MLVTG: Mamba-Based Feature Alignment and LLM-Driven Purification for multi-modal Video Temporal Grounding

1st Zhiyi Zhu
Communication University of China
Beijing, China
zhuzhyi@cuc.edu.cn

2nd Xiaoyu Wu*
Communication University of China
Beijing, China
wuxiaoyu@cuc.edu.cn

3rd Zihao Liu
Communication University of China
Beijing, China
liuzihao@cuc.edu.cn

4th Linlin Yang
Communication University of China
Beijing, China
mu4yang@gmail.com

***Abstract*—Video Temporal Grounding (VTG), which aims to localize video clips corresponding to natural language queries, is a fundamental yet challenging task in video understanding. Existing Transformer-based methods often suffer from redundant attention and suboptimal multi-modal alignment. To address these limitations, we propose MLVTG, a novel framework comprising two key designed modules: MambaAligner and LLMRefiner. MambaAligner uses the bidirectional scanning and gated filtering strategy to model temporal dependencies and extract robust video representations for multi-modal alignment. LLMRefiner leverages the specific frozen layer of a pre-trained Large Language Model (LLM) to implicitly transfer semantic priors, enhancing multi-modal alignment without fine-tuning. This dual alignment strategy, temporal modeling via structured state-space dynamics and semantic purification via textual priors, enables more precise localization. Extensive experiments on QVHighlights, Charades-STA, and TVSum demonstrate that MLVTG achieves highly competitive performance.**

***Index Terms*—Video Temporal Grounding, Mamba, Large Language Model**

I. Introduction

In video understanding, untrimmed videos dominate digital content but contain sparse valuable clips, creating strong demand for Video Temporal Grounding (VTG)—a task that aligns natural language queries with specific temporal clips in videos. VTG encompasses two core tasks: Temporal Localization (TL) and Highlight Detection (HD). However, effectively performing these tasks remains challenging due to the heterogeneous representations between visual dynamics and linguistic structures, which lead to multimodal matching ambiguities and significantly impair performance [4].

Early methods rely on 3D CNNs [15], [22], [27], but limit receptive fields resulted in insufficient video representation capability, leading to coarse multi-modal alignment. Transformer-based models [2]–[4], [25], leveraging the attention mechanism, have advanced VTG by effectively capturing global contextual dependencies. However, Transformers often suffer from redundant attention across frames [5]–[7], weakening temporal discrimination. Recently, state space models(SSMs) based Mamba [9] have demonstrated strong sequence modeling capacity in vision tasks [21], [39] by dynamically filtering redundant signals. Yet, directly applying Mamba to VTG lacks temporal expressiveness. To address this, we adopt the bidirectional scanning and the gated filtering strategy from [8], [21] to extract robust video representations for multi-modal alignment.

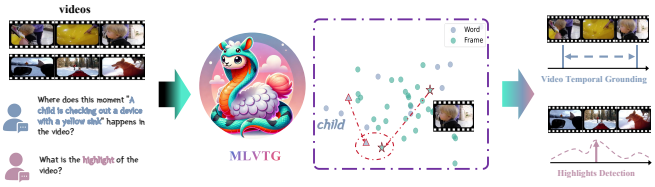

Fig. 1: **MLVTG** projects features of video-text pairs into a shared semantic space, narrows the semantic gap, and better aligns visual-text features. Its dual-branch architecture separately handles temporal localization and highlight detection.

However, enhanced visual representations alone are still insufficient for achieving the most precise semantic alignment. To elevate powerful sequence modeling capabilities into advanced semantic comprehension and reasoning, we further introduce large language mode (LLM) to enhance multi-modal alignment. It exhibits strong multi-modal reasoning abilities [10]–[12], effectively suppressing noise, purifying semantics and enhancing multi-modal alignment in video tasks [13], [14]. To leverage such capabilities, we utilize the specific frozen layer of pre-trained LLMs. With frozen pre-trained parameters, LLMs can transfer textual priors to visual domains, filtering redundant video content. According to the Platonic Representation Hypothesis [23], neural networks converge to a shared semantic space across modalities, allowing LLMs to associate abstract textual concepts with temporal visual patterns, thereby improving alignment and robustness.

Overall, we propose Mamba-based feature alignment and

* Corresponding author

This work was supported by the Fundamental Research Funds for the Central Universities CUC25SG008 and in part by the State Key Development Program in 14th Five-Year under Grant 2021YFF0900701.

LLM-driven purification for multi-modal Video Temporal Grounding termed MLVTG, a novel framework that presents MambaAligner and mamba-based LLMRefiner. MLVTG employs a dual-stage alignment strategy: the MambaAligner stage performs adaptive feature selection, while the LLMRefiner stage conducts semantic purification. MLVTG mitigates redundant attention, strengthens temporal modeling for better multi-modal alignment and introduces high-level semantic guidance without fine-tuning. To our knowledge, MLVTG is the first work to jointly exploit Mamba and LLM for VTG, achieving highly competitive performance on three benchmarks. Our key contributions include:

- We propose MLVTG, a novel dual-stage framework that handles temporal localization and highlight detection separately.
- We propose two key modules: MambaAligner, which uses bidirectional scanning and gated filtering for robust video representation and multi-modal alignment, and a mamba-based LLMRefiner that leverages frozen LLM layers to purify semantics through implicit cross-modal interactions without fine-tuning.
- Extensive experiments demonstrate that MLVTG achieves highly competitive performance across three benchmarks: QVHighlights, Charades-STA, and TVSum.

