Text Query to Web Image to Video: A Comprehensive Ad-hoc Video Search

Nhat-Minh Nguyen[1,2], Tien-Dung Mai[1,2], and Duy-Dinh Le[1,2]

[1] University of Information Technology, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam
21521135@gm.uit.edu.vn
{dungmt,duyld}@uit.edu.vn

Abstract. In this study, we propose a novel approach for Ad-hoc Video Search that leverages the power of image search engines to synthesize query images for corresponding textual sentence query. Existing methods primarily rely on pre-trained language-image models to extract features from textual queries and video keyframes of video segments. While recent approaches using generative models to generate visual representations based on text descriptions show promise, they are limited by diversity, authenticity, speed, and hardware requirements. In contrast, our proposed method leverages the vast and diverse image database available on the Internet through image search engines to directly synthesize query images based on input text descriptions. Moreover, to enhance computational efficiency, each video segment is represented by only single keyframe. Specifically, we use only two general-purpose multimodal models for extracting feature embeddings for textual queries, query images, and keyframes. To return a list of relevant video segments for each query, we compute the weighted average similarity between each keyframe and both the textual query and query images. Experiments conducted on the TRECVID dataset (V3C2) and main set of textual queries from 2022 and 2023 demonstrate the efficiency of our method.

Keywords: Ad-hoc Video Search · Image Search Engines · Keyframe

1 Introduction

The Ad-hoc Video Search task aims to model the end-user search use case, where textual sentence queries are used to search for video segments containing specific persons, objects, activities, locations, or combinations thereof. Given a textual query describing objects in a frame, the system is expected to return relevant video segments containing frames that match the query, the example is shown in Fig. 1. Unlike conventional text-query based video retrieval, which only allows structured searches based on predefined information such as title, date, time, and video description, these systems are unable to understand the semantic content of users' queries. Ad-hoc Video Search, on the other hand, allows for more flexible queries that can handle complex descriptions and better grasp the semantic meaning of both the queries and the video content.

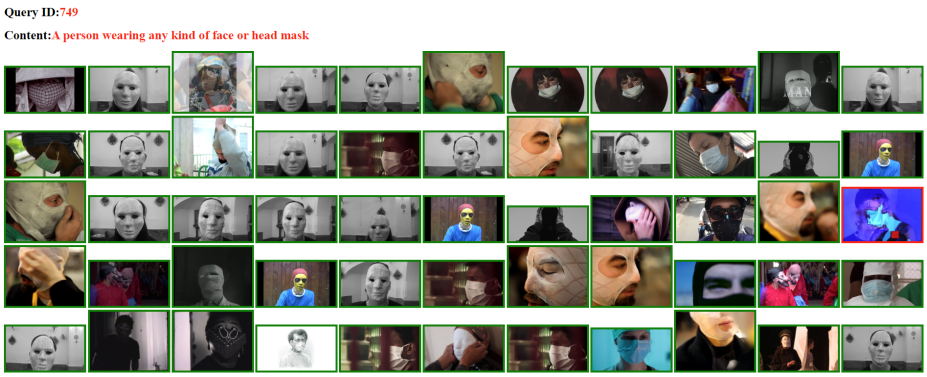

Fig. 1: Visualize the retrieved results with the displayed image being the keyframe of each resulting video segment using our proposal method. Query content is *"A person wearing any kind of face or head mask"*. Keyframes with green borders represent video segments that match the ground truth, while those with red borders do not.

Recent successful methods for Ad-hoc Video Search (AVS) often utilize multimodal language-image models. For instance, Waseda_Meisei_Softbank [12] achieved high precision on the TRECVID2022 AVS task by combining multiple language-image models (VSE++, GSMN, CLIP [8], SLIP [7]) and a Diffusion Model. They calculated cosine similarity between frame videos and text queries, and between generated images and video frames, use every 10 frames of each video segment. Waseida_Meisei_Softbank at TRECVID2023 AVS [13], they implemented the latest pre-trained models from OpenCLIP and applied Query Expansion techniques using ChatGPT, achieving the second position (mean xinfAP = 0.285) in the query result evaluation rankings.

