GRANULARITY-ALIGNED TEXT-TO-VIDEO RETRIEVAL WITH SEMANTIC CONDUITS

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-video retrieval (T2VR) aims to align textual descriptions with video content, yet the core challenge lies in the inherent asymmetry of information density between the two modalities. Existing approaches often overlook this asymmetry in feature construction, either compressing videos into global representations or relying on coarse frame-level features, which conflate distinct visual concepts and yield a granularity mismatch with token-level textual features. To address this, we propose a *granularity-aligned retrieval paradigm* (GAR) that organizes video contents into semantic units better matched to fine-grained textual features. Drawing on the patch-level visual granularity that underpins fine-grained image-text alignment, we instantiate this paradigm with a hierarchical, joint spatial-temporal video representation built upon *Semantic Conduits*. Appearance-similar patches are first coalesced within each frame into semantic prototypes and then associated across frames. Soft patch-to-conduit assignments aggregate temporally coherent visual evidence into pooled conduit tokens, with a Transformer contextualizes by modeling relations among semantic components. MeanMaxSim aligns each text token with its most similar conduit and averages the token-wise scores, yielding fine-grained cross-modal correspondence while permitting video representations to be precomputed. Experiments on four benchmarks demonstrate strong retrieval performance. With a ViT-B/32 backbone, our method achieves R@1 scores of 48.4% on MSR-VTT, 48.1% on MSVD, 48.8% on DiDeMo, and 46.0% on ActivityNet. Code will be released.

1 INTRODUCTION

Text-to-video retrieval (T2VR) is the task of retrieving videos from natural language queries. Recent advances in retrieval architectures (Gabeur et al., 2020; Lei et al., 2021) and image-text pretraining, particularly CLIP (Radford et al., 2021), have substantially improved retrieval performance (Luo et al., 2022; Xue et al., 2023). At its core, T2VR requires compressing both text and video content into features in a shared semantic space so that their correspondence can be established.

A fundamental challenge lies in the inherent information-density asymmetry between modalities. Text is a compact one-dimensional token sequence, whereas video is a dense spatiotemporal signal spanning two spatial dimensions and one temporal dimension, where multiple objects, attributes, and backgrounds evolve across frames. Consequently, video contains substantially denser and more diverse information than its textual description. Therefore, effective T2VR requires not only compressing both modalities into a shared feature space, but also constructing visual features to a semantic granularity comparable to that of text, ensuring valid cross-modal alignment.

Existing approaches can be broadly characterized by how they compress video along the spatial and temporal dimensions. Global methods (Fig. 1(a)) compress both dimensions into a single embedding (Bain et al., 2021; Luo et al., 2022); while efficient, this discards fine-grained visual evidence and entangles multiple semantic concepts. Local matching methods (Ma et al., 2022; Wang et al., 2022a; Reddy et al., 2025) retain multiple features for finer cross-modal interaction between textual token features and frame-level video features, which typically preserve the temporal dimension while compressing each frame spatially (Fig. 1(b)). In this way, multiple semantic concepts in a frame are compressed into a single feature, whereas individual text tokens represent much more localized semantics. As a result, despite the use of multiple frame-level features, the two modalities

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

can remain mismatched in semantic granularity. Overall, existing T2VR methods often overlook the information-density asymmetry between text and video, resulting in visual matching units that are too coarse for precise token-level correspondence.

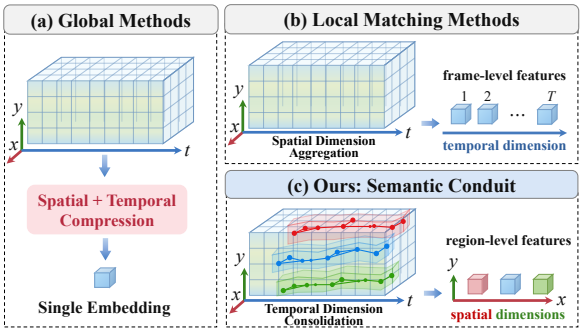


Figure 1: Comparison of video feature construction strategies for T2VR. (a) Global methods compress both spatial and temporal dimensions, entangling semantics. (b) Local methods keep the temporal dimension but compress each frame spatially, mixing intra-frame semantics. (c) Our method preserves spatial semantic diversity and consolidates temporal semantics into compact Semantic Conduits, aligning visual units with token-level text at a matched granularity.

To address this issue, we argue that a more appropriate video content compression strategy is to preserve the spatial dimension, thereby keeping localized visual evidence, and consolidate the temporal dimension by associating and aggregating the same semantics across frames (Fig. 1(c)). To this end, we propose a **Granularity-Aligned Retrieval** (GAR) paradigm that makes the organization of visual semantics better matched to fine-grained textual features. The paradigm is instantiated with **Semantic Conduit**, which aggregate related visual evidence across frames into a compact token, enabling a text token to interact with localized evidence from multiple frames through a shared semantic component. Specifically, within each frame, appearance-similar patches are aggregated into semantic prototypes, decomposing a semantically diverse frame into coherent visual units. These prototypes are then associated across frames, forming initial conduits whose connected components determine the conduit count and initialize a Gaussian mixture. Soft patch-to-conduit assignments then produce compact semantic conduit tokens that preserve localized spatial semantics while integrating temporal evidence across frames. A transformer contextualizes these tokens by modeling semantic relations and temporal dependencies, and MeanMaxSim matches each text token to its most similar conduit and averages the scores, yielding fine-grained word-to-conduit correspondence. Through this hierarchical patch–region–conduit construction, the dense video signal is compressed into compact semantic units whose granularity better matches token-level textual features. Moreover, conduit construction is query-independent, allowing video features to be precomputed for efficient retrieval.

Our contributions are as follows. (1) We propose a *granularity-aligned retrieval paradigm* (GAR) for T2VR, addressing the semantic-granularity mismatch caused by the information-density asymmetry between text and video. (2) We instantiate GAR with a hierarchical patch–region–conduit representation, which preserves intra-frame semantic diversity through spatial aggregation and consolidates inter-frame semantic evidence through temporal association, producing compact Semantic Conduit tokens for fine-grained cross-modal matching. (3) GAR achieves competitive results on MSR-VTT, MSVD, DiDeMo, and ActivityNet with ViT-B/32 and ViT-B/16 backbones.

2 Related Work

2.1 Text-to-Video Retrieval

Text-to-video retrieval has developed from aggregating pretrained expert features to adapting large-scale vision-language representations. Collaborative Experts (Liu et al., 2020) and MMT (Gabeur et al., 2020) combine multiple sources of video information, while T2VLAD (Wang et al., 2021) learns global and local cross-modal alignment. Large-scale video-language pretraining builds on HowTo100M (Miech et al., 2019): ActBERT models global actions together with local regions (Zhu & Yang, 2020), VideoCLIP uses contrastive pretraining with retrieved hard negatives (Xu et al., 2021), and support-set bottlenecks encourage representations to capture semantics shared across examples (Patrick et al., 2021). Frozen in Time (Bain et al., 2021) jointly trains on image-text and video-text pairs. End-to-end frameworks have further explored sparse clip sampling (Lei et al., 2021), region-entity prompting (Li et al., 2022), unified video-language backbones (Wang et al.,

2023), systematic pretraining recipes (Cheng et al., 2023), and hierarchical temporal objectives (Ye et al., 2023). Following CLIP (Radford et al., 2021), CLIP4Clip (Luo et al., 2022) studies temporal aggregation for retrieval, and CLIP-ViP (Xue et al., 2023) adapts image-text pretraining to video. These methods establish transferable encoders and mechanisms for integrating frame information.

Fine-grained alignment gives query-relevant visual content a more direct role in similarity computation. TACo emphasizes content words in a token-aware contrastive objective (Yang et al., 2021), BridgeFormer (Ge et al., 2022) learns local video-text associations through a multiple-choice pretext task, and Cap4Video (Wu et al., 2023) incorporates auxiliary captions at the data, feature, and score levels. X-Pool (Gorti et al., 2022) uses text-conditioned pooling, X-CLIP (Ma et al., 2022) combines multiple alignment granularities, and disentangled representation learning (Wang et al., 2022a) explores token-wise interactions. Late interaction was introduced for efficient fine-grained text retrieval by ColBERT (Khattab & Zaharia, 2020), and FILIP (Yao et al., 2022) applies token-wise maximum similarity to image-text pretraining. Video-ColBERT (Reddy et al., 2025) is closely related to our matching design: its contextualized late interaction retains token-level comparisons while encoding text and video separately. We use this separation to match text tokens with visual evidence consolidated through region discovery and cross-frame association. A Transformer contextualizes the pooled conduit tokens before their interaction with text, making cross-frame semantic components the visual units exposed to matching.

