Video Retrieval of Autonomous Driving Scenarios Based on Large Models

12310520 Rui Yuhan 12310437 Qiao Shihan

***Abstract*—Video retrieval in autonomous driving scenarios demands models to comprehend complex visual environments and align them with detailed textual descriptions. While multimodal large language models (MLLMs) have demonstrated potential in vision-language tasks, their application to domain-specific video retrieval remains underexplored. Furthermore, the scarcity of publicly available high-quality annotated video data in the autonomous driving domain makes it challenging for models to accurately learn video content semantics from limited data. Additionally, significant differences exist between autonomous driving videos and those in general video datasets, leading to suboptimal performance of generic large models on autonomous driving datasets.**

To address these challenges, we evaluate the performance of two distinct pretrained MLLMs on a carefully curated autonomous driving dataset. This dataset features high-quality captions that have undergone multi-round optimization, combining human annotation expertise with advanced automated labeling techniques, while also serving as a benchmark platform for exploring more efficient annotation strategies. Experimental results demonstrate that our approach effectively retrieves relevant video segments based on textual queries, highlighting the potential of MLLMs to enhance retrieval accuracy and scalability.

***Index Terms*—Autonomous Driving Video Retrieval, Multimodal Large Language Models (MLLMs), Video Understanding, Automated Annotation Optimization**

I. Introduction

II. Two knowledge distillation methods used

A. Research Background and Motivation of Knowledge Distillation

With the rapid advancement of large-scale foundation models, Vision-Language Models (VLMs), and Large Language Models (LLMs), deploying these models on edge devices faces significant challenges due to high computational and memory requirements. Knowledge Distillation (KD), first proposed by Hinton et al. [**?**], addresses this issue by training a lightweight student network under the supervision of a complex teacher model. Unlike compression methods that modify network structures, KD transfers knowledge by having the student mimic the teacher's output distribution as soft labels, enabling effective model compression while maintaining performance. Additionally, KD facilitates knowledge transfer from source tasks to target tasks with limited labeled data, making it particularly valuable for domain-specific applications such as autonomous driving video retrieval.

1) Mainstream Knowledge Distillation Approaches: Knowledge distillation methods can be categorized based on the source of knowledge being distilled [**?**]. The three primary categories are: **logit-based distillation**, which transfers final predictions using softmax with temperature to create soft labels; **feature-based distillation**, which transfers intermediate layer representations and attention maps, providing richer information than logits alone; and **similarity-based distillation**, which transfers structural knowledge through pairwise similarities between features, channels, or instances. Additionally, distillation can be organized by training schemes: **offline distillation** (pre-trained teacher with frozen weights), **online distillation** (simultaneous training), and **self-distillation** (same network transferring knowledge across layers or stages). Other notable approaches include **attention-based distillation**for transferring focus patterns, **contrastive distillation** leveraging contrastive learning principles, and **cross-modal distillation** for transferring knowledge between different modalities, which is particularly relevant for multimodal tasks like video retrieval.

B. DLDKD

DLDKD (Dual Learning with Dynamic Knowledge Distillation) [**?**] represents a sophisticated knowledge distillation framework specifically designed for video understanding tasks, particularly video retrieval in autonomous driving scenarios. The architecture employs a dual-stream teacher-student framework that effectively transfers rich multimodal knowledge from a large-scale teacher model to a lightweight student model while maintaining competitive retrieval performance.

1) Framework: The DLDKD framework architecture is illustrated in Figure 1.

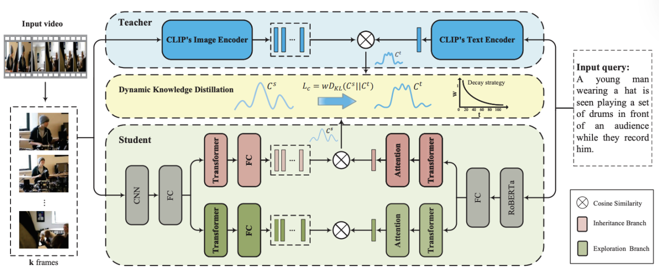


Fig. 1. The framework of DLDKD.

2) Teacher-Student Architecture: The DLDKD frameworkconsists of two key components: a large teacher model and a compact student model. The teacher model is typically a pretrained large-scale vision-language model (VLM) with extensive parameters, capable of capturing complex spatiotemporal relationships in video sequences. This teacher model serves as the knowledge source, providing rich feature representations and semantic understanding of video content. The teacher architecture typically incorporates multiple transformer layers with cross-modal attention mechanisms, enabling it to effectively align visual features with textual descriptions.

In contrast, the student model is designed with a significantly reduced parameter count and computational complexity, making it suitable for deployment on resource-constrained edge devices commonly used in autonomous driving systems. The student architecture maintains a similar structural design to the teacher but with fewer layers, reduced hidden dimensions, and optimized attention mechanisms. Despite its compact nature, the student model is trained to mimic the teacher's behavior through carefully designed distillation objectives.