II. Related Work

A. Video Temporal Grounding

Video Temporal Grounding (VTG), a key task linking visual content with semantics, has evolved into two subtasks: Temporal Localization (TL) and Highlight Detection (HD). TL focuses on grounding natural language queries to video clips, with early proposal-based methods [15] suffering from low coverage, while proposal-free models [1], [22] improve efficiency via direct boundary regression. HD, in contrast, identifies salient video moments, progressing from ranking-based methods [16] to DETR-based detectors [3] with improved accuracy. Though traditionally studied separately, TL and HD share temporal reasoning goals. M-DETR [2] unified them using QVHighlights, inspiring subsequent work on audio integration [17], negative sample learning, and multi-task setups [31]. However, existing methods still struggle with fine-grained multi-modal alignment. To this end, we introduce MLVTG, which jointly optimizes low-level temporal structure by MambaAligner and high-level semantics by LLMRefiner, achieving improvements in both TL and HD tasks.

B. Mamba for Sequence Tasks

State Space Models (SSMs) represent long-range sequences with linear complexity [9], [18], offering an efficient alternative to Transformers by leveraging continuous state representations. However, traditional SSMs lack the ability to dynamically focus on task-relevant features across varying contexts. To address this, Mamba [19] introduces gated selective SSMs, improving both noise suppression and salient temporal pattern extraction. Originally surpassing Transformers in language modeling and time-series forecasting [19], [20], [40], Mamba has recently been extended to vision task [21], [41] and video understanding [8], [39], [42]. However, existing Mamba-based models primarily focus on unimodal tasks and overlook fine-grained video-language alignment. To overcome this, we incorporate a bidirectional SSM mechanism inspired by VideoMamba, and stack multiple Vision Mamba blocks to capture robust video representations. This enables precise localization for VTG tasks by enhancing temporal modeling and multi-modal alignment.

C. LLM for Video Temporal Grounding

Research is increasingly focused on applying LLMs to video tasks, exploring various methods. For example, VTimeLLM [26] employs LLM as the VTG decoder to directly output start and end timestamps. However, these existing methods face significant challenges, including limited capabilities in fine-grained temporal reasoning and precise timestamp localization inherent to current LLM architectures. Additionally, the requirement for extensive training exacerbates computational overhead. Motivated by [24], we propose the mamba-based LLMRefiner, a module that efficiently exploits semantic priors embedded within the specific frozen LLM layer. This strategy achieves enhanced semantic alignment between linguistic concepts and corresponding video clips, thereby significantly improving VTG performance.

III. Methodology

As illustrated in Fig2, MLVTG presents the MambaAligner with LLMRefiner, optimizing a multi-modal feature alignment mechanism. Through a dual-branch decoupled task architecture, collaborative reasoning optimization is achieved for temporal localization and highlight detection.

A. Feature Extraction and Projection.

Given an input video with L_v clips and a text query with L_q tokens, we follow the encoder design of [4] to extract features, which are then projected into a shared D-dimensional space via separate Feed-Forward Networks. This yields video features $\mathbf{V} = \{\mathbf{v}_i\}_{i=1}^{L_v} \in \mathbb{R}^{L_v \times D}$ and text features $\mathbf{Q} = \{\mathbf{q}_j\}_{j=1}^{L_q} \in \mathbb{R}^{L_q \times D}$.

We propose a dual branch task decoupled architecture. In the one branch, an attentive pooling mechanism aggregates query tokens $\mathbf{Q} \in \mathbb{R}^{L_q \times D}$ into a sentence-level representation $\mathbf{S} \in \mathbb{R}^{1 \times D}$ to compute a saliency score via similarity measurement:

$$\mathbf{S} = \mathbf{M}\mathbf{Q}, \quad \mathbf{M} = \text{Softmax}(\mathbf{W}\mathbf{Q}) \in \mathbb{R}^{1 \times L_q}, \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{1 \times L_q}$ is a learnable embedding. In another branch, we incorporate positional embeddings $\mathbf{E}^{pos}$ and modality-type embeddings $\mathbf{E}^{type}$ into both video and query tokens to preserve spatial and modality distinctions:

$$\tilde{\mathbf{V}} = \mathbf{V} + \mathbf{E}_V^{pos} + \mathbf{E}_V^{type}, \quad \tilde{\mathbf{Q}} = \mathbf{Q} + \mathbf{E}_T^{pos} + \mathbf{E}_T^{type}. \tag{2}$$

These representations are concatenated as $\mathbf{Z} = [\tilde{\mathbf{Q}}; \tilde{\mathbf{V}}] \in \mathbb{R}^{L \times D}$, where $L = L_q + L_v$. Subsequently, $\mathbf{Z}$ passes through MambaAligner for robust video representations and multi-modal alignment.