WHU-NERCMS [3] achieved the highest precision on the TRECVID2023 AVS task by weighting the similarity of different language-image models between textual queries and keyframe images. They used various versions of CLIP [8], SLIP [7], BLIP [5], BLIP-2 [4], and LaCLIP [1] models, and employed a Diffusion Model to generate images and calculate cosine similarity with keyframes. With LaCLIP being a CLIP model retrained on extended corpus from Large Language Model to augment text understanding capabilities.

Both research groups used Diffusion models to generate images, which can provide diverse content based on text descriptions. However, this method is time-consuming, computationally expensive, and may lack diversity and authenticity due to limitations in the training dataset.

Our approach also uses general-purpose language-image models to extract embedding feature vectors for images and text. However, instead of using Diffusion models, we leverage image search engines and the vast amount of images on the Internet to synthesize relevant images for text queries. This saves time and computational costs while providing more diverse and realistic images. We scrape result images from image search engines, extract one keyframe per video

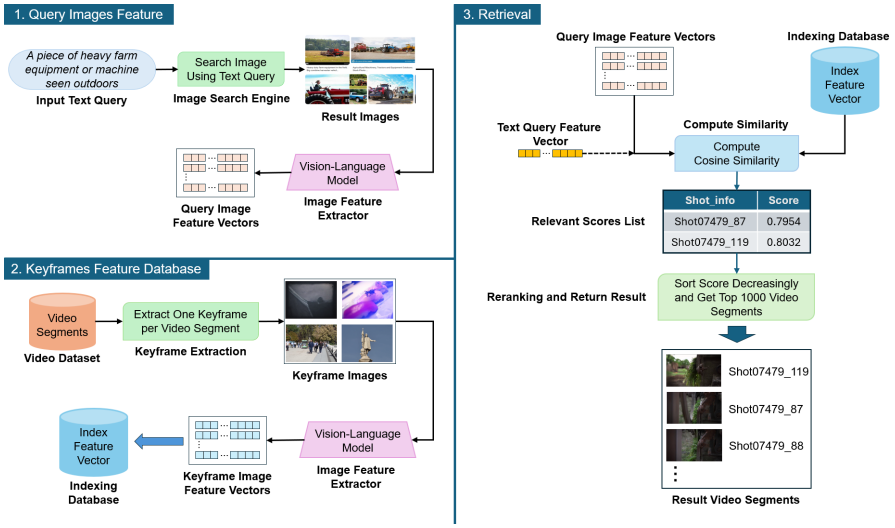

Fig. 2: Proposed Method Framework for Single Model

segment, and only use pre-trained BEiT-3 [14] and CLIP [8] models to extract feature vectors for keyframe images and search engine result images. We then query keyframe images using a combination of query images and text queries with the extracted feature vectors.

2 Related Work

2.1 Language-Image Models

In the Ad-hoc Video Search (AVS) task, the end-user provides a text query describing the desired video content. Traditional deep learning models like VGG-16 and ResNet are insufficient for this task due to its complexity. General-purpose language-image models have proven essential for AVS, as demonstrated by the success of Waseida_Meisei_Softbank [12] (1st position in 2022) and WHU-NERCMS [3] (1st position in 2023) on the TRECVID V3C2 dataset.

These models take images or text as input and, through an encoding process, extract feature vectors that represent them in a shared vector space. The proximity of vectors in this space reflects their semantic similarity. This property enables the AVS problem to be approached as a Text-based Image Retrieval task, where the retrieved images are keyframes. This approach is widely adopted by research groups, with popular models like CLIP, BLIP, BLIP-2, and SLIP demonstrating notable success.

2.2 Text-to-Image Generative Models

The inherent subjectivity in image perception and description among individuals can lead to a semantic gap between human and machine understand-

ing of visual content. Textual descriptions of images may not fully capture the nuances of human interpretation, suggesting that image-based querying could potentially bridge this gap. This rationale underlies the utilization of Diffusion models by aforementioned research groups to generate images from input text queries. Subsequently, features are extracted from these generated images and compared with video frame features using the corresponding model. The integration of text queries with relevant images generated by Diffusion models has demonstrated promising results in enhancing video retrieval tasks, as evidenced by the experiments conducted by WHU-NERCMS [3].