2.2 Vision Token Compression

Visual token reduction changes the amount and organization of information available to subsequent computation. EViT (Liang et al., 2022) reorganizes tokens according to attention, Adaptive Token Sampling (Fayyaz et al., 2022) varies the retained token count by input, TPS (Wei et al., 2023) combines pruning with token squeezing, and ToMe (Bolya et al., 2023) merges similar tokens using bipartite matching. For video retrieval, TS2-Net (Liu et al., 2022) uses token shift and selection, while CenterCLIP (Zhao et al., 2022) clusters tokens within video segments to obtain representative features. These approaches establish selection, merging, and clustering as useful tools for reducing visual redundancy. Our hierarchy uses aggregation to construct the semantic units exposed to late interaction, with separate stages for spatial consolidation and association across frames.

Adaptive aggregation also offers alternatives to retaining a uniform patch grid. TokenLearner (Ryoo et al., 2021) learns to extract a small set of tokens from visual inputs, GroupViT (Xu et al., 2022) hierarchically merges visual tokens into irregular semantic groups, and Slot Attention (Locatello et al., 2020) groups perceptual features through iterative competitive attention. Our region discovery uses connected components of an appearance-affinity graph, retains their patch memberships, and selects regions by patch count under a patch-grid-dependent budget. Adjacent-frame matching of temporally encoded semantic prototypes produces initial conduits that initialize a video-specific Gaussian mixture for soft cross-frame association. The resulting Semantic Conduits connect the spatial support of grouped observations to appearance summaries used for word-to-conduit matching.

3 Method

3.1 Overview

The overall framework of GAR is illustrated in Fig. 2. It contains two independent encoding branches for video and text, and performs fine-grained cross-modal matching on their outputs. The text branch encodes a query into contextualized token features. The video branch hierarchically transforms dense video signals into compact semantic units: patch features are first aggregated within each frame into region-level semantic prototypes, which are then associated across frames to initialize *Semantic Conduits*. Soft assignments pool evidence from all video patches into compact conduit tokens. A Transformer further models semantic relations and temporal dependencies among these conduits to produce contextualized video features. Given a video $\mathcal{V}$, we sample T frames and use a visual encoder to extract P patch-level features from each frame. Let $\mathbf{x}_{t,p} \in \mathbb{R}^d$ denote the feature of patch p in frame t, where $1 \leq t \leq T$, $1 \leq p \leq P$, and d is the feature dimension, we elaborate on each module below.

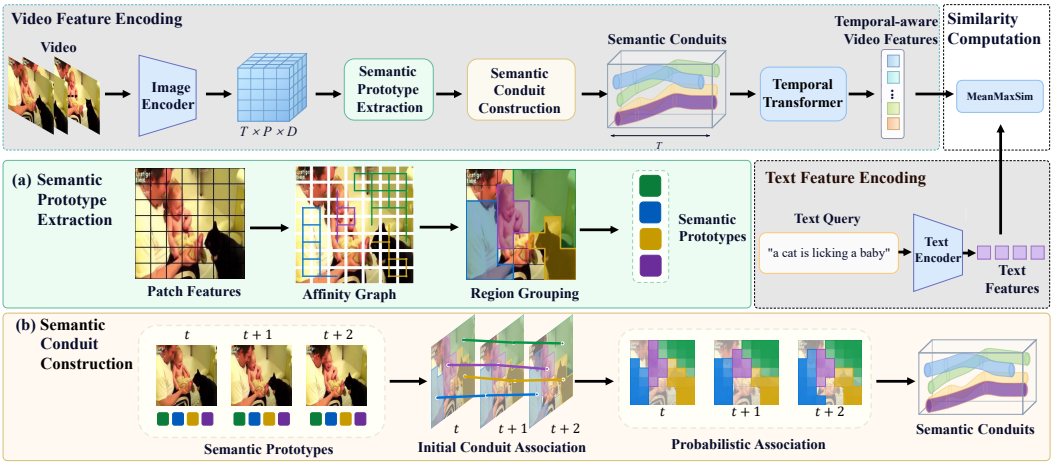


Figure 2: Overview of the proposed framework. The video branch hierarchically transforms patch features into Semantic Conduits through (a) *Semantic Prototype Extraction*, which groups appearance-similar patches within each frame into semantic prototypes, and (b) *Semantic Conduit Construction*, which links prototypes across adjacent frames to initialize conduits and then assigns all video patches to them probabilistically. A Temporal Transformer contextualizes the conduit tokens, which are then aligned with independently encoded text features via MeanMaxSim.

3.2 Semantic Prototype Extraction

This stage aims to identify distinct visual concepts within each frame as prototypes for cross-frame association. For each frame t, we construct a patch affinity graph whose nodes are all patch features in this frame, and an undirected edge is added between two patches p and q if their cosine similarity reaches the affinity threshold τ_c: $\cos(\mathbf{x}_{t,p}, \mathbf{x}_{t,q}) \geq \tau_c$. The resulting graph therefore connects appearance-similar patches within frame t, and its connected components form candidate regions. Since grouping depends on appearance rather than spatial adjacency, non-adjacent patches within the same frame may belong to the same candidate region.

Per frame, we retain candidate regions containing at least $K_s = \lceil \alpha P \rceil$ patches and keep at most the $K_r = \lceil \beta P \rceil$ largest eligible regions. If none qualifies, we retain the largest candidate region. For each selected region, we compute the mean of the features of all its member patches as its semantic prototype. After selection, frame t has $R_t \leq K_r$ prototypes, denoted by $\mathcal{P}_t = \{\mathbf{p}_{t,1}, \ldots, \mathbf{p}_{t,R_t}\}$, which are then associated across frames.

3.3 Semantic Conduit Construction

Given the retained prototypes from all frames, this stage associates them across frames to gather visual evidence belonging to the same semantic component into a Semantic Conduit.

Temporal encoding and conduit initialization. To associate prototypes across frames, we add the fixed sinusoidal frame encoding $\mathbf{e}_t \in \mathbb{R}^d$ to each prototype:

$$\tilde{\mathbf{p}}_{t,k} = \mathbf{p}_{t,k} + \mathbf{e}_t. \tag{1}$$

For each $\mathbf{p}_{t,k} \in \mathcal{P}_t$ with $t < T$, we find the prototype in $\mathcal{P}_{t+1}$ with the highest cosine similarity between their temporally augmented features. If this similarity exceeds τ_a, the two prototypes are connected. Each prototype has at most one outgoing link, while multiple prototypes may share the same target.

These links form a graph G_V whose nodes are the frame-indexed prototypes in $\mathcal{P}_1, \ldots, \mathcal{P}_T$ and edges $\mathcal{E}_V$ are the accepted cross-frame links. We extract its connected components, which partition the prototypes into groups of cross-frame associations. We require a connected component to cover at least two distinct sampled frames when $T = 12$, or at least four when $T = 64$. Components below this threshold are discarded. If none meets this threshold, we retain the component covering

the most frames to ensure the video still has a conduit. The retained M components for video $\mathcal{V}$ are denoted by $\mathcal{T}_1, \ldots, \mathcal{T}_M$, where each $\mathcal{T}_m$ is the initial **Semantic Conduit**: a set of frame-specific prototypes associated across time, as illustrated in Fig. 2(b). The values of the grouping and association hyperparameters τ_c, τ_a, α, and β are reported in Appendix A.2.