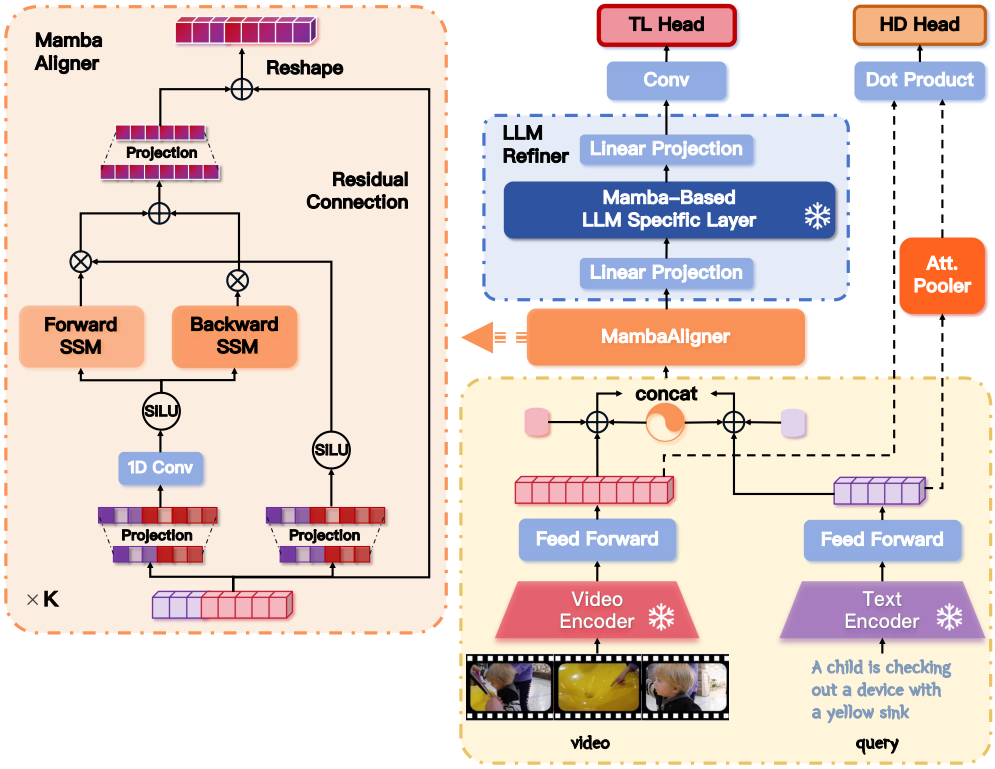

Fig. 2: **Overview of the MLVTG Framework.** The MLVTG first processes the input video and query using frozen feature encoders and projects them into a shared semantic space (Sec. III-A). It then computes saliency scores directly in one branch, while in the other, fused features are processed by the **MambaAligner** (Sec. III-B). The aligned features undergo semantic purification via the Mamba-based **LLMRefiner** (Sec. III-C). Finally, task-specific heads (Sec. III-D) classify the refined features to produce results.

B. MambaAligner.

To better capture video representations for multi-modal alignment, MambaAligner consists of K stacked vision mamba blocks and takes the concatenated feature sequence $\mathbf{Z} = [\tilde{\mathbf{Q}}; \tilde{\mathbf{V}}]$ as the input (see Fig 2). The input is first normalized and projected into two branches:

$$\hat{\mathbf{Z}} = \mathrm{Norm}(\mathbf{Z}), \quad \mathbf{x} = W_x \hat{\mathbf{Z}}, \quad \mathbf{g} = W_g \hat{\mathbf{Z}}, \tag{3}$$

where Norm denotes layer normalization, and W_x, W_g are learnable projections. Here, $\mathbf{x}$ is used for sequence modeling, while $\mathbf{g}$ serves as the gating signal.

A 1D convolution layer extracts local features from $\mathbf{x}$, generating forward and backward sequences $\mathbf{x}^f$ and $\mathbf{x}^b$, respectively. Each is passed through a SSM with parameters (A, B, C):

$$\mathbf{y} = \mathbf{x} * \mathbf{K}, \quad \mathbf{K} = \left[CA^0B, CA^1B, \cdots, CA^{L-1}B\right], \tag{4}$$

where $\mathbf{K}$ is the impulse response kernel and $*$ denotes convolution. This yields $\mathbf{y}^f$ and $\mathbf{y}^b$, capturing global context in both directions.

To fuse outputs, a gating mechanism uses the control signal $\mathbf{g}$:

$$\mathbf{y}_{\text{fused}} = \sigma(\mathbf{g}) \odot \mathbf{y}^f + \big(1 - \sigma(\mathbf{g})\big) \odot \mathbf{y}^b, \tag{5}$$

where $\sigma(\cdot)$ is the SiLU activation, and $\odot$ denotes element-wise multiplication. The final output is computed via residual connection:

$$\mathbf{Z}_{\text{out}} = \mathbf{Z} + \mathbf{y}_{\text{fused}}.$$

C. LLMRefiner.

To refine the semantics of latent representations and enhance multi-modal alignment, we propose the Mamba-based LLMRefiner module. This design addresses the performance deviations caused by structural differences. The module integrates two linear layers F_L^1 and F_L^2, and a pre-trained Mamba-based large language model layer F_{LLM} to improve semantic representation and multi-modal alignment. Leveraging the inherent strong multi-modal reasoning capabilities of large language models, the frozen-parameter F_{LLM} layer effectively transfers textual priors into the visual domain, filtering out redundant or irrelevant information in video content.