2.3 Query Expansion

In the methods employed by the research group that secured the second position at TRECVID2023 AVS, query expansion techniques were utilized to enhance the capabilities of their query systems. Notably, Waseda_Meisei_Softbank [13] incorporated ChatGPT at TRECVID2023 AVS to generate 100 new queries derived from an original query. Given the semantic diversity inherent in language, multiple expressions can convey the same object, and employing diverse phrasing can lead to a more nuanced and comprehensive understanding. This approach aligns with the practices of popular search engines like Google, Bing, and Yahoo, which also leverage query expansion to improve search effectiveness. Consequently, query expansion has been demonstrated to enhance both the diversity and effectiveness of the image retrieval process.

3 Approach

The proposed method operates as follows: for each input text query q, we leverage image search engines to retrieve relevant images from the Internet, selecting the top K results as query images. Subsequently, we employ two general-purpose language-image models, CLIP and BEiT-3, to extract features for both the text query q and the K query images. For each model, the representative relevance score between video segment s_i and text query q is computed as a weighted average similarity between the text query q and the K query images, in conjunction with keyframe k_i. To determine the final relevance score between query q and video segment s_i, we calculate the weighted average of the relevance scores obtained from both models. From the list of relevance scores for N video segments with query q, we return the top S video segments exhibiting the highest keyframe similarity to the query. This entire process for single model is shown in Fig. 2.

3.1 Web Images

In addressing the Ad-hoc Video Search problem, leading research groups, including our own, recognize the critical role of synthesized images in enhancing retrieval effectiveness. Previous methods typically employ generative models to

produce visual representations based on input text queries. This technique has demonstrated significant efficacy, as the generated images, when combined with text query features, substantially improve the performance of retrieval systems. However, these generative models require substantial computational resources and time for both training and image generation. The images produced by generative models are not real-life photographs and may exhibit limited diversity in style and composition. This can potentially constrain the robustness and applicability of the retrieval system in diverse real-world scenarios.

In our proposed method, we utilize web images to enrich the representation of text queries. For each input text query, we employ popular image search engines, including Google Images, Bing Images, and Yahoo Images, to retrieve relevant images. Specifically, we automatically download the top K image results returned by the search engine, designating these images as "query images" (illustrated in the first part of Fig. 2).

The use of web images is motivated by their broad and diverse nature, which captures various aspects and interpretations of the text query. This diversity enhances the robustness of the query representation, enabling a more effective and nuanced video search. By integrating web images, we aim to bridge the semantic gap between textual descriptions and visual content, thereby improving the retrieval accuracy.

Unlike using generative models that necessitate extensive training and fine-tuning of models on large, domain-specific datasets, our method leverages existing, readily available web images. This substantially reduces the computational cost and time required for processing, making the system more scalable and adaptable to a wide range of queries without the need for continuous retraining.

In essence, the incorporation of web images in our Ad-hoc Video Search methodology not only enhances the semantic richness of text queries but also provides a practical, efficient, and scalable solution for improving video retrieval performance.

3.2 Keyframe Extraction

In our approach, keyframe extraction is crucial for the Ad-hoc Video Search process. We select a single keyframe from the middle of each video segment to serve as a visual summary of the segment's content. This strategy offers several advantages: computational efficiency, storage optimization, retrieval speed. Extracting only one keyframe per segment reduces the computational resources required for feature extraction, storage, and retrieval. Limiting the number of keyframes minimizes storage requirements, which is beneficial for large video collections. Fewer keyframes result in faster retrieval times, essential for real-time search applications.

The middle frame is chosen to balance simplicity and representativeness, capturing the central theme of the segment. These keyframes are then processed using pre-trained multimodal models, such as CLIP and BEiT-3, to obtain feature embeddings. These embeddings, combined with text query and query image embeddings, facilitate accurate and efficient video retrieval.

3.3 Feature Embeddings Extraction

Prior research has consistently demonstrated that the utilization of general-purpose multimodal models for feature extraction from images and text is a fundamental aspect of effective methodologies in this domain. In our proposed method, we also leverage a combination of language-image models to extract features for text queries, query images, and keyframes. However, in contrast to approaches that employ multiple configurations of different CLIP variants like SLIP, BLIP, and BLIP-2, or those that involve retraining models as in WHU-NERCMS [3] and Waseda_Meisei_Softbank [12, 13], our method exclusively utilizes two pre-trained multimodal models: CLIP (version CLIP ViT-B/32) and BEiT-3 (version BEiT-3_Large_Patch16_384_Coco_Retrieval).

