ECLIPSE: Efficient Long-range Video Retrieval using Sight and Sound

Yan-Bo Lin Jie Lei Mohit Bansal Gedas Bertasius
UNC Chapel Hill

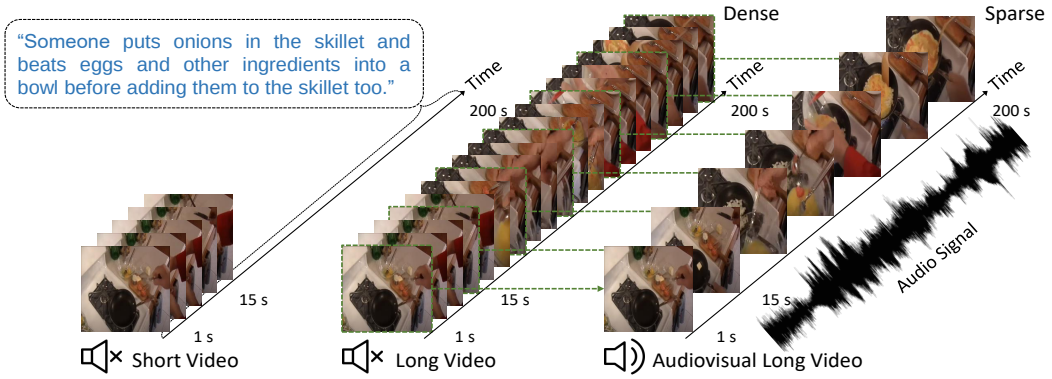

Figure 1. Comparison of different high-level frameworks for long-range text-to-video retrieval. Most traditional text-to-video retrieval methods (**Leftmost Column**) are designed for short videos (e.g., 5-15 seconds in duration). Adapting these approaches to several-minute long videos by stacking more input frames (**Middle Column**) is impractical due to excessive computational cost. Instead, our proposed framework operates on sparsely sampled video frames and dense audio cues, which are cheaper to process (**Rightmost Column**). In addition to being more efficient, our framework also achieves higher text-to-video retrieval accuracy than standard video-only approaches.

Abstract

We introduce an audiovisual method for long-range text-to-video retrieval. Unlike previous approaches designed for short video retrieval (e.g., 5-15 seconds in duration), our approach aims to retrieve minute-long videos that capture complex human actions. One challenge of standard video-only approaches is the large computational cost associated with processing hundreds of densely extracted frames from such long videos. To address this issue, we propose to replace parts of the video with compact audio cues that succinctly summarize dynamic audio events and are cheap to process. Our method, named ECLIPSE *(Efficient CLIP with Sound Encoding), adapts the popular CLIP model to an audiovisual video setting, by adding a unified audiovisual transformer block that captures complementary cues from the video and audio streams. In addition to being* $2.92\times$ *faster and* $2.34\times$ *memory-efficient than long-range video-only approaches, our method also achieves better text-to-video retrieval accuracy on several diverse long-range video datasets such as ActivityNet, QVHighlights, YouCook2, DiDeMo and Charades.*

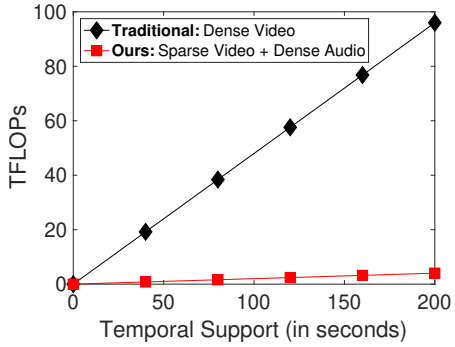

Figure 2. We can scale our audiovisual ECLIPSE framework to long videos more efficiently than dense video-only approaches.

1. Introduction

Fueled by the growing availability of video data, the last few years have witnessed remarkable progress in the area of text-to-video retrieval [1–3]. However, modern video retrieval systems are predominantly designed for very short videos (e.g., 5-15 seconds in length). In contrast, the majority of real-world videos often capture complex human actions, which may last several minutes or even hours.

Among prior vision-and-language methods [1, 4], CLIP [5] stands out as one of the most widely adopted

models. Several recent approaches extended the original CLIP method [5] to video [6] by independently processing individual video frames and then averaging their predictions across time. However, these prior approaches are often impractical in the long-range video setting because of the large computational cost required to process hundreds or even thousands of densely extracted video frames (See Figure 2). Furthermore, we note that while video modality is rich in the information it stores, it also has high informational redundancy (i.e., the video content often changes little in neighboring frames).

Based on this motivation, we introduce ECLIPSE, an Efficient CLIP with Sound Encoding. Instead of processing many densely-extracted frames from a long video (the middle column in Figure 1), our framework leverages complementary audio and video cues by operating on sparsely sampled video frames accompanied by dense audio (the rightmost column in Figure 1).