Due to the dimension mismatch between the output of the MambaAligner($\mathbf{Z}out$), and the input dimension requirements of the F_{LLM} layer, we integrate linear projection layers F_L^1 and F_L^2 before and after the F_{LLM} block, respectively. These projection layers harmonize the dimensional compatibility, modifying the network output according to the following formulation:

$$\mathbf{Z}_{refine} = F_L^1 * F_{LLM}(\mathbf{Z}_{out}) * F_L^2. \tag{6}$$

In our implementation, the parameters of the pre-trained F_{LLM} block remain entirely frozen to preserve their robust semantic representation capability. In contrast, the linear projection layers, F_L^1 and F_L^2, are fully trainable and optimized throughout the training process. This targeted training approach ensures continuous refinement of semantic alignment

TABLE I: **Performances on QVHighlights.** The optimal results are bold, and the suboptimal results are underlined.

Split	Test							Val						
Method	TL					HD		TL					HD	
	R1		mAP			≥ Very Good		R1		mAP			≥ Very Good	
	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1
M-DETR [2]	52.9	33.0	54.8	29.4	30.7	35.7	55.6	53.9	34.8	-	-	32.2	35.7	55.6
UMT [17]	56.2	41.2	53.4	37.0	36.1	38.2	60.0	60.3	44.3	-	-	38.6	39.9	64.2
UniVTG [4]	58.9	40.9	57.6	35.6	35.5	38.2	61.0	59.7	-	-	-	36.1	38.8	61.8
MH-DETR [28]	60.0	42.4	60.7	38.1	38.3	38.2	60.5	60.8	44.9	60.7	39.6	39.2	38.7	61.7
QD-DETR [3]	62.4	45.0	62.5	39.9	39.9	38.9	62.4	62.7	46.7	62.2	41.8	41.2	39.1	63.0
TR-DETR [29]	**64.6**	**48.9**	**63.9**	43.7	42.6	**39.9**	63.4	-	-	-	-	-	-	-
TaskWeave [30]	-	-	-	-	-	-	**64.2**	64.2	50.0	**65.3**	46.4	45.3	39.2	63.6
UVCOM [31]	63.5	47.4	63.3	42.6	**43.1**	39.7	**64.2**	65.1	**51.8**	-	-	**45.7**	-	-
TimeChat [43]	-	-	-	-	-	-	-	-	-	-	-	-	14.5	23.9
VTG-LLM [44]	-	-	-	-	-	-	-	-	-	-	-	-	16.5	33.5
MLVTG	64.0	48.3	63.3	**43.9**	43.0	**39.9**	63.9	**65.1**	50.5	63.7	**46.8**	44.6	**40.5**	**65.2**

TABLE II: **Temporal Localization Performances on Charades-STA.**

Method	Temporal Localization		
	R1@0.5	R1@0.7	mIoU
M-DETR [2]	53.6	31.3	-
UniVTG [4]	58.0	35.6	50.1
QD-DETR [3]	55.5	34.1	-
TR-DETR [29]	57.6	33.5	-
TaskWeave [30]	56.5	33.6	-
UVCOM [31]	**59.2**	36.6	-
TimeChat [43]	32.2	13.4	-
VTG-LLM [44]	33.8	15.7	-
VTimeLLM [26]	34.3	14.7	-
MLVTG	58.3	**38.7**	**50.3**

TABLE III: **Highlight Detection performance of Top-5 mAP on TVSum.** "†" denotes utilizing the audio modality. The suboptimal results are underlined.

Method	Highlight Detection										
	VT	VU	GA	MS	PK	PR	FM	BK	BT	DS	Avg.
Trailer [32]	61.3	54.6	65.7	60.8	59.1	70.1	58.2	64.7	65.6	68.1	62.8
SL-Module [33]	86.5	68.7	74.9	**86.2**	79.0	63.2	58.9	72.6	78.9	64.0	73.3
PLD-VHD [34]	84.5	80.9	70.3	72.5	76.4	87.2	71.9	74.0	74.4	**79.1**	77.1
UniVTG [4]	83.9	85.1	**89.0**	80.1	**84.6**	81.4	70.9	**91.7**	73.5	69.3	**81.0**
MINI-Net† [35]	80.6	68.3	78.2	81.8	78.1	65.8	57.8	75.0	80.2	65.5	73.2
TCG† [36]	85.0	71.4	81.9	78.6	80.2	75.5	71.6	77.3	78.6	68.1	76.8
Joint-VA† [37]	83.7	57.3	78.5	86.1	80.1	69.2	70.0	73.0	**97.4**	67.5	76.3
CO-AV† [38]	**90.8**	72.8	84.6	85.0	78.3	78.0	**72.8**	77.1	89.5	72.3	80.1
MLVTG	83.6	**85.6**	74.7	75.9	82.4	**87.7**	64.6	91.6	80.9	73.7	80.1

specifically tailored for downstream multi-modal alignment tasks.