Probabilistic cross-frame association. To include visual evidence beyond the selected prototypes, we fit an M-component diagonal-covariance Gaussian mixture to the features of *all* TP video patches. Each patch receives its frame encoding, $\tilde{\mathbf{x}}_{t,p} = \mathbf{x}_{t,p} + \mathbf{e}_t$. We initialize the m-th Gaussian mean with the average temporally augmented prototype feature in the corresponding graph component $\mathcal{T}_m$. The mixture assigns every patch a responsibility for every conduit:

$$r_{t,p,m} = \frac{\pi_m \mathcal{N}(\tilde{\mathbf{x}}_{t,p}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}{\sum_{j=1}^{M} \pi_j \mathcal{N}(\tilde{\mathbf{x}}_{t,p}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \tag{2}$$

Here π_m, $\boldsymbol{\mu}_m$, and the diagonal covariance $\boldsymbol{\Sigma}_m$ are the fitted mixture weight, mean, and covariance estimated from all patches, and $\mathcal{N}$ is the Gaussian density. Eq. (2) then gives the soft assignment probability $r_{t,p,m} \geq 0$ that patch p in frame t belongs to conduit m, with $\sum_m r_{t,p,m} = 1$. This soft assignment allows patches not initially selected into any prototype to still contribute to the conduits, mitigating the impact of hard-assignment errors.

Conduit token aggregation. After constructing the patch-to-conduit assignments, we obtain each Semantic Conduit representation by aggregating the original patch features according to their assignment weights as follows:

$$\mathbf{o}_m = \frac{\sum_{t=1}^{T} \sum_{p=1}^{P} r_{t,p,m} \mathbf{x}_{t,p}}{\sum_{t=1}^{T} \sum_{p=1}^{P} r_{t,p,m}}, \qquad \mathbf{O} = [\mathbf{o}_1, \ldots, \mathbf{o}_M]. \tag{3}$$

Consequently, the multiple Semantic Conduits preserve the semantic diversity of the video along the spatial dimension, while each conduit consolidates video semantics along the temporal dimension. The resulting sequence $\mathbf{O}$, serving as the conduit representation of video $\mathcal{V}$, is then passed to the Temporal Transformer.

3.4 Temporal Transformer for Conduit Contextualization

To model relations among Semantic Conduits while preserving the temporal relationships established during conduit construction, we contextualize the pooled conduit tokens with a Transformer whose attention incorporates a temporal-overlap bias. Because graph links connect adjacent frames, each retained graph component spans a continuous interval. Let t_m^{start} and t_m^{end} be the first and last sampled-frame indices of $\mathcal{T}_m$. We use this initial interval as a temporal prior for conduit m:

$$\mathcal{F}_m = \{t \in \{1, \ldots, T\} : t_m^{\text{start}} \leq t \leq t_m^{\text{end}}\}. \tag{4}$$

The attention bias between conduits m and n is the temporal intersection over union of their initial intervals:

$$B_{mn} = \frac{|\mathcal{F}_m \cap \mathcal{F}_n|}{|\mathcal{F}_m \cup \mathcal{F}_n|}. \tag{5}$$

Greater temporal overlap gives a larger positive bias, while disjoint supports give zero bias. For one attention head, the output is

$$\mathbf{Z} = \text{softmax}\left(\frac{\mathbf{Q}_o \mathbf{K}_o^\top}{\sqrt{d_k}} + \mathbf{B}\right) \mathbf{V}_o, \tag{6}$$

where $\mathbf{Q}_o$, $\mathbf{K}_o$, and $\mathbf{V}_o$ are the head's query, key, and value projections of the conduit tokens, and d_k is the key dimension. The same bias matrix $\mathbf{B}$ is used across heads. This soft bias favors interactions between conduits whose initial intervals overlap, while allowing conduits with disjoint initial intervals to interact through their query-key compatibility. Taking $\mathbf{O}$ in Eq. 3 as input, the Transformer produces the contextualized sequence:

$$\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_M]. \tag{7}$$

Each output token corresponds to one Semantic Conduit and incorporates information from the others. These tokens provide the video features for fine-grained matching with text.

3.5 MEANMAXSIM MATCHING AND OPTIMIZATION

Using the contextualized conduit token embeddings $\mathbf{H}$ obtained above, we compute text-video similarity through localized interactions with text token embeddings. The text encoder independently produces contextualized features $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_L]$ for the L valid query tokens, excluding padding and sequence-boundary tokens. MeanMaxSim computes the cosine similarity between each valid text token, which may be a subword, and each conduit token, then averages the token-wise maxima:

$$s = \frac{1}{L} \sum_{\ell=1}^{L} \max_{1 \le m \le M} \frac{\mathbf{q}_\ell^\top \mathbf{h}_m}{\|\mathbf{q}_\ell\|_2 \|\mathbf{h}_m\|_2}. \tag{8}$$

The maximum selects the most relevant visual evidence for each token, and the mean normalizes over the valid query tokens. Conduits containing content absent from the query therefore do not lower the score merely because they lack a textual match.

For a batch of B paired examples, let s_{ij} denote the MeanMaxSim score between query i and video j. We optimize symmetric contrastive cross-entropy:

$$\mathcal{L} = -\frac{1}{2B} \sum_{i=1}^{B} \left[\log \frac{\exp(\gamma s_{ii})}{\sum_{j=1}^{B} \exp(\gamma s_{ij})} + \log \frac{\exp(\gamma s_{ii})}{\sum_{j=1}^{B} \exp(\gamma s_{ji})} \right]. \tag{9}$$

The two terms encourage retrieval of the paired video from each query and the paired text from each video, respectively. As in CLIP, the logit scale is $\gamma = \exp(g)$, with learnable g initialized to $\log(1/0.07)$. Graph construction and mixture fitting are detached during optimization. Retrieval gradients pass through the conduit Transformer and weighted patch pooling to the trainable visual features, and through the text features to the trainable text encoder.

At inference, candidate videos are ranked by their MeanMaxSim scores. Since prototype extraction, conduit construction, and contextualization depend only on the video, the features $\mathbf{H}$ can be precomputed and cached for retrieval.

3.6 DISCUSSION

In real-world text-to-video retrieval systems, video galleries routinely scale to billions of items (Wang et al., 2022b), making the size of cached representations and the cost of repeated query-video matching the dominant online costs. Consequently, the granularity of the cached matching units is not merely an accuracy trade-off but a first-order scalability constraint. Let T be the sampled-frame count, L the scored query-token count, d the feature dimension, M the conduit count, and N_q the number of learned concept queries in GLSCL. Table 1 compares the cached visual features and single query-video scoring costs of representative local methods, excluding the visual and text encoders.

GAR compresses all video patches into M cached conduits and matches each query token against these M vectors. In practice, $M < T$, as demonstrated in our experiments, it reduces both cached storage and repeated pairwise matching relative to frame-token methods, at the cost of one-time video-side construction. This analytical advantage does not imply a measured end-to-end speedup. Appendix A.3 details the construction costs, baseline scoring designs, and a fixed-setting example.

Table 1: Cached visual features and online scoring cost per query-video pair.

Method	Cached features	Pairwise scoring
DRL (WTI) (Wang et al., 2022a)	$O(Td)$	$O(LTd)$
X-CLIP (Ma et al., 2022)	$O((T+1)d)$	$O((T+1)(L+1)d)$
GLSCL (Zhang et al., 2025)	$O(Td)$	$O((T+N_q)d)$
Video-ColBERT (Reddy et al., 2025)	$O((2T+2)d)$	$O(L(2T+2)d)$
GAR (ours)	$O(Md)$	$O(LMd)$

4 EXPERIMENTS

We evaluate our framework on four text-to-video retrieval benchmarks: MSR-VTT (Xu et al., 2016), MSVD (Chen & Dolan, 2011), DiDeMo (Anne Hendricks et al., 2017), and ActivityNet (Krishna et al., 2017). We report Recall at K (R@K, in percent), mean rank (MnR), and nDCG@10 (abbreviated as nDCG) with ViT-B/32 and ViT-B/16 backbones. Dataset descriptions and detailed experimental settings are provided in Appendix A.

Table 2: Text-to-video retrieval on MSR-VTT and MSVD. GAR ranks in the top two for R@1 and leads among reported nDCG@10 scores in all settings. **Bold** and underlined mark the best and second-best results. Note that SigLIP4CLIP uses SigLIP-B/16, which outperformed CLIP-B/16 on these benchmarks in a prior backbone comparison (Reddy et al., 2025).