In summary, our contributions are threefold. First, we propose ECLIPSE, an audiovisual adaptation of CLIP that leverages complementary video and audio cues for long-range video retrieval. Second, we demonstrate that compared to long-range video-only approaches, our audiovisual framework leads to better video retrieval results at a reduced computational cost. Lastly, we provide comprehensive ablation studies investigating the success factors of ECLIPSE to inspire more future work in this area.

2. ECLIPSE: Efficient CLIP with Sound Encoding

2.1. Obtaining Multimodal Input Embeddings

Video Patch Decomposition. Following the ViT [7], we decompose each frame into N non-overlapping patches, each of size $P \times P$, and flatten these patches into vectors $\mathbf{x}_{(p,t)} \in \mathbb{R}^{3P^2}$ where $p = 1, \ldots, N$ denotes spatial locations and $t = 1, \ldots, T$ indicates a frame index. A specialized CLS token $\mathbf{v}_{cls}^{(0)}$ is prepended to the visual sequence of each frame. Finally, the embeddings $\mathbf{V}^{(0)} \in \mathbb{R}^{T \times (N+1) \times d}$ are used as visual inputs to our ECLIPSE model.

Audio Embeddings. Given an audio spectrogram $Z_t \in \mathbb{R}^{M \times C}$, an audio encoder maps it into audio embeddings $\mathbf{A}_t^{(0)} \in \mathbb{R}^d$ for each timestep $t = 1 \ldots T$ where as before, T denotes the number of video frames.

Text Embeddings. We use a pretrained CLIP [5] text encoder to embed a textual video description $y = (y_1, \ldots, y_L)$ into a textual embedding $\mathbf{g} \in \mathbb{R}^d$ where $\mathbf{g}$ corresponds to the CLS token of a given text sequence.

2.2. Audiovisual Attention Block

Although videos contain rich information, they are also redundant and costly to process. In contrast, audio is more compact and cheaper. Thus, we propose an audiovisual attention block that gradually incorporates relevant audio cues into the visual representation. Our audiovisual attention block consists of three distinct attention schemes: (i) spatial visual attention, (ii) audio-to-video attention, and (iii) video-to-audio attention. We next describe each of these attention schemes in more detail.

Multi-Head Self-Attention. All of our three attention schemes are implemented using a standard multi-head self-attention:

$$\text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V}, \quad (1)$$

where $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ are the query, key and value matrices obtained using learnable projection weights $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V \in \mathbb{R}^{d \times d}$ respectively. With this formal description of the MHA function, we can now proceed to the definitions of the three attention schemes in our audiovisual attention block.

Spatial Attention. In order to preserve the pretrained network structure of CLIP, we use an identical spatial attention scheme as in their model. Intuitively, the spatial attention enables our model to obtain discriminative frame-level representation by aggregating relevant information from the visual tokens in the individual video frames. We can implement this scheme using our previously defined MHA function as:

$$\mathbf{S}_t^{(\ell)} = \text{MHA}(\mathbf{V}_t^{(\ell-1)}, \mathbf{V}_t^{(\ell-1)}, \mathbf{V}_t^{(\ell-1)}) + \mathbf{V}_t^{(\ell-1)}. \quad (2)$$

Here, $\mathbf{S}_t^{(\ell)} \in \mathbb{R}^{(N+1) \times d}$ is our newly computed spatial self-attention representation for frame t, and $\mathbf{V}_t^{(\ell-1)}$ is a visual patch representation for frame t from the previous transformer layer $l-1$, which is used as input to the the transformer layer l. Note that in the spatial self-attention, the multi-head self-attention is applied independently for each of T video frames. As discussed above, this enables us to preserve the network structure of the original CLIP model, which is essential for good text-to-video retrieval performance. For brevity, we omit the layer normalization operation, which is applied to $\mathbf{V}_t^{(\ell)}$ before feeding it to the spatial attention block. The right part of Figure 3 provides a visual illustration of where spatial attention fits within our audiovisual attention block

Audio-to-Video Attention (A2V). To efficiently incorporate temporal audio cues into static video frame representation, we use an audio-to-video (A2V) attention mechanism, which is also illustrated in the right part of Figure 3 (labeled as Cross-Attn A2V module). This operation can be written as:

$$\mathbf{V}_t^{(\ell)} = \text{MHA}(\mathbf{S}_t^{(\ell-1)}, \mathbf{A}^{(\ell-1)}, \mathbf{A}^{(\ell-1)}) + \mathbf{S}_t^{(\ell-1)}. \quad (3)$$