D. Joint Optimization for TL & HD.

To jointly solve TL and HD, we design a dual-head module to produce task-specific outputs coupled with corresponding optimization objectives. Specifically, the TL Head takes refined feature $\mathbf{Z}_{refine}$ as input and integrates two parallel parts composed of 1D conv layers. One predicts the start and end timestamps ($[st, ed]$), while the other performs frame-level binary classification to discriminate foreground from background frames. In contrast, the HD Head receives video features $\mathbf{V}$ and sentence-level representations $\mathbf{S}$ as input and focuses on frame-level salient scores. The overall training objective aggregates task-specific losses with balancing coefficients, formulated as follows:

$$\mathcal{L}_{overall} = \lambda_f \mathcal{L}_f + \lambda_{reg} \mathcal{L}_{reg} + \lambda_1 \mathcal{L}^{inter} + \lambda_2 \mathcal{L}^{intra}, \quad (7)$$

where $\mathcal{L}_f$ represents the classification loss, and $\mathcal{L}_{reg}$ denotes the regression loss, both dedicated to optimizing TL. Conversely, $\mathcal{L}^{inter}$ and $\mathcal{L}^{intra}$ are explicitly employed to enhance the accuracy and discriminative power of HD [4].

IV. Experiments

A. Implementation Details.

We use CLIP and SlowFast features from pre-segmented video clips at fixed frame rates (0.5/1 FPS), while text features are extracted through the CLIP text encoder. The model employs a MambaAligner (4 blocks, hidden size 1024) and an LLMRefiner which operates on 2048 dimension features extracted from the 20th layer. Training used a batch size of 32 for 200 epochs on a RTX4090 GPU.

B. Comparison with Start-of-the-Arts

Table I demonstrates outstanding performance on QVHighlights. On the test set, it achieves the highest mAP@0.75 and HD mAP. Since TR-DETR specifically designs a HD module to assist temporal localization, it performs well. However, more generalized MLVTG outperforms it on the generic TL dataset Charades-STA, leading QD-DETR by 8.6% in R1@0.7 and outperforming both UVCOM and QD-DETR in HD task HIT@1. On the Charades-STA, MLVTG achieves optimal results under R1@0.7 and mIoU in Table II. Moreover, without relying on complex detection architectures, it delivers superior overall performance compared to detector-based methods such as TR-DETR. On TVSum , MLVTG attains an average AP of 80.1,matching SOTA CO-AV in Table III. Even without multi-dataset training like UniVTG, MLVTG remains highly competitive. And without introducing audio modality, it exhibits stronger generalization capability than complex models such as Joint-VA.

TABLE V: **Effectiveness of LLM Pretrained Parameters.** "–" means no LLM layer. "Rand" means randomly initialized.

LLM Param	Frozen	R1@0.7	mAP
–	–	43.5	37.7
Rand	No	43.6	36.7
Rand	Yes	43.3	37.1
Yes	No	42.8	36.4
Yes	Yes	**45.1**	**38.4**

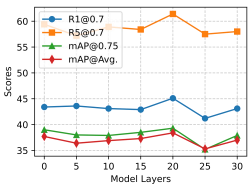

Fig. 3: **Effectiveness of LLM Different Layers.**

TABLE IV: **Ablation Study on Three Benchmarks.** We select the most representative indicators as metrics for each task.

Mamba Aligner	LLM Refiner	Temporal Localization									Highlight Detection		
		QVHighlights					Charades-STA				QVHighlights		TVSum
		R1	R5	mIoU	mAP		R1	mIoU	mAP		≥ Very Good		Avg.
		@0.7	@0.7		@0.75	Avg.	@0.7		@0.75	Avg	HIT@1	mAP	
✗	✗	43.5	59.4	54.7	59.5	37.7	35.6	49.4	38.5	39.0	64.2	39.3	72.7
✓	✗	48.9	66.3	59.1	62.9	42.4	38.7	50.2	41.0	40.4	64.8	40.1	77.1
✗	✓	45.1	61.4	55.8	60.8	38.4	35.4	49.4	38.3	39.4	62.8	38.2	77.7
✓	✓	**50.5**	**66.9**	**59.9**	**63.7**	**44.6**	**38.7**	**50.3**	**41.0**	**40.8**	**65.2**	**40.5**	**80.1**

V. Ablation Study

As shown in the Table IV, MambaAligner alone significantly improves temporal localization performance: on QVHighlights, R1@0.7 increased by 12.4%, mIoU by 8.0%, and mAP@Avg by 8.7%; on Charades-STA, it improved by 8.7%, 1.6%, and 3.6%, respectively. LLMRefiner provides improvements (e.g., +3.7% in R1@0.7 on QVHighlights), and the combination of both yields the best results, with total gains of 16.1% in R1@0.7, 9.5% in mIoU, and 14.4% in mAP@Avg. On the HD task, the full model also achieves the highest performance (+3.1% in mAP and +10.2% in Avg). Furthermore, Table V demonstrate that using frozen pretrained LLM parameters delivers the best outcomes (e.g., 45.1 R1@0.7 and 38.4 mAP@Avg), while fine-tuning or random initialization leads to degradation.

