38.10	+2.81/+0.15	40.84	+3.66/+0.17	0.22
Replace LLaMA2-7B with 2B	36.73	+1.44/-1.22	39.19	+2.01/-1.48	0.18
w/ all	37.95	+2.66/-0.00	40.67	+3.49/-0.00	0.19

accurate and stable predictions. In contrast, existing 4D methods exhibit limited capability in capturing long-range temporal dependencies, which may result in inconsistent segmentation under complex dynamic conditions. Furthermore, the consistent performance improvement over strong baselines indicates that our method generalizes well across both spatial and temporal dimensions. It not only enhances per-frame segmentation accuracy but also improves temporal coherence across sequences. Overall, these results validate that our approach can effectively exploit multi-frame information and cross-modal memory mechanisms, enabling high-precision and robust understanding of complex outdoor environments in both 3D and 4D semantic segmentation tasks.

IV.6. More Ablation Experiments

Complete ablation results on our InstructKITTI 3D/4D benchmark are provided in Table IV. In this table, each Δ value follows a dual-indicator format. The number before the "/" represents the performance difference with respect to the minimal model. The number after the "/" represents the difference with respect to the complete model. This notation helps evaluate both the improvement over the minimal system and the remaining gap to the full configuration.

The removal of the CFTVA module and the use of standard cross-attention produce clear gains over the Baseline. The 3D mIoU increases by 0.82 and the 4D mIoU increases by 2.35. This variant still remains below the complete model, and the differences reach 1.84 in 3D and 1.14 in 4D. These results indicate that CFTVA improves the alignment between language queries and voxel-level features. The removal of the State-Space Inter-Frame Update in STEM also yields stable improvements over the Baseline. The gains reach 1.58 in 3D mIoU and 1.97 in 4D mIoU. This setting still performs below the complete model by 1.08 in 3D and 1.52 in 4D. The observation shows that inter-frame state propagation preserves temporal continuity and supports consistent 4D reasoning. The ablations of LTM and STM further confirm the importance of temporal modeling. Both variants outperform the Baseline in 3D and 4D accuracy. The improvements can reach 1.85 in 3D and 2.36 in 4D. Each variant still falls short of the complete model, and the remaining gap ranges from 0.81 to 1.72. These results demonstrate that both short-term and long-term temporal cues contribute to reliable spatio-temporal segmentation.

The language-model ablations provide an additional perspective. Replacing LLaMA2-7B with the larger LLaMA2-13B produces the highest gains over the Baseline. The 3D mIoU increases by 2.81 and the 4D mIoU increases by 3.66. This setting is slightly higher than the complete model in mIoU. However, its latency increases from 0.19 seconds to 0.22 seconds, which reflects a clear computational cost introduced by a larger backbone. Replacing LLaMA2-7B with the smaller 2B model results in reduced accuracy. The 3D mIoU decreases by 1.22 relative to the complete model, and the 4D mIoU decreases by 1.48. Its latency stays close to the original setting and does not provide a meaningful speed advantage. This trade-off indicates that reducing the linguistic capacity harms grounding quality without offering practical benefits in inference time.

The complete configuration achieves the highest overall performance with 37.95 in 3D mIoU and 40.67 in 4D mIoU. All other variants remain lower in at least one evaluation dimension. These results confirm that each module contributes essential reasoning ability and that their joint design produces the most reliable solution for instruction-

Table V. Ablation study for the module replacement on Instruct3D [10] and InstructKITTI-3D/4D datasets. Δ denotes the performance difference with respect to the full MORE-STEM.

Choices of Modules	Instruct3D / InstructKITTI-3D / InstructKITTI-4D mIoU	Δ
SSM $\rightarrow$ GRU	28.92 / 35.74 / 38.31	-2.51 / -2.21 / -2.36
SSM $\rightarrow$ Transformer	30.89 / 37.13 / 39.48	-0.54 / -0.82 / -1.19
Memory $\rightarrow$ feature queues	30.26 / 36.86 / 38.96	-1.17 / -1.09 / -1.71
Full MORE-STEM	31.43 / 37.95 / 40.67	+0.00 / +0.00 / +0.00

guided 3D and 4D segmentation.

Furthermore, Table V presents ablation results on the state space model (SSM) and the memory modules. Replacing the SSM with GRU or Transformer, as well as substituting the memory modules with a simple feature queue, consistently leads to performance degradation on Instruct3D and InstructKITTI for both 3D and 4D settings. In particular, the GRU-based variant exhibits the most significant drop on InstructKITTI 4D (-2.36 mIoU), highlighting the critical role of the full SSM in enabling effective cross-frame feature propagation and long-term dependency modeling. In contrast, the complete MORE-STEM achieves the best performance across all benchmarks (31.43 / 37.95 / 40.67 mIoU), demonstrating the effectiveness of the integrated design. The results indicate that the proposed components are complementary, jointly improving cross-modal alignment, spatiotemporal consistency, and memory-enhanced reasoning. Removing or simplifying any of these components disrupts this synergy and leads to suboptimal performance.

Overall, the ablation study verifies that each proposed module plays an indispensable role in the framework, and further confirms the importance of sufficient model capacity for instruction-guided segmentation. The full MORE-STEM consistently delivers the most robust and reliable performance across diverse benchmarks, providing strong support for complex query-driven 3D/4D segmentation tasks.

V. Author Contributions

The first two authors contributed equally to this work. Chade Li mainly designed the network, conducted most experiments, and performed manual refinement and improvement of the dataset annotations. Haida Feng focused on dataset construction and partial comparison experiments on proposed benchmark.