Here, $\mathbf{A}^{(\ell-1)} \in \mathbb{R}^{T \times d}$ depicts our previously defined audio representation at layer $l-1$, and $\mathbf{S}_t^{(\ell-1)} \in \mathbb{R}^{(N+1) \times d}$ de-

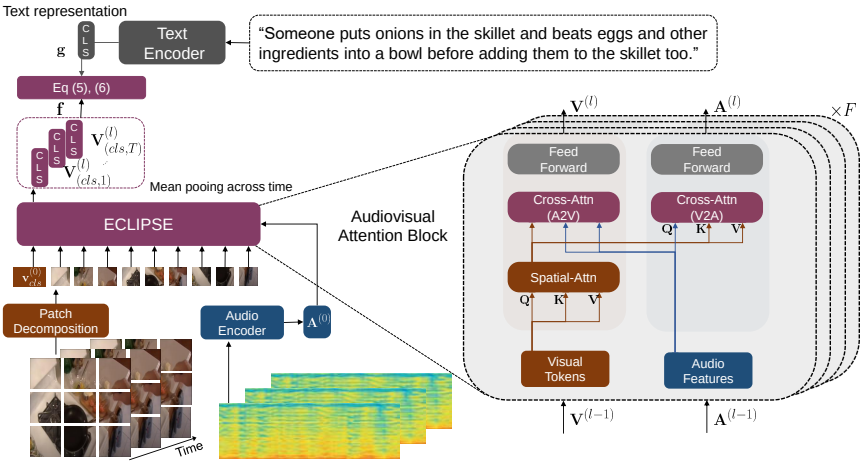

Figure 3. We adapt CLIP [5] to long-range text-to-video retrieval by adding an efficient audiovisual attention block into the Transformer architecture. First, we obtain fixed dimensional text, audio, and visual feature embeddings. Afterward, the visual and audio embeddings are fed into our ECLIPSE audiovisual backbone, which injects relevant audio information to video and vice-versa. This is accomplished using a dual-pathway audiovisual attention block (illustrated on the right), which is stacked on top of each other F times. Afterward, the audiovisual video segments are aggregated using temporal pooling, and the model is optimized by maximizing the similarity between audiovisual and textual embeddings using a contrastive loss function.

notes a spatial video representation at timestep t computed using our previously defined spatial attention block. Intuitively, the new visual representation $\mathbf{V}_t^{(\ell)}$ is computed as a weighted summation of the audio features, which enables the model to incorporate long-range audio cues into the visual features. Furthermore, because the audio representation is compact, the operation above can be implemented efficiently.

Video-to-Audio Attention (V2A). Conversely, to inject rich visual information into compact audio features, we use a video-to-audio (V2A) attention mechanism (illustrated in Figure 3 as Cross-Attn V2A module). We implement this attention scheme as:

$$\mathbf{A}_t^{(\ell)} = \text{MHA}(\mathbf{A}_t^{(\ell-1)}, \mathbf{S}_t^{(\ell-1)}, \mathbf{S}_t^{(\ell-1)}) + \mathbf{A}_t^{(\ell-1)}. \quad (4)$$

At a high-level, the operation above computes a new audio feature representation for each timestep t as a weighted combination of all the visual token features at tiemstep t. This allows us to improve the richness and expressivity of the audio representation.

Final Audiovisual Representation. Following CLIP4Clip [6], we stack our audiovisual attention block F times (F typically being set to 12). Afterward, we perform temporal pooling over the CLS tokens across all video frames, to obtain the final audiovisual representation $\mathbf{f} \in \mathbb{R}^d$.

2.3. Loss Function

We use a contrastive video-to-text matching loss to train our model. The similarity between text and video is computed using a normalized dot product between the two embeddings $\mathbf{f}$ and $\mathbf{g}$. We consider the matching text-video pairs in a given batch as positive samples and all the other pairs in that same batch as negative samples. To train our model, we minimize the sum of the video-to-text and text-to-video matching losses:

$$\mathcal{L}_{\text{v2t}} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(\mathbf{f}_i^\top \mathbf{g}_i)}{\sum_{j=1}^{B} \exp(\mathbf{f}_i^\top \mathbf{g}_j)}, \quad (5)$$

$$\mathcal{L}_{\text{t2v}} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(\mathbf{g}_i^\top \mathbf{f}_i)}{\sum_{j=1}^{B} \exp(\mathbf{g}_i^\top \mathbf{f}_j)}, \quad (6)$$

where $\mathbf{f}_i$ and $\mathbf{g}_j$ are the normalized embeddings of i-th video and the j-th text sequence, respectively.

3. Experimental Setup

3.1. Downstream Datasets

We evaluate ECLIPSE on five diverse datasets that contain long-range videos: ActivityNet Captions [8], QVHighlights [9], DiDeMo [10], YouCook2 [11], and Charades [12].