[23] suggests that large language models can learn shared, task-agnostic representations, and prior empirical evidence shows that such universal semantic spaces are most prominent in intermediate layers rather than at the output. Subsequent studies [24], [45] further indicate that layers closer to the output increasingly encode task-specific, objective-driven features with limited cross-task and cross-domain generalization, as they tend to over-specialize toward pretraining or fine-tuning objectives and thus weaken transferability. Consistent with these findings, our empirical analysis confirms that middle-to-late layers provide a better balance between semantic generality and task relevance; therefore, we select Layer-20 as the representative layer for semantic optimization, a choice further validated by the results in Fig 3. It is also worth noting that fine-tuning large-scale models on small datasets often leads to semantic collapse, damaging the model's original semantic structure, and Table V (rows 2,3,5) reveals overfitting when fine-tuning is conducted at suboptimal layers; hence, we freeze the parameters of the selected layer.

TABLE VI: **Effectiveness of Bidirectional Scanning.**

Model	QVHighlights			Charades-STA		
	R1@0.7	mIoU	mAP	R1@0.7	mIoU	mAP
Unidirection	31.2	49.8	29.5	32.9	46.4	33.5
Bidirection	48.9	59.1	42.4	38.7	50.2	40.4

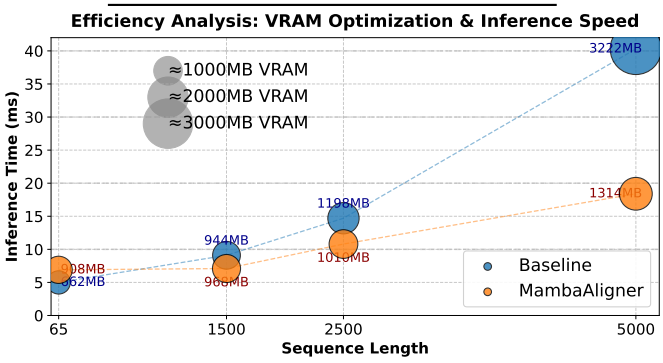

Fig. 4: **Efficiency Analysis of MambaAligner.**

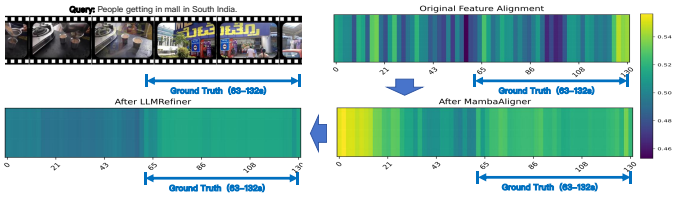

Fig. 5: **Feature Alignment Visualization.** Darker colors indicate lower similarity.

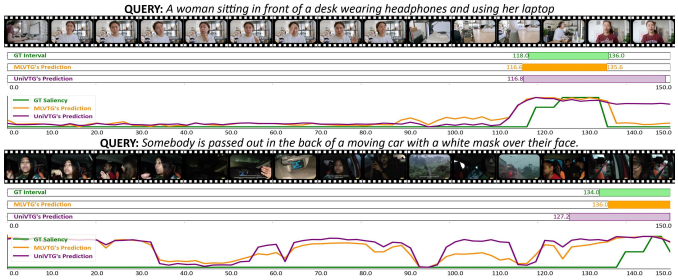

Fig. 6: **Experiment Results Visualization**

Table VI shows that bidirectional scanning consistently outperforms unidirectional scanning on both QVHighlights and Charades-STA. Fig. 4 further indicates that MambaAligner achieves markedly lower VRAM consumption and faster inference than the baseline as sequence length increases; while existing datasets are relatively short and thus understate its linear-time benefits, the architecture is well suited for long-sequence processing.

VI. Visualization

As shown in Fig 5, the cosine similarity heatmap between query and video features reveals poor initial alignment. After introducing MambaAligner, a substantial improvement in alignment quality is observed, though spurious high-similarity regions remain—particularly in the early clips due to background distractors. Further refinement with LLMRefiner effectively suppresses noise and concentrates high similarity within the ground-truth interval (63–132 seconds), resulting in more precise alignment. Furthermore, Fig 6 further compares our method MLVTG with the baseline on QVHighlights.

VII. Conclusion

This paper proposes MLVTG, a novel framework for video temporal grounding that presents MambaAligner and LLM-

Refiner for effective multi-modal alignment. Experiments on three benchmarks show that MLVTG outperforms highly competitive performance in both temporal localization and highlight detection. Ablation studies further confirm the effectiveness of each component, demonstrating the robustness and superiority of MLVTG. Future work will explore extending MLVTG to incorporate audio modalities for more comprehensive video temporal grounding.