Method	MSR-VTT					MSVD				
	R@1↑	R@5↑	R@10↑	MnR↓	nDCG↑	R@1↑	R@5↑	R@10↑	MnR↓	nDCG↑
ViT-B/32										
Global matching										
CLIP4Clip-meanP (Luo et al., 2022)	43.1	70.4	80.8	16.2	-	46.2	76.1	84.6	10.0	-
CLIP4Clip-seqTransf (Luo et al., 2022)	44.5	71.4	81.6	15.3	-	45.2	75.5	84.3	10.3	-
CenterCLIP (Zhao et al., 2022)	44.2	71.6	82.1	15.1	-	47.6	77.5	86.0	9.8	-
PIG (Lan et al., 2025)	**48.6**	72.8	81.6	15.4	-	47.9	75.9	83.4	-	-
TempMe (Shen et al., 2025)	46.1	71.8	80.7	14.8	-	-	-	-	-	-
DAVRL (Bian et al., 2026)	46.2	71.8	81.7	14.6	-	-	-	-	-	-
Local matching										
X-CLIP (Ma et al., 2022)	46.1	73.0	83.1	13.2	-	47.1	77.8	-	9.5	-
DRL (Wang et al., 2022a)	47.4	74.6	83.8	12.8	-	**48.3**	**79.1**	**87.3**	-	-
GLSCL (Zhang et al., 2025)	48.1	73.9	82.3	13.8	-	47.5	76.3	86.1	9.8	-
Video-ColBERT (Reddy et al., 2025)	48.1	74.9	83.9	-	0.652	46.0	75.0	84.0	-	0.645
Ours	48.4	**75.1**	**84.0**	**12.4**	**0.657**	48.1	78.2	86.7	**9.1**	**0.651**
ViT-B/16										
Global matching										
SigLIP4CLIP-meanP (Reddy et al., 2025)	46.2	71.9	81.6	-	0.633	50.1	78.3	86.7	-	0.680
SigLIP4CLIP-seqTransf (Reddy et al., 2025)	45.7	71.0	80.0	-	0.623	47.4	76.6	85.0	-	0.659
MV-Adapter (Jin et al., 2024)	46.2	73.2	82.7	-	-	49.4	78.3	87.0	-	-
TempMe (Shen et al., 2025)	49.0	74.4	83.3	**11.9**	-	-	-	-	-	-
Local matching										
X-CLIP (Ma et al., 2022)	49.3	75.8	84.8	12.2	-	**50.4**	80.6	-	**8.4**	-
DRL (Wang et al., 2022a)	50.2	76.5	84.7	12.4	-	50.0	**81.5**	**89.5**	-	-
GLSCL (Zhang et al., 2025)	49.5	75.1	83.9	12.6	-	-	-	-	-	-
Video-ColBERT (Reddy et al., 2025)	51.0	77.1	85.5	-	0.677	50.2	79.6	87.8	-	0.683
Ours	**51.2**	**77.5**	**85.8**	**11.9**	**0.680**	50.3	79.8	88.3	8.5	**0.686**

4.1 Benchmark Results

Tables 2 and 3 compare GAR with global-matching and local-matching baselines across four benchmarks and two backbone settings, testing whether visual units aligned with text-token granularity can address the information-density asymmetry between video and language. We report TempMe's main LoRA variant; DAVRL and GLSCL results are without post-processing. GAR outperforms the global-matching group in nearly all comparable results and ranks first or second among local-matching methods on nearly every reported measure. Among methods reporting nDCG@10, GAR achieves the highest score in all eight settings. On DiDeMo and ActivityNet, which pair paragraph queries with videos sampled at 64 frames, GAR also attains the highest R@1 with both backbones. Paragraphs can refer to distinct entities and events across a video; these results are consistent with the value of preserving separate spatial concepts and consolidating their evidence across frames before matching them to individual text tokens.

Beyond accuracy, GAR also offers clear advantages in storage and online retrieval cost. On MSR-VTT with ViT-B/32, GAR stores 6.613 conduit tokens per video on average, compared with 13 visual tokens for X-CLIP and 26 for Video-ColBERT. This amounts to 13.23, 26.00, and 52.00 KiB of FP32 embeddings per video, respectively; at a reference query length of 32 tokens, scoring requires 211.6, 429, and 832 pairwise similarities. These analytical counts describe retrieval-stage storage and scoring, not measured end-to-end latency (Appendix A.3.1). Together with the benchmark results, they show that GAR can retain localized evidence for token-level alignment while using fewer cached tokens and similarity computations than these baselines.

4.2 Ablation Study

We examine GAR at two levels on MSR-VTT with ViT-B/32, with results in Tab. 4. Here, RG denotes region grouping, SC Semantic Conduit association, CT the conduit Transformer, TE temporal encoding, and SA soft assignment. Panel 4(a) progressively adds RG, SC, and CT to the baseline, while panel (b) removes TE, GMM refinement, or SA from the full model. Since SA relies on mixture responsibilities, removing GMM also disables SA.

Table 3: Text-to-video retrieval on DiDeMo and ActivityNet. GAR leads in R@1 and reported nDCG@10 in all settings. **Bold** and underlined mark the best and second-best results.

Method	DiDeMo					ActivityNet				
	R@1↑	R@5↑	R@10↑	MnR↓	nDCG↑	R@1↑	R@5↑	R@10↑	MnR↓	nDCG↑
ViT-B/32										
Global matching										
CLIP4Clip-meanP (Luo et al., 2022)	43.4	70.2	80.6	17.5	-	40.5	72.4	-	7.4	-
CLIP4Clip-seqTransf (Luo et al., 2022)	42.8	68.5	79.2	18.9	-	40.5	72.4	-	7.5	-
CenterCLIP (Zhao et al., 2022)	-	-	-	-	-	43.9	74.6	85.8	6.7	-
PIG (Lan et al., 2025)	41.2	67.9	80.5	-	-	-	-	-	-	-
TempMe (Shen et al., 2025)	48.0	72.4	81.8	13.7	-	44.9	**75.2**	85.5	6.8	-
DAVRL (Bian et al., 2026)	47.8	74.2	83.5	14.4	-	42.5	73.7	84.3	7.4	-
Local matching										
X-CLIP (Ma et al., 2022)	45.2	74.0	-	14.6	-	44.3	74.1	-	7.9	-
DRL (Wang et al., 2022a)	47.9	73.8	82.7	-	-	44.2	74.5	**86.1**	-	-
GLSCL (Zhang et al., 2025)	46.7	75.0	82.9	13.0	-	41.4	72.1	83.9	7.5	-
Video-ColBERT (Reddy et al., 2025)	48.2	75.1	83.7	-	0.654	45.5	74.6	85.5	-	0.645
Ours	**48.8**	**75.5**	**84.9**	**12.1**	**0.667**	**46.0**	**75.2**	85.8	**6.4**	**0.652**
ViT-B/16										
Global matching										
CLIP4Clip-meanP (Luo et al., 2022)	44.8	75.1	-	13.0	-	44.0	73.9	-	7.0	-
CLIP4Clip-seqTransf (Luo et al., 2022)	44.8	73.4	-	13.5	-	44.5	75.2	-	6.4	-
MV-Adapter (Jin et al., 2024)	44.3	72.1	80.5	-	-	42.9	74.5	85.7	-	-
PIG (Lan et al., 2025)	43.3	70.8	81.9	-	-	-	-	-	-	-
Local matching										
X-CLIP (Ma et al., 2022)	47.8	**79.3**	-	12.6	-	46.2	75.5	-	6.8	-
DRL (Wang et al., 2022a)	49.0	76.5	84.5	-	-	46.2	77.3	**88.2**	-	-
Video-ColBERT (Reddy et al., 2025)	51.9	78.3	85.6	-	0.682	50.6	78.0	87.9	-	0.685
Ours	**52.1**	78.4	**85.9**	**11.7**	**0.687**	**50.9**	**78.4**	88.0	**5.8**	**0.690**

For the cumulative-stage ablation, the Baseline directly uses frame-level visual features as video tokens. With RG, the retained semantic prototypes from all frames are directly used as video tokens for cross-modal matching. Adding SC further constructs Semantic Conduits across frames, whose resulting tokens are directly used for matching before CT is introduced. For the conduit-construction ablations, *w/o TE* removes temporal encoding, *w/o GMM + SA* removes GMM-based refinement and soft assignment, and *w/o SA* replaces soft assignment with hard assignment while retaining the fitted GMM. Detailed implementations of these variants are provided in the appendix.