The distillation process in DLDKD operates at multiple levels: feature-level distillation transfers intermediate representations from the teacher's encoder layers to corresponding 2 layers in the student, logit-level distillation aligns the final output distributions, and attention-level distillation preserves the teacher's focus patterns. This multi-level knowledge transfer ensures that the student model captures both low-level visual features and high-level semantic understanding from the teacher, enabling effective knowledge compression without significant performance degradation.

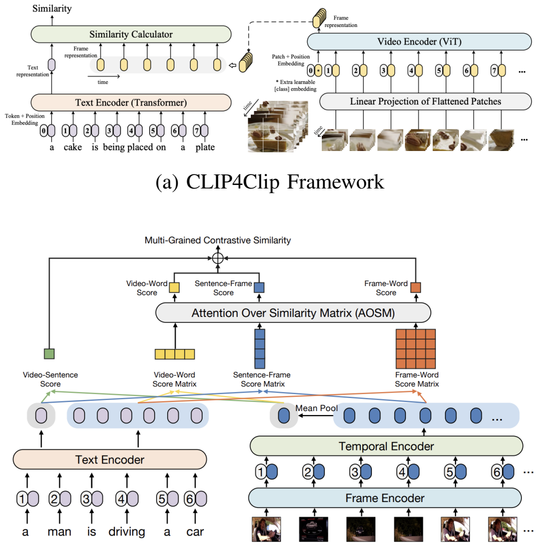
3) Application to Video Retrieval: In the context of video retrieval for autonomous driving scenarios, DLDKD addresses the critical challenge of efficiently matching textual queries with relevant video segments. The teacher model, pretrained on large-scale video-text datasets, possesses strong capabilities in understanding complex driving scenarios, including object detection, motion analysis, and scene understanding. During the distillation process, the student model learns to replicate these capabilities in a more efficient manner.

For video retrieval tasks, the DLDKD framework processes input videos through the student model's encoder to extract compact yet discriminative feature representations. These features are then compared with query embeddings in a shared semantic space, enabling efficient similarity computation. The distillation process ensures that the student model maintains the teacher's ability to understand fine-grained visual details crucial for autonomous driving scenarios, such as vehicle types, traffic conditions, weather patterns, and road infrastructure.

The application of DLDKD to video retrieval offers several advantages: first, the lightweight student model enables real-time retrieval on edge devices, which is essential for autonomous driving applications requiring low-latency responses. Second, the knowledge transferred from the teacher model helps the student generalize better to diverse driving scenarios, even when trained on limited domain-specific data. Third, the multi-level distillation approach preserves the teacher's ability to handle complex queries involving multiple objects, temporal relationships, and spatial configurations, which are common in autonomous driving video retrieval tasks.

C. TeachCLIP

1) Inspiration: Existing VTR models generally fall into two categories: lightweight global feature models (e.g., CLIP4Clip) and heavy fine-grained models (e.g., X-CLIP). The former is computationally efficient but often lacks frame-level details, resulting in suboptimal retrieval accuracy. The latter leverages frame-level interactions to enhance performance but incurs high computational costs.



(a) CLIP4Clip Framework

(b) X-CLIP Framework

Fig. 2. Comparison of CLIP4Clip and X-CLIP frameworks. Both use similar bottom-up encoding but differ in upper-layer feature usage.

In terms of vision and text encoding, both CLIP4Clip and X-CLIP share nearly identical bottom-level encoding mechanisms (Frame → Patch → ViT, Word → Embedding → Transformer). The primary distinction lies in how they utilize these features in the upper layers: CLIP4Clip performs pooling to retain only global video vectors, whereas X-CLIP preserves fine-grained features and performs multi-granularity alignment.

TeachCLIP aims to bridge this gap by distilling the fine-grained alignment capability of heavy models into a lightweight student model.

2) Framework: The TeachCLIP framework is designed to transfer the fine-grained knowledge from a teacher model to a student model while maintaining high inference efficiency.

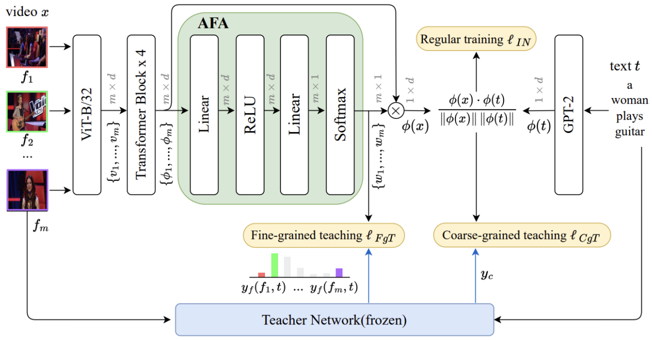


Fig. 3. The framework of TeachCLIP.