REFERENCES

[1] Zhang, H., Sun, A., Jing, W. & Zhou, J. Span-based Localizing Network for Natural Language Video Localization. *ACL*. pp. 6543-6554 (2020)

[2] Lei, J., Berg, T. & Bansal, M. Detecting Moments and Highlights in Videos via Natural Language Queries. *NeurIPS*. pp. 11846-11858 (2021)

[3] Moon, W., Hyun, S., Park, S., Park, D. & Heo, J. Query - Dependent Video Representation for Moment Retrieval and Highlight Detection. *CVPR*. pp. 23023-23033 (2023)

[4] Lin, K., Zhang, P., Chen, J., Pramanick, S., Gao, D., Wang, A., Yan, R. & Shou, M. UniVTG: Towards Unified Video-Language Temporal Grounding. *ICCV*. pp. 2782-2792 (2023)

[5] Zeng, W., Jin, S., Liu, W., Qian, C., Luo, P., Ouyang, W. & Wang, X. Not all tokens are equal: Human-centric visual analysis via token clustering transformer. *CVPR*. pp. 11101-11111 (2022)

[6] Beltagy, I., Peters, M. & Cohan, A. Longformer: The Long-Document Transformer. *CoRR*. **abs/2004.05150** (2020)

[7] Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lucic, M. & Schmid, C. ViViT: A Video Vision Transformer. *ICCV*. pp. 6816-6826 (2021)

[8] Li, K., Li, X., Wang, Y., He, Y., Wang, Y., Wang, L. & Qiao, Y. VideoMamba: State Space Model for Efficient Video Understanding. *ECCV*. **15084** pp. 237-255 (2024)

[9] Gu, A., Goel, K. & Ré, C. Efficiently Modeling Long Sequences with Structured State Spaces. *ICLR*. (2022)

[10] Wang, Y., Chen, W., Han, X., Lin, X., Zhao, H., Liu, Y., Zhai, B., Yuan, J., You, Q. & Yang, H. Exploring the Reasoning Abilities of Multimodal Large Language Models (MLLMs): A Comprehensive Survey on Emerging Trends in Multimodal Reasoning. *CoRR*. **abs/2401.06805** (2024)

[11] Ye, Q., Xu, H., Xu, G., Ye, J., Yan, M., Zhou, Y., Wang, J., Hu, A., Shi, P., Shi, Y., Li, C., Xu, Y., Chen, H., Tian, J., Qi, Q., Zhang, J. & Huang, F. mPLUG-Owl: Modularization Empowers Large Language Models with Multimodality. *CoRR*. **abs/2304.14178** (2023)

[12] Driess, D., Xia, F., Sajjadi, M., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., Huang, W., Chebotar, Y., Sermanet, P., Duckworth, D., Levine, S., Vanhoucke, V., Hausman, K., Toussaint, M., Greff, K., Zeng, A., Mordatch, I. & Florence, P. PaLM-E: An Embodied Multimodal Language Model. *International Conference On Machine Learning, ICML*. **202** pp. 8469-8488 (2023)

[13] Zheng, R., Qi, L., Chen, X., Wang, Y., Wang, K., Qiao, Y. & Zhao, H. ViLLa: Video Reasoning Segmentation with Large Language Model. *CoRR*. **abs/2407.14500** (2024)

[14] Chen, G., Zheng, Y., Wang, J., Xu, J., Huang, Y., Pan, J., Wang, Y., Wang, Y., Qiao, Y., Lu, T. & Wang, L. VideoLLM: Modeling Video Sequence with Large Language Models. *CoRR*. **abs/2305.13292** (2023)

[15] Hendricks, L., Wang, O., Shechtman, E., Sivic, J., Darrell, T. & Russell, B. Localizing Moments in Video with Natural Language. *ICCV*. pp. 5804-5813 (2017)

[16] Gygli, M., Song, Y. & Cao, L. Video2GIF: Automatic Generation of Animated GIFs from Video. *CVPR*. pp. 1001-1009 (2016)

[17] Liu, Y., Li, S., Wu, Y., Chen, C., Shan, Y. & Qie, X. UMT: Unified Multi-modal Transformers for Joint Video Moment Retrieval and Highlight Detection. *CVPR*. pp. 3032-3041 (2022)

[18] Gupta, A., Gu, A. & Berant, J. Diagonal State Spaces are as Effective as Structured State Spaces. *NeurIPS*. (2022)

[19] Gu, A. & Dao, T. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *CoRR*. **abs/2312.00752** (2023)

[20] Fu, D., Dao, T., Saab, K., Thomas, A., Rudra, A. & Ré, C. Hungry Hungry Hippos: Towards Language Modeling with State Space Models. *ICLR*. (2023)

[21] Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W. & Wang, X. Vision Mamba: Efficient Visual Representation Learning with Bidirectional State Space Model. *ICML*. (2024)

[22] Lei, J., Yu, L., Berg, T. & Bansal, M. TVR: A Large-Scale Dataset for Video-Subtitle Moment Retrieval. *ECCV*. **12366** pp. 447-463 (2020)