3.4 Similarity Computation

After extracting features for the text query, query images, and keyframes, we calculate a representative value for the relevance score between the text query and each video segment through the keyframes. We use a combination of both text queries and query images to enhance retrieval effectiveness, specifically as follows:

First, we compute the Cosine similarity between the text query q and each keyframe of every video segment, with $\mathbf{t}$ and $\mathbf{k}_i$ represent the feature vector of the text query and the feature vector of the i-th keyframe, respectively:

$$text\_kf_i = CosineSimilarity(\mathbf{t},\ \mathbf{k}_i) \tag{1}$$

To reduce computational cost, instead of calculating the similarity for all K query images got from the image search engine based on the text query, we applied K-Means Clustering to group the K feature embedding vectors of those images into C distinct clusters. Subsequently, we calculated the average cosine similarity between the C centroid vectors (representing the centers of the C clusters) and each keyframe.

Let $qrImg\_kf_i$ be the similarity score between K query images and i-th keyframe, $\mathbf{c}_j$ and $\mathbf{k}_i$ represent the vector of the j-th cluster centroid and the feature vector of the i-th keyframe, respectively:

$$qrImg\_kf_i = \frac{1}{C}\sum_{j=0}^{C-1} CosineSimilarity(\mathbf{c}_j, \mathbf{k}_i) \tag{2}$$

We combine both features from pre-trained language-image model CLIP and BEiT-3 models to calculate a representative value for the relevance score between the query and the keyframes. With the features extracted by each model, we calculate the similarity between the text query and the keyframes, as well as between the query images and the keyframes, as described above. With $CLIP\_text\_kf_i$, $CLIP\_qrImg\_kf_i$ being the similarity calculated using the feature vectors extracted by CLIP and $BEiT3\_text\_kf_i$, $BEiT3\_qrImg\_kf_i$ being the similarity calculated using the feature vectors extracted by BEiT-3 for

i-th keyframe following Eq. 1 and Eq. 2. Let α, β represent the respective weights of the similarity values derived from the CLIP model's features in the weighted average calculation of the similarity between the text query and keyframes, and between the query images and keyframes of the two models.

The weighted average similarity between the text query and the keyframe is calculated as follows:

$$avg\_text\_kf_i = \alpha \times CLIP\_text\_kf_i + (1-\alpha) \times BEiT3\_text\_kf_i \quad (3)$$

The weighted average similarity between the query images and the keyframe is calculated as follows:

$$avg\_qrImg\_kf_i = \beta \times CLIP\_qrImg\_kf_i + (1-\beta) \times BEiT3\_qrImg\_kf_i \quad (4)$$

From the calculated values of $avg\_text\_kf_i$ and $avg\_qrImg\_kf_i$, we proceed to calculate the weighted average of these two values to obtain the final similarity value, representing the relevance score of the i-th keyframe (and its corresponding i-th video segment s_i) to the text query. With φ representing the weight of $avg\_text\_kf_i$'s contribution to the relevance score, we can calculate this relevance score as follows:

$$relScore\_query\_kf_i = \varphi \times avg\_text\_kf_i + (1-\varphi) \times avg\_qrImg\_kf_i \quad (5)$$

With a dataset containing N video segments, corresponding to N extracted keyframes, we will have a list of N $relScore\_query\_kf$ values, representing the relevance score of N video segments to the text query q. We return the top S video segments with the highest relevance scores for each input text query q, $S \leq N$.

4 Experiments

4.1 Dataset

The video dataset employed in this study is V3C2, compiled by the National Institute of Standards and Technology (NIST). The V3C2 dataset (drawn from a larger V3C video dataset [10]) was adopted as a testing dataset. It is composed of 9760 Vimeo videos (1.6 TB, 1300 h) with Creative Commons licenses and mean duration of 8 min, are stored in MP4, MOV, M4V, and AVI formats [9]. The dataset has been segmented into 1,425,454 short video segments based on provided master shot boundary files. TRECVID utilizes this dataset for the Ad-hoc Video Search task, annually releasing query sets and corresponding ground truth for evaluating proposed video retrieval systems. Accompanying the V3C2 dataset are .tsv files containing video metadata, including video ID, video segment ID (shot ID), start/end frame ID, and start/end time for each segmented video segment.