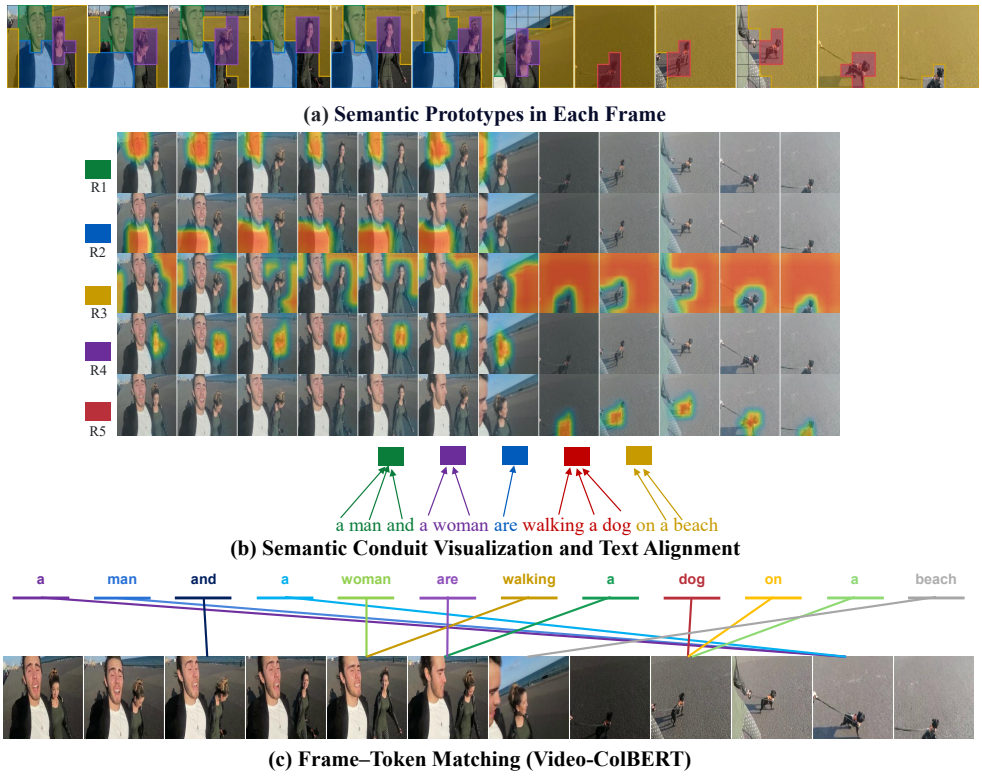
Table 4: GAR ablations on MSR-VTT.

(a) Cumulative stages Setting	RG	SC	CT	R@1↑
Baseline	×	×	×	43.8
+ RG	✓	×	×	44.9
+ SC	✓	✓	×	47.7
Full Model	✓	✓	✓	**48.4**
(b) Conduit construction Variant	TE	GMM	SA	R@1↑
w/o TE	×	✓	✓	48.1
w/o GMM + SA	✓	×	×	47.2
w/o SA	✓	✓	×	47.9
Full Model	✓	✓	✓	**48.4**

Panel (a) shows that each stage improves retrieval, with the largest increment coming from Semantic Conduit association: R@1 rises from 44.9% to 47.7% when it is added after region grouping. The conduit Transformer then raises R@1 to 48.4%. Panel (b) probes how the conduits are constructed. Removing temporal encoding reduces R@1 to 48.1%, and disabling soft assignment reduces it to 47.9%. Removing both GMM refinement and its dependent soft assignment lowers R@1 further to 47.2%. This last variant measures their joint effect, not the isolated contribution of GMM refinement.

4.3 Case Study

We select Video-ColBERT (Reddy et al., 2025) for this case study because it is a state-of-the-art local-matching method and one of the strongest local baselines in Tables 2 and 3. Like GAR, it uses token-wise late interaction, making the granularity of the visual units directly comparable. Figure 3 follows 12 sampled frames of a video in which a close-up of a man and woman gives way to a wider view of a dog on the beach. Panel (a) shows the frame-wise prototypes that initialize cross-frame



(a) Semantic Prototypes in Each Frame

(b) Semantic Conduit Visualization and Text Alignment

(c) Frame–Token Matching (Video-ColBERT)

Figure 3: Case study for "a man and a woman are walking a dog on a beach." (a) Frame-wise semantic prototypes. (b) GMM patch-to-conduit responsibilities for five Semantic Conduits; warmer colors indicate higher assignment probabilities, and arrows show text-token matches. (c) Schematic of Video-ColBERT (Reddy et al., 2025) token-to-frame matching.

association. Panel (b) displays the GMM responsibilities $r_{t,p,m}$ for five conduits: R1 emphasizes the man's face, R2 his torso, R3 the beach and surrounding background, R4 the woman, and R5 the dog, which becomes more prominent in the later frames. The query words "man," "woman," "dog," and "beach" align with R1, R4, R5, and R3, respectively. The illustrated "walking" match also selects R5, providing action-related evidence without precisely localizing motion. These heatmaps show soft patch assignments, not hard object masks.

Video-ColBERT encodes each frame as one visual feature and applies a temporal Transformer to obtain contextualized frame features. It matches text tokens against both feature streams; panel (c) illustrates these frame-derived matches. A frame feature can contain the people, dog, and beach together, even after temporal contextualization. GAR instead pools related patches into distinct conduits, allowing the text to select more specific visual evidence as the scene and camera view change. An additional example is shown in Appendix C.

5 Conclusion

We introduced Granularity-Aligned Retrieval (GAR) to address the semantic-granularity mismatch arising from the information-density asymmetry between text and video. Its hierarchical patch-region-conduit representation preserves localized evidence within frames while consolidating related semantics across time into compact Semantic Conduits. A Transformer contextualizes these units before MeanMaxSim aligns them with text tokens. Across four benchmarks and two backbones, GAR achieves strong retrieval results. Its query-independent video representation can be precomputed; on MSR-VTT with ViT-B/32, our analytical comparison also shows lower retrieval-stage embedding storage and scoring costs than X-CLIP and Video-ColBERT. These findings support preserving spatial detail while consolidating temporal evidence for efficient, fine-grained retrieval.

AI USE STATEMENT

ChatGPT assisted with manuscript revision, title suggestions, drafting the conclusion, interpreting the supplied result tables, explaining the stated pooling equations and computational costs, and checking presentation and submission requirements. The authors are responsible for the final text, claims, and experimental evidence.

REFERENCES

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Max Bain, Arsha Nagrani, Gul Varol, and Andrew Zisserman. Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1708–1718, Montreal, QC, Canada, October 2021. IEEE. ISBN 978-1-6654-2812-5. doi: 10.1109/ICCV48922.2021.00175.

Ziyi Bian, Yadan Luo, Fangzhi Zhu, Xiangbo Shu, and Zheng Zhang. Dual-attention video representation learning for parameter efficient text-video retrieval. *IEEE Transactions on Multimedia*, 28:7204–7216, 2026. doi: 10.1109/TMM.2026.3673509.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JroZRaRw7Eu.

David Chen and William Dolan. Collecting Highly Parallel Data for Paraphrase Evaluation. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea (eds.), *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 190–200, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

Feng Cheng, Xizi Wang, Jie Lei, David Crandall, Mohit Bansal, and Gedas Bertasius. Vindlu: A recipe for effective video-and-language pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10739–10750, June 2023.

Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *European conference on computer vision*, pp. 396–414. Springer, 2022.

Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal Transformer for Video Retrieval. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV 2020*, volume 12349, pp. 214–229. Springer International Publishing, Cham, 2020. ISBN 978-3-030-58547-1 978-3-030-58548-8. doi: 10.1007/978-3-030-58548-8_13.

Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. Bridging video-text retrieval with multiple choice questions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16167–16176, June 2022.

Satya Krishna Gorti, Noel Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. X-Pool: Cross-Modal Language-Video Attention for Text-Video Retrieval. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4996–5005, New Orleans, LA, USA, June 2022. IEEE. ISBN 978-1-6654-6946-3. doi: 10.1109/CVPR52688.2022.00495.

Xiaojie Jin, Bowen Zhang, Weibo Gong, Kai Xu, Xueqing Deng, Peng Wang, Zhao Zhang, Xiaohui Shen, and Jiashi Feng. Mv-adapter: Multimodal video transfer learning for video text retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27144–27153, June 2024.