3.2. Evaluation Metrics

We use standard video retrieval evaluation metrics [4, 6] such as text-to-video $R@1$, $R@5$, $R@10$, video-to-text $R@1$, and mean rank (MnR) to validate the effectiveness of our ECLIPSE model.

Table 1. Our results on ActivityNet Caption, QVHighlights (QVH), YouCook2 (YC2), Charades and DiDeMo using the R@1 metric. ECLIPSE outperforms all prior approaches while also being more efficient.

Method	Pretrain	Frames	ActivityNet [8]	QVH [9]	DiDeMo [10]	YC2 [11]	Charades [12]	GFLOPs
ClipBERT [4]	C+G	32	21.3	43.2	20.4	29.8	6.7	-
FiT [2]	CW	32	-	55.0	34.6	32.2	11.9	-
CLIP4Clip [6]	W	64	40.7	68.5	43.4	36.7	12.6	836
CLIP4Clip*	W	96	41.7	70.2	42.5	37.6	13.9	1251
ECLIPSE	W+V	32	**42.3**	**70.8**	**44.2**	**38.5**	**15.7**	**827**

4. Results and Analysis

4.1. Results on Long-range Datasets

We validate our approach on five long-range video datasets: ActivityNet Caption [8], QVHighlights [9] (QVH), DiDeMo [10], YouCook2 [11] (YC2), and Charades [12].

Our results in Table 1, demonstrate that ECLIPSE outperforms all prior methods on all four datasets by a substantial margin. In particular, ECLIPSE achieves 1.6%, 2.3%, 0.8%, 1.8%, and 2.9% better R@1 accuracy than the strong CLIP4Clip baseline at a roughly similar computational cost of $\approx$ 830 GFLOPS on QVHighlights, DiDeMo, YouCook2, and Charades respectively.

Furthermore, ECLIPSE also outperforms our stronger 96-frame CLIP4Clip* method on all four datasets while operating on $3\times$ less frames (i.e., 32 vs. 96), thus, being considerably more efficient (827 vs. 1251 GFLOPs).

Based on these results, we can make the following two observations: (i) ECLIPSE is highly effective when applied to long videos spanning at least several minutes, (ii) our method generalizes to a diverse set of videos (e.g., cooking, fitness instructions, daily activity, news videos, etc.), (iii) audio cues can be used to replace many redundant and costly to process video frames.

5. Conclusions

In this paper, we present a novel audiovisual framework, ECLIPSE, for long-range video retrieval. By replacing costly and redundant parts of the video, with compact audio cues, ECLIPSE efficiently processes long-range videos while also obtaining better performance than standard video-only methods. Our audiovisual framework is (i) flexible, (ii) fast, (iii) memory-efficient, and (iv) it achieves state-of-the-art results on five diverse long-range video benchmarks. In the future, we plan to extend our method to other multimodal video understanding tasks such as video question answering and video captioning.

References

[1] S. Liu, H. Fan, S. Qian, Y. Chen, W. Ding, and Z. Wang, "Hit: Hierarchical transformer with momentum contrast for video-text retrieval," in *ICCV*, 2021. 1

[2] M. Bain, A. Nagrani, G. Varol, and A. Zisserman, "Frozen in time: A joint video and image encoder for end-to-end retrieval," in *ICCV*, 2021. 1, 4

[3] V. Gabeur, C. Sun, K. Alahari, and C. Schmid, "Multi-modal transformer for video retrieval," in *ECCV*, 2020. 1

[4] J. Lei, L. Li, L. Zhou, Z. Gan, T. L. Berg, M. Bansal, and J. Liu, "Less is more: Clipbert for video-and-language learning via sparse sampling," in *CVPR*, 2021. 1, 3, 4

[5] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *ICML*, 2021. 1, 2, 3

[6] H. Luo, L. Ji, M. Zhong, Y. Chen, W. Lei, N. Duan, and T. Li, "CLIP4Clip: An empirical study of clip for end to end video clip retrieval," *arXiv Preprint*, 2021. 2, 3, 4

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021. 2

[8] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, "Dense-captioning events in videos," in *ICCV*, 2017. 3, 4

[9] J. Lei, T. L. Berg, and M. Bansal, "Qvhighlights: Detecting moments and highlights in videos via natural language queries," in *NeurIPS*, 2021. 3, 4

[10] L. Anne Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with natural language," in *ICCV*, 2017. 3, 4

[11] L. Zhou, C. Xu, and J. J. Corso, "Towards automatic learning of procedures from web instructional videos," in *AAAI*, 2018. 3, 4

[12] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *ECCV*, 2016. 3, 4