[23] Huh, M., Cheung, B., Wang, T. & Isola, P. Position: The Platonic Representation Hypothesis. *ICML*. (2024)

[24] Pang, Z., Xie, Z., Man, Y. & Wang, Y. Frozen Transformers in Language Models Are Effective Visual Encoder Layers. *ICLR*. (2024)

[25] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. & Zagoruyko, S. End-to-End Object Detection with Transformers. *ECCV*. **12346** pp. 213-229 (2020)

[26] Huang, B., Wang, X., Chen, H., Song, Z. & Zhu, W. VTimeLLM: Empower LLM to Grasp Video Moments. *CVPR*. pp. 14271-14280 (2024)

[27] Escorcia, V., Soldan, M., Sivic, J., Ghanem, B. & Russell, B. Temporal Localization of Moments in Video Collections with Natural Language. *CoRR*. **abs/1907.12763** (2019)

[28] Xu, Y., Sun, Y., Zhai, B., Jia, Y. & Du, S. MH-DETR: Video Moment and Highlight Detection with Cross-modal Transformer. *IJCNN*. pp. 1-8 (2024)

[29] Sun, H., Zhou, M., Chen, W. & Xie, W. TR-DETR: Task-Reciprocal Transformer for Joint Moment Retrieval and Highlight Detection. *AAAI*. pp. 4998-5007 (2024)

[30] Yang, J., Wei, P., Li, H. & Ren, Z. Task-Driven Exploration: Decoupling and Inter-Task Feedback for Joint Moment Retrieval and Highlight Detection. *CVPR*. pp. 18308-18318 (2024)

[31] Xiao, Y., Luo, Z., Liu, Y., Ma, Y., Bian, H., Ji, Y., Yang, Y. & Li, X. Bridging the Gap: A Unified Video Comprehension Framework for Moment Retrieval and Highlight Detection. *CVPR*. pp. 18709-18719 (2024)

[32] Wang, L., Liu, D., Puri, R. & Metaxas, D. Learning Trailer Moments in Full-Length Movies with Co-Contrastive Attention. *ECCV*. **12363** pp. 300-316 (2020)

[33] Xu, M., Wang, H., Ni, B., Zhu, R., Sun, Z. & Wang, C. Cross-category Video Highlight Detection via Set-based Learning. *ICCV*. pp. 7950-7959 (2021)

[34] Wei, F., Wang, B., Ge, T., Jiang, Y., Li, W. & Duan, L. Learning Pixel-Level Distinctions for Video Highlight Detection. *CVPR*. pp. 3063-3072 (2022)

[35] Hong, F., Huang, X., Li, W. & Zheng, W. MINI-Net: Multiple Instance Ranking Network for Video Highlight Detection. *ECCV*. **12358** pp. 345-360 (2020)

[36] Ye, Q., Shen, X., Gao, Y., Wang, Z., Bi, Q., Li, P. & Yang, G. Temporal Cue Guided Video Highlight Detection with Low-Rank Audio-Visual Fusion. *ICCV*. pp. 7930-7939 (2021)

[37] Badamdorj, T., Rochan, M., Wang, Y. & Cheng, L. Joint Visual and Audio Learning for Video Highlight Detection. *ICCV*. pp. 8107-8117 (2021)

[38] Li, S., Zhang, F., Yang, K., Liu, L., Liu, S., Hou, J. & Yi, S. Probing Visual-Audio Representation for Video Highlight Detection via Hard-Pairs Guided Contrastive Learning. *BMVC*. pp. 709 (2022)

[39] Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Jiao, J. & Liu, Y. VMamba: Visual State Space Model. *NeurIPS*. (2024)

[40] Liang, A., Jiang, X., Sun, Y. & Lu, C. Bi-Mamba4TS: Bidirectional Mamba for Time Series Forecasting. *CoRR*. **abs/2404.15772** (2024)

[41] Shen, Y., Xiao, L., Chen, J., Du, Q. & Ye, Q. Learning Cross-Task Features With Mamba for Remote Sensing Image Multitask Prediction. *IEEE Trans. Geosci. Remote. Sens.*. **63** pp. 1-16 (2025)

[42] Pan, Z., Zeng, H., Cao, J., Chen, Y., Zhang, K. & Xu, Y. MambaSCI: Efficient Mamba-UNet for Quad-Bayer Patterned Video Snapshot Compressive Imaging. *NeurIPS*. (2024)

[43] Al., S. TimeChat: A Time-sensitive Multimodal Large Language Model for Long Video Understanding. *CVPR*. pp. 14313-14323 (2024)

[44] Al., Y. VTG-LLM: Integrating Timestamp Knowledge into Video LLMs for Enhanced Video Temporal Grounding. *AAAI*. (2025)

[45] Skean, O., Arefin, M., Zhao, D., Patel, N., Naghiyev, J., LeCun, Y. & Shwartz-Ziv, R. Layer by Layer: Uncovering Hidden Representations in Language Models. (2025), https://arxiv.org/abs/2502.02013