Omar Khattab and Matei Zaharia. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, pp. 39–48, New York, NY, USA, July 2020. Association for Computing Machinery. ISBN 978-1-4503-8016-4. doi: 10.1145/3397271.3401075.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Bangxiang Lan, Ruobing Xie, Ruixiang Zhao, Xingwu Sun, Zhanhui Kang, Gang Yang, and Xirong Li. Hybrid-tower: Fine-grained pseudo-query interaction and generation for text-to-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24497–24506, October 2025.

Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7331–7341, June 2021.

Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven C.H. Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4953–4963, June 2022.

Youwei Liang, Chongjian GE, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. EVit: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=BjyvwnXXVn_.

Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use What You Have: Video Retrieval Using Representations From Collaborative Experts, February 2020.

Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. TS2-Net: Token Shift and Selection Transformer for Text-Video Retrieval. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, volume 13674, pp. 319–335. Springer Nature Switzerland, Cham, 2022. ISBN 978-3-031-19780-2 978-3-031-19781-9. doi: 10.1007/978-3-031-19781-9_19.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-Centric Learning with Slot Attention. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11525–11538. Curran Associates, Inc., 2020.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. CLIP4Clip: An empirical study of CLIP for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, October 2022. ISSN 09252312. doi: 10.1016/j.neucom.2022.07.028.

Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-CLIP: End-to-End Multi-grained Contrastive Learning for Video-Text Retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, pp. 638–647, New York, NY, USA, October 2022. Association for Computing Machinery. ISBN 978-1-4503-9203-7. doi: 10.1145/3503161.3547910.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2630–2640, Seoul, Korea (South), October 2019. IEEE. ISBN 978-1-7281-4803-8. doi: 10.1109/ICCV.2019.00272.

Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander G Hauptmann, Joao F. Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *International Conference on Learning Representations*, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763. PMLR, July 2021.

Arun Reddy, Alexander Martin, Eugene Yang, Andrew Yates, Kate Sanders, Kenton Murray, Reno Kriz, Celso M. de Melo, Benjamin Van Durme, and Rama Chellappa. Video-colbert: Contextualized late interaction for text-to-video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19691–19701, June 2025.

Michael Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. TokenLearner: Adaptive Space-Time Tokenization for Videos. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12786–12797. Curran Associates, Inc., 2021.

Leqi Shen, Tianxiang Hao, Tao He, Sicheng Zhao, Yifeng Zhang, Pengzhang Liu, Yongjun Bao, and Guiguang Ding. TempMe: Video temporal token merging for efficient text-video retrieval. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/98a5c0470e57d518ade4e56c6ee0b363-Paper-Conference.pdf.

Jinpeng Wang, Yixiao Ge, Rui Yan, Yuying Ge, Kevin Qinghong Lin, Satoshi Tsutsui, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. All in one: Exploring unified video-language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6598–6608, June 2023.

Qiang Wang, Yanhao Zhang, Yun Zheng, Pan Pan, and Xian-Sheng Hua. Disentangled Representation Learning for Text-Video Retrieval, March 2022a.

Xiaohan Wang, Linchao Zhu, and Yi Yang. T2VLAD: Global-Local Sequence Alignment for Text-Video Retrieval. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5075–5084, Nashville, TN, USA, June 2021. IEEE. ISBN 978-1-6654-4509-2. doi: 10.1109/CVPR46437.2021.00504.

Xun Wang, Bingqing Ke, Xuanping Li, Fangyu Liu, Mingyu Zhang, Xiao Liang, and Qiushi Xiao. Modality-balanced embedding for video retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieva*, pp. 2578–2582, 2022b.

Siyuan Wei, Tianzhu Ye, Shen Zhang, Yao Tang, and Jiajun Liang. Joint Token Pruning and Squeezing Towards More Aggressive Compression of Vision Transformers. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2092–2101, Vancouver, BC, Canada, June 2023. IEEE. ISBN 979-8-3503-0129-8. doi: 10.1109/CVPR52729.2023.00208.

Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. Cap4Video: What Can Auxiliary Captions Do for Text-Video Retrieval? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10704–10713, 2023.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. VideoCLIP: Contrastive Pre-training for Zero-shot Video-Text Understanding. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6787–6800, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.544.

Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18134–18144, June 2022.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5288–5296, 2016.

Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. CLIP-vip: Adapting pre-trained image-text model to video-language alignment. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=GNjzMAgawq.

Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11562–11572, October 2021.

Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: Fine-grained interactive language-image pre-training. In *International Conference on Learning Representations*, 2022.

Qinghao Ye, Guohai Xu, Ming Yan, Haiyang Xu, Qi Qian, Ji Zhang, and Fei Huang. Hitea: Hierarchical temporal-aware video-language pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15405–15416, October 2023.

Haonan Zhang, Pengpeng Zeng, Lianli Gao, Jingkuan Song, Yihang Duan, Xinyu Lyu, and Heng Tao Shen. Text-video retrieval with global-local semantic consistent learning. *IEEE Transactions on Image Processing*, 34:3463–3474, 2025. doi: 10.1109/TIP.2025.3574925.

Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. CenterCLIP: Token Clustering for Efficient Text-Video Retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 970–981, Madrid Spain, July 2022. ACM. ISBN 978-1-4503-8732-3. doi: 10.1145/3477495.3531950.

Linchao Zhu and Yi Yang. ActBERT: Learning Global-Local Video-Text Representations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8743–8752, Seattle, WA, USA, June 2020. IEEE. ISBN 978-1-7281-7168-5. doi: 10.1109/CVPR42600.2020.00877.

A Experimental Details

A.1 Datasets

We evaluate our method on four text-to-video retrieval benchmarks: MSR-VTT, MSVD, DiDeMo, and ActivityNet Captions. MSR-VTT and MSVD use sentence-to-video retrieval, while DiDeMo and ActivityNet Captions use paragraph-to-video retrieval.

MSR-VTT. MSR-VTT (Xu et al., 2016) contains 10,000 video clips, each annotated with 20 captions, for a total of 200,000 video-caption pairs. We use the Training-9K split, comprising 9,000 videos and 180,000 captions for training, and evaluate on the 1K-A test split of 1,000 videos.

MSVD. MSVD (Chen & Dolan, 2011) contains 1,970 short YouTube video clips with approximately 80,000 English descriptions, or roughly 40 descriptions per video. We use 1,200 videos for training, 100 for validation, and 670 for testing.

DiDeMo. For DiDeMo (Anne Hendricks et al., 2017), our retrieval protocol uses 8,391 training videos and 1,004 test videos. Each video has approximately four temporally localized captions. We concatenate the captions associated with a video to form a paragraph query for paragraph-to-video retrieval.

ActivityNet Captions. For ActivityNet Captions (Krishna et al., 2017), we use 10,009 videos for training and 4,917 videos from the `val_1` split for retrieval evaluation. We concatenate each video's temporally localized event captions to form a paragraph query.

A.1.1 Evaluation Protocols

We evaluate text-to-video retrieval using Recall at K (R@K for $K \in \{1, 5, 10\}$), mean rank (MnR), and normalized discounted cumulative gain at rank 10 (nDCG@10, abbreviated as nDCG in the tables). R@K is the percentage of text queries whose paired video appears among the top K retrieved videos. MnR is the mean rank of the paired videos over all queries; nDCG@10 evaluates ranking quality within the top ten results. For DiDeMo and ActivityNet Captions, we follow the established paragraph-to-video retrieval protocol (Lei et al., 2021): the temporally localized captions associated with each video are concatenated into a single paragraph query, which is matched against whole videos. Higher R@K and nDCG values are better; lower MnR values are better.

A.2 Implementation Details

Grouping and retrieval parameters. We set the intra-frame patch-affinity threshold to $\tau_c = 0.7$ and the adjacent-frame association threshold to $\tau_a = 0.75$. For the minimum region support and per-frame region budget, respectively, we use $\alpha = \beta = 0.1$ with ViT-B/32 and $\alpha = \beta = 0.05$ with ViT-B/16. MeanMaxSim introduces no temperature or directional-weight hyperparameters. Following CLIP's logit-scale parameterization, the contrastive scale is $\gamma = \exp(g)$, where g is learnable and initialized to $\log(1/0.07)$ (equivalently, γ is initially $1/0.07$).

Table 5: Optimization settings shared across datasets.