We utilize the main task query sets provided by TRECVID for AVS2022 and AVS2023. The availability of ground truth with this dataset and query set enables an objective evaluation of our proposed method.

4.2 Metrics

The performance of our method on the V3C2 dataset, with its corresponding query set and ground truth, is evaluated using the infAP (inferred Average Precision) metric. This metric, employed by TRECVID to assess search systems in the Ad-hoc Video Search Task, is chosen for several reasons. TRECVID provides the sample_eval.pl software and ground truth files for assessing retrieval results using the mean xinfAP (mean extended infAP) metric. The adoption of this metric by other research groups in the Ad-hoc Video Search task, along with its recognition by TRECVID, enables a direct comparison of our method's feasibility with previous approaches.

4.3 Detail Implementation

Given a text query q as input, we utilize an image search engine to retrieve relevant images. The top $K = 30$ images from each search engine's results are selected as query images. This process is fully automated through software implementation. Recognizing that different image search engines may yield varying results for the same text query, we conducted experiments with three prominent image search engines: Google Images, Bing Images, and Yahoo Images.

For each model employed, we extract embedding feature vectors for the text query q, K query images synthesized from image search engine, and N = 1,425,454 keyframes representing N video segments within the V3C2 dataset. Subsequently, we calculate the values $text\_kf_i$ and $qrImg\_kf_i$ for i-th keyframe as described in Eq. 1 and Eq. 2. In the K-Means Clustering step to cluster the K query images, various values of $C = [2, 29]$ were explored to identify the configuration yielding the highest mean xinfAP.

To demonstrate the efficacy of our proposed approach, we experimented with various strategies for utilizing the embedding feature vectors:

1. Using single model: CLIP or BEiT-3
   - Using only text query q with $text\_kf_i$ calculated follows Eq. 1 or only K query images with $qrImg\_kf_i$ calculated follows Eq. 2 to obtain the list of relevant scores of N video segments with text query q.
   - Using both text query q and K query images: we calculate the value $relScore\_query\_kf_i$ according to Eq. 5 for each keyframe kf_i, where $avg\_text\_kf_i$ in the case of using a single model is equal to the value $text\_kf_i$, and $avg\_qrImg\_kf_i$ is equal to the value $qrImg\_kf_i$. Experiment with $\varphi = [0.1 : 0.9 : 0.1]$.
2. Using both models: CLIP and BEiT-3
   - Using only text query q: we calculate $avg\_text\_kf_i$ for each kf_i follows Eq. 3 to obtain the list of relevant scores of N video segments with text query q. Experiment with $\alpha = [0.1 : 0.9 : 0.1]$.
   - Using only K query images: we calculate $avg\_qrImg_kf_i$ for each kf_i follows Eq. 4 to obtain the list of relevant scores of N video segments with text query q. Experiment with $\beta = [0.1 : 0.9 : 0.1]$.

– Using both text query q and K query images: we calculate the $relScore\_query\_kf_i$ for each kf_i follows Eq. 5 to obtain the list of relevant scores. Experiment with $\varphi = [0.1 : 0.9 : 0.1]$.

For each query, we return the top S = 1000 video segments with the highest relevance scores in the result list.

4.4 Results

Through experimentation with various values for the hyperparameters C, α, β, and φ, we selected the best results achieved with each method of utilizing embedding feature vectors, as shown in Tables 1, 2. The highest result we achieved on the TRECVID2022 AVS main task query set is 0.2356, shown in Table 1, and TRECVID2023 AVS main task query set is 0.2481, shown in Table 2.

The experimental results indicate that combining multiple general-purpose models generally outperforms using a single model for this task, except when relying solely on query images. Furthermore, leveraging both text queries and externally synthesized query images proves more effective than using either alone. This observation aligns with the findings of Waseda_Meisei_Softbank [12, 13], and WHU-NERCMS [3] in their respective experiments. Notably, the use of images synthesized from text queries via internet image search engines significantly enhances the video retrieval process, particularly when employing our proposed approach of utilizing a single keyframe per video segment.