Student Model: The student model is built upon CLIP4Clip. It samples the video into frames, processes each frame through the CLIP ViT to obtain frame features, and enhances them via a temporal Transformer. A key innovation is replacing the original mean pooling with Attentional Frame-Feature Aggregation (AFA). AFA generates frame weights $\{w_i\}$ to compute a weighted sum of frame features:

$$\phi(x) = \sum_{i=1}^{m} w_i \cdot \phi_i \tag{1}$$

The AFA module consists of a lightweight structure (Linear → ReLU → Linear → Softmax) and introduces negligible parameters.

Teacher Model: The teacher model provides two types of supervision:

- **Video-level soft labels (y_c):** The teacher provides the distribution of video-text correlations to guide the student's ranking.
- **Frame-level soft labels (y_f):** The teacher provides the distribution of frame-text correlations to guide the student's frame weight allocation in AFA.

Learning Objectives: The training involves three losses:

1) *Frame-level Distillation (ℓ_{FgT}):* Optimizes the AFA weights to match the teacher's frame-text similarity distribution. The loss function is defined as:

$$\ell_{FgT} = -\frac{1}{b}\sum_{i=1}^{b}\sum_{k=1}^{m} y_f(f_{i,k}, t_i)\,\log w_{i,k} \tag{2}$$

where b is the batch size, m is the number of frames, and y_f represents the frame-text similarity from the teacher.

2) *Video-level Distillation (ℓ_{CgT}):* Aligns the student's video-text similarity matrix with the teacher's multi-grained similarity matrix using the Pearson distance d_p:

$$\begin{aligned}\ell_{CgT} = &\frac{1}{b}\sum_{i} d_p(\sigma(B_{i,\cdot}), \sigma(y_c(v_i,\cdot))) \\ +&\frac{1}{b}\sum_{j} d_p(\sigma(B_{\cdot,j}), \sigma(y_c(\cdot,t_j)))\end{aligned} \tag{3}$$

where σ denotes the softmax function and B is the similarity matrix of the student.

3) *Contrastive Learning (ℓ_{IN}):* Standard maximization of similarity for positive pairs and minimization for negative pairs, using symmetric InfoNCE loss:

$$\ell_{IN} = \frac{1}{2}\left[\ell_{NCE}^{v\to t} + \ell_{NCE}^{t\to v}\right] \tag{4}$$

where

$$\ell_{NCE}^{v\to t} = -\frac{1}{b}\sum_{i=1}^{b}\log\frac{\exp(B_{ii}/\tau)}{\sum_{j=1}^{b}\exp(B_{ij}/\tau)} \tag{5}$$

and $\ell_{NCE}^{t\to v}$ is defined symmetrically.

The total loss is $\ell = \ell_{CgT} + \ell_{FgT} + \ell_{IN}$. During inference, the teacher model is discarded, and only the lightweight student model with AFA is used.

3) TeachCLIP vs X-CLIP Parameters and FLOPs Comparison: Table I presents the comparison of parameters and FLOPs between TeachCLIP and the teacher model (X-CLIP).

TABLE I
COMPARISON OF PARAMETERS AND FLOPS

Model	Parameters	FLOPs (Inference)	Description
TeachCLIP	≈ 200M	53.65G (12 frames)	Student model, based on CLIP4Clip + AFA, maintains lightweight inference
X-CLIP (Teacher)	≈ 220M	145G (8 frames) / 287G (16 frames)	Teacher model, introduces multi-grained contrastive similarity, higher inference cost

III. ALTERATIONS ON TEACHCLIP

A. On student model

B. On teacher model

IV. CONCLUSION

Autonomous driving video retrieval presents unique challenges due to the complexity of real-world driving scenarios and the scarcity of high-quality annotated datasets. In this work, we explored the potential of multimodal large language models (MLLMs) to bridge the gap between visual perception and textual understanding in autonomous driving contexts. By leveraging a carefully curated dataset with optimized annotations, combining human expertise and automated refinement using Video-LLama2, we demonstrated that pretrained MLLMs can be effectively adapted for domain-specific video retrieval tasks.

Our experiments with two state-of-the-art MLLMs (Vast and Clip-Vip) revealed that fine-tuning on high-quality annotated data significantly improves retrieval accuracy, validating the importance of robust dataset construction. Additionally, our work establishes a benchmark for future research in autonomous driving video understanding, providing a foundation for developing more interpretable and scalable retrieval systems.

Moving forward, we anticipate that further advancements in MLLMs, combined with more sophisticated annotation pipelines, will enhance the generalization capabilities of autonomous driving systems. Future research could explore dynamic annotation strategies, real-time retrieval optimization, and the integration of multi-sensor data (e.g., LiDAR and radar) to further improve model performance in complex driving environments.

Ultimately, this study highlights the critical role of high-quality data and domain-specific adaptation in advancing autonomous driving technologies, paving the way for safer and more intelligent transportation systems.