Setting	Value
Learning-rate schedule	Linear
Warmup proportion	10%
CLIP parameter learning rate	10^{-7}
Conduit-token Transformer learning rate	10^{-4}
Optimizer	Adam
Adam betas	$\beta_1 = 0.9$, $\beta_2 = 0.98$
Adam ϵ	10^{-6}
Weight decay	0.01
Maximum gradient norm	1

Network architecture and optimization. For the ViT-B/32 and ViT-B/16 variants, the video and query encoders are initialized from the corresponding CLIP visual and text encoders, respectively. During fine-tuning, we freeze the CLIP positional embeddings, visual patch embeddings, and text

Table 6: Dataset-specific training settings.

Dataset	Backbone	Batch size	Epochs
MSR-VTT	ViT-B/32	256	5
	ViT-B/16	128	5
MSVD	ViT-B/32	256	5
	ViT-B/16	128	5
DiDeMo	ViT-B/32	64	20
	ViT-B/16	64	20
ActivityNet Captions	ViT-B/32	64	20
	ViT-B/16	64	20

token embeddings. The remaining CLIP parameters are trained with a learning rate of 10^{-7}, while the conduit-token Transformer uses a learning rate of 10^{-4}. The Transformer has four layers, a width of 512, eight attention heads, and a feed-forward hidden dimension of 2,048 (an MLP ratio of 4). Table 5 summarizes the optimization settings, and Table 6 lists the dataset-specific batch sizes and training epochs.

Temporal encoding and Gaussian mixture fitting. We use a fixed sinusoidal temporal encoding for each sampled frame. Let p_t denote the frame's position in the sampled sequence. For $i = 0, \ldots, d/2 - 1$, its coordinates are

$$e_{t,2i} = \sin\Big(\frac{p_t}{10000^{2i/d}}\Big), \qquad e_{t,2i+1} = \cos\Big(\frac{p_t}{10000^{2i/d}}\Big). \tag{10}$$

The same encoding is added to every patch and retained semantic prototype from frame t. We fit one diagonal-covariance Gaussian mixture per video using `sklearn.mixture.GaussianMixture` in scikit-learn v1.6.1. The mixture is fitted to all TP temporally augmented patch features; the retained graph components determine M and provide the initial component means through `means_init`. Thus, for ViT-B/32 with $T = 12$ and $P = 49$, fitting uses 588 patch observations rather than at most 60 retained region prototypes. We set `n_components=` M and `covariance_type=diag`. The remaining parameters use the library defaults: `reg_covar=` 10^{-6}, `tol=` 10^{-3}, `max_iter=` 100, `n_init=` 1, and `init_params=kmeans`. The fitted posterior probabilities provide the patch-to-conduit responsibilities.

Text preprocessing. We use the tokenizer and special-token mappings of the corresponding CLIP model. Each text sequence begins with `<|startoftext|>` and ends with `<|endoftext|>`. We set the text sequence length to 32 for MSR-VTT and MSVD, and to 64 for DiDeMo and ActivityNet Captions.

Video preprocessing. We sample frames uniformly across each video. Each selected frame is resized directly to 224×224 pixels without preserving its aspect ratio or applying a center crop. We sample 12 frames per video for MSR-VTT and MSVD, and 64 frames for DiDeMo and ActivityNet Captions.

A.3 Representation Granularity and Computational Cost

Let T be the number of sampled frames, P the number of patch tokens per frame, and d the feature dimension. We distinguish the minimum patch-support threshold K_s from the per-frame region-count budget K_r. In our experiments, we set $K_s = \lceil \alpha P \rceil$ and $K_r = \lceil \beta P \rceil$, using the fractions specified in Section A.2. At an input resolution of 224×224, ViT-B/32 gives $P = 7 \times 7 = 49$ and $K_s = K_r = 5$, while ViT-B/16 gives $P = 14 \times 14 = 196$ and $K_s = K_r = 10$. A candidate region is eligible when its patch count is at least K_s; at most K_r eligible regions are retained per frame. If none meets the support threshold, we retain the largest region as a fallback. With R_t retained regions in frame t, the total number of region observations is $N_R = \sum_{t=1}^{T} R_t \le TK_r$. We retain graph components covering at least two sampled frames when $T = 12$, or at least four when $T = 64$, using the longest component as a fallback when needed. After removing short components, the adjacent-frame association graph has M retained components; these determine the number of

Semantic Conduits. Hence $M \leq N_R$, and the conduit-token Transformer processes one token per component.

The stages that construct and contextualize conduit tokens are performed once per video, independently of the query. Excluding CLIP feature extraction, computing all pairwise patch affinities within frames costs $O(TP^2d)$. Adjacent-frame region matching costs $O\Big(d\sum_{t=1}^{T-1} R_t R_{t+1}\Big)$, followed by $O(N_R + |\mathcal{E}_V|)$ for graph connected components, where $\mathcal{E}_V$ is the set of accepted association edges. For a diagonal-covariance Gaussian mixture with M components, each fitting iteration evaluates patch responsibilities in $O(TPMd)$ time. Pooling all patch features into conduit tokens also costs $O(TPMd)$. Each layer of the conduit-token Transformer costs $O(M^2d + Md^2)$ for attention and feature projections; our model uses four such layers. Writing I for the number of GMM iterations and $J = 4$ for the Transformer depth, the video-only cost after visual feature extraction is therefore $O(TP^2d + TK_r^2d + ITPMd + TPMd + J(M^2d + Md^2))$, plus $O(N_R + |\mathcal{E}_V|)$ for connected components. This cost is incurred once when a video enters the retrieval collection.

After video encoding, the contextualized conduit embeddings can be cached using $O(Md)$ feature values per video. For a query with L valid text tokens, scoring one query-video pair forms LM token similarities in $O(LMd)$ time; MeanMaxSim adds $O(LM)$ work to compute the row-wise maxima and their mean. A patch-level late-interaction representation retaining all TP visual tokens would instead form LTP similarities. Thus the ratio TP/M describes the reduction in pairwise similarity computations, not an end-to-end runtime speedup: total latency also depends on video encoding, mixture fitting, and the distribution of M across videos.

A.3.1 Comparison with Local Retrieval Methods

The symbolic comparison in Table 1 uses the same T, L, and d for each method. DRL's weighted token-wise interaction (Wang et al., 2022a) compares each of L text tokens with T frame tokens, giving $O(LTd)$ work per query-video pair and $O(Td)$ cached visual features. Its additional video-token weights require $O(T)$ storage, while text-token weights can be computed once per query. X-CLIP (Ma et al., 2022) forms video-sentence, video-word, sentence-frame, and frame-word scores: in total $1 + L + T + LT = (T + 1)(L + 1)$ dot products, or $O((T + 1)(L + 1)d)$ per pair. The T frame vectors and one pooled video vector use $O((T + 1)d)$ storage. Its attention-over-similarity reduction adds $O(LT + L + T)$ work per pair, while its three-layer temporal Transformer costs $O(3(T^2d + Td^2))$ once per video beyond the frame encoder. Video-ColBERT (Reddy et al., 2025) separately matches text tokens to T static frame vectors and $T + 2$ contextualized vectors, including two visual expansion tokens. It therefore stores $O((2T+2)d)$ feature values and computes $O(L(2T + 2)d)$ work per pair; L here includes query expansion tokens when used. Its four-layer temporal Transformer processes $T + 2$ visual inputs once per video at $O(4((T+2)^2d + (T+2)d^2))$ cost.

For N_{vid} videos with a common conduit count M, GAR requires $O(N_{\text{vid}}Md)$ cached feature values and $O(N_{\text{vid}}LMd)$ work for an exhaustive query. For varying counts M_i, these become $O(d\sum_i M_i)$ and $O(Ld\sum_i M_i)$. The smaller repeated cost follows when $M < T$; the one-time construction work and encoder costs must be considered separately when evaluating end-to-end latency.

For a concrete reference setting, we quantify embedding storage and token-pair similarity counts on MSR-VTT with ViT-B/32, $T = 12$ sampled frames, and feature dimension $d = 512$. Our mean conduit count is $\overline{M} = 6.613$. For an average of $\overline{N}$ stored visual tokens, uncompressed FP32 embedding storage and similarity counts at a reference query length $L_{\text{ref}} = 32$ are

$$S_{\text{FP32}} = \frac{4d\overline{N}}{1024} \quad \text{KiB/video}, \qquad C_{\text{sim}} = L_{\text{ref}}\overline{N} \quad \text{similarities/query-video}. \tag{11}$$

Substituting $\overline{N} = \overline{M}$ gives $S_{\text{FP32}} = 13.226 \approx 13.23$ KiB per video and $C_{\text{sim}} = 211.616 \approx 211.6$ similarities per query-video pair.

Video-ColBERT (Reddy et al., 2025) scores queries against both static frame features and temporally contextualized features. We derive its costs from the published configuration: 12 static frame tokens and 14 temporal output tokens, including the two visual expansion tokens, give 26 stored visual tokens. At the same dimension, precision, and reference query length, this yields 52.00 KiB per video and 832 token similarities. X-CLIP (Ma et al., 2022) combines video-sentence, video-word,

sentence-frame, and frame-word scores. For $T = 12$ frames and $L_{\text{ref}} = 32$ text tokens, these comprise $1 + L_{\text{ref}} + T + L_{\text{ref}}T = 429$ similarity scores per query-video pair, before its attention-based aggregation. Its released implementation uses the 12 original CLIP frame features for local matching and one pooled, temporally encoded video feature for global and cross-grained matching. Caching these 13 visual vectors requires $13 \times 512 \times 4/1024 = 26.00$ KiB per video in FP32. These reference counts use 32 scored query tokens; GAR's actual valid query length varies, so 211.6 is not an empirical mean over queries. They exclude index metadata and offline video encoding and do not measure end-to-end latency.

B Additional Ablation Studies

B.1 Implementation of Ablation Variants

We provide the implementation details of the ablation variants in Table 4. For the cumulative-stage ablation, the Baseline directly uses frame-level visual features as video tokens for cross-modal matching. For *+RG*, all retained semantic prototypes across frames are directly used as video tokens, without cross-frame association. For *+SC*, the complete Semantic Conduit Construction is applied, but the conduit Transformer is omitted; the resulting conduit tokens are directly used for cross-modal matching.

For the conduit-construction ablations, *w/o TE* removes temporal encoding while keeping the remaining construction unchanged. In *w/o GMM + SA*, GMM refinement and posterior-based soft assignment are removed. Conduit membership is therefore determined solely by the connected components obtained from adjacent-frame prototype association, and the retained prototypes within each component are average-pooled to form its conduit token. In *w/o SA*, the GMM is retained but soft assignment is replaced by hard assignment. Specifically, each patch is assigned to the component with the highest posterior responsibility,

$$\hat{m}_{t,p} = \arg\max_{m} r_{t,p,m}, \tag{12}$$

and the patches assigned to each component are average-pooled to form its conduit token. All other training and retrieval settings are kept identical to the Full Model.

B.2 Effect of Conduit Transformer Depth

We vary the number of conduit-token Transformer layers while keeping the remaining configuration fixed. The zero-layer setting directly matches text tokens to the pooled conduit tokens without Transformer contextualization. Table 7 reports retrieval results on MSR-VTT with ViT-B/32.

Table 7: Effect of conduit-token Transformer depth on MSR-VTT with ViT-B/32. Recall values are percentages; nDCG denotes nDCG@10. The best result in each column is shown in bold.

Layers	R@1↑	R@5↑	R@10↑	nDCG↑
0	47.7	73.9	81.8	0.645
2	47.8	74.2	82.5	0.648
4	48.4	**75.1**	**84.0**	**0.657**
8	**48.7**	74.6	82.9	0.655

Contextualizing the conduit tokens improves all four metrics relative to the zero-layer setting. Four layers give the highest R@5, R@10, and nDCG@10, whereas eight layers give the highest R@1, exceeding the four-layer setting by 0.3 percentage points. We use four layers in the main experiments because they provide the strongest results on three of the four reported metrics.

B.3 Sensitivity to Region Selection Parameters

We vary α and β over $\{0.06, 0.10, 0.14\}$ while keeping the remaining settings fixed. Figure 4 shows that the highest R@1, 48.5%, occurs at $\alpha = 0.06$ and $\beta = 0.14$. This setting retains more conduits on average, increasing the number of cached visual tokens and token-wise similarities at retrieval. We therefore use $\alpha = \beta = 0.10$ in the main ViT-B/32 experiments: it achieves 48.4% R@1, only

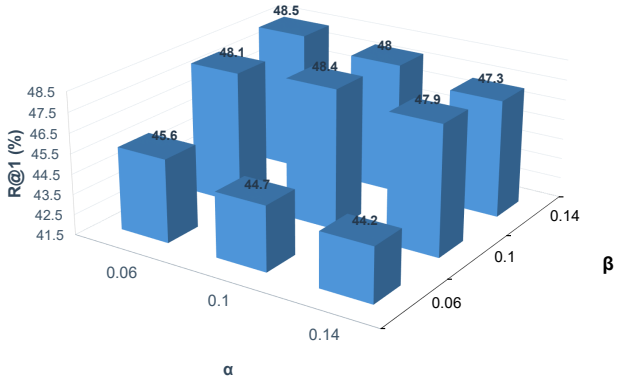


Figure 4: Sensitivity to the minimum region-support fraction α and per-frame region-budget fraction β on MSR-VTT with ViT-B/32. The bars report R@1 (%); the vertical axis begins at 41.5%.

0.1 percentage points below the maximum, with a smaller average conduit count M. A tighter region budget ($\beta = 0.06$) yields substantially lower R@1 across the tested α values, indicating that excessive compression can discard useful visual evidence. Overall, the selected setting balances retrieval accuracy and representation compactness.

C Additional Case Study

Figure 5 presents a second example with the query "a cat is licking a baby." The frame-wise prototypes in panel (a) seed cross-frame association. In the GMM responsibility maps of panel (b), R1 mostly covers the background, R2 the adult and supporting arm, R3 the cat, and R4 the baby. The query words "cat" and "licking" match R3, while "baby" matches R4. The "licking" match identifies relevant cat-side evidence rather than pinpointing the moment of contact. Panel (c) illustrates Video-ColBERT's token-to-frame matches for the same video; each matched frame feature includes multiple scene elements, whereas the conduits retain separate evidence for the cat and baby.

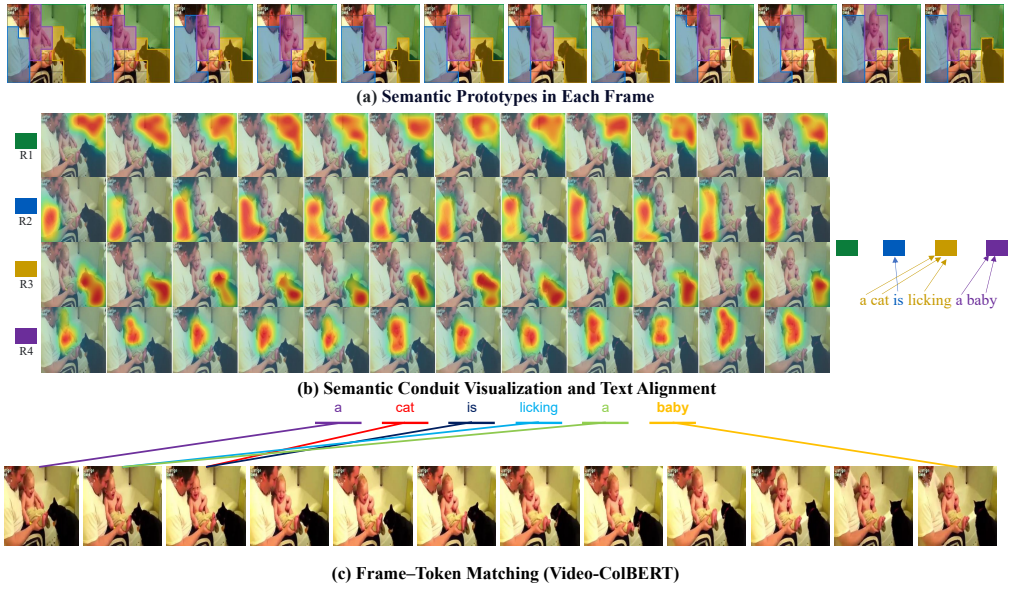


Figure 5: Additional case study for "a cat is licking a baby." (a) Frame-wise semantic prototypes. (b) GMM patch-to-conduit responsibilities for R1–R4; warmer colors indicate higher probabilities, and arrows show text-token matches. (c) Schematic of Video-ColBERT (Reddy et al., 2025) token-to-frame matching.