However, when solely using query images, combining both models does not yield substantial improvements compared to using only the BEiT-3 model on the TRECVID2022 AVS main task query set, shown in Table 1, and even slightly underperforms BEiT-3 on the TRECVID2023 AVS main task query set, shown in Table 2. In the approach that integrates both the query q and K query images, employing CLIP and BEiT-3 for retrieval, the optimal weights consistently demonstrate a pattern: the model or individual approach that produces superior results is assigned a higher weight.

While our method does not achieve the highest results compared to other research groups on the TRECVID2022 and TRECVID2023 AVS main task query sets, it still outperforms several approaches, shown in Table 3. Notably, top-performing groups like WHU-NERCMS [3] and Waseda_Meisei_Softbank [12, 13] utilize numerous general-purpose visual-language models for feature extraction, employ multiple frames per video segment for comparison with the query, and incorporate generative models into their methods. In contrast, our approach leverages the search capabilities of image search engines, employs only two models for feature extraction, and utilizes a single keyframe per video segment for comparison with the text query. Our analysis of previous results shows that selecting one keyframe per shot is sufficient, reducing computational costs while still delivering results comparable to denser sampling methods. This streamlined approach results in a simpler, less computationally intensive, and more time-efficient method for each query.

Table 1: Evaluation results table with mean xinfAP metric for the method using only text query q, only web images, both text query q and web images for TRECVID2022 AVS main task query set . Using only text queries with CLIP+BEiT-3, α=0.3. Using only web images with CLIP+BEiT-3, β=0.1. Using both text query and web images: φ=0.8 with CLIP, BEiT-3; with BEiT-3+CLIP: α=0.3, β=0.0, φ=0.8 for Bing and Google Images, φ=0.7 for Yahoo Images.

		mean xinfAP		
		CLIP	BEiT-3	BEiT-3 + CLIP
Original query		0.0859	0.1818	*0.2054*
Only web images	Bing Images	0.0531	0.1640	*0.1649*
	Google Images	0.0578	0.1562	*0.1579*
	Yahoo Images	0.0506	*0.1637*	0.1632
Original query + Web images	Bing Images	0.1134	0.2114	*0.2265*
	Google Images	0.1123	0.2080	*0.2243*
	Yahoo Images	0.1148	0.2168	***0.2356***

Table 2: Evaluation results table with mean xinfAP metric for the method using only text query q, only web images, both text query q and web images for TRECVID2023 AVS main task query set . Using only text queries with CLIP+BEiT-3, α=0.3. Using only web images with CLIP+BEiT-3, β=0.1. Using both text query and web images: φ=0.8 with CLIP, φ=0.6 with BEiT-3; with BEiT-3+CLIP: α=0.3, β=0.0, φ=0.8 for Bing and Google Images, φ=0.9 for Yahoo Images.

		mean xinfAP		
		CLIP	BEiT-3	BEiT-3 + CLIP
Original query		0.0733	0.2112	*0.2401*
Only web images	Bing Images	0.0286	0.1102	*0.1093*
	Google Images	0.0366	0.1317	*0.1341*
	Yahoo Images	0.0242	*0.0909*	0.0901
Original query + Web images	Bing Images	0.0983	0.2296	***0.2481***
	Google Images	0.1015	0.2280	*0.2468*
	Yahoo Images	0.0974	0.2251	*0.2469*

Further analysis was conducted to evaluate the performance of our optimal method (combining CLIP and BEiT-3 for feature extraction, and integrating query images and text query for retrieval) using the mAP (mean Average Precision) and Average Recall metrics, shown in Table 4. Our proposed method consistently achieves mAP scores exceeding 0.44 across both query sets, with Avg. Recall surpassing 0.42 on the TRECVID2022 AVS main task query set and 0.37 on the TRECVID2023 AVS main task query set. Notably, the mean xinfAP score on the 2023 query set is considerably higher than that on the 2022 query set. While the difference in mAP is not substantial, the Avg. Recall score is significantly lower. This discrepancy can be attributed to the variation in the number of correct video segments present in the ground truth for each query between 2022 and 2023. Statistical analysis of the ground truth file reveals an average of 670.8 video segments per query in 2022, compared to 755.9 in 2023.

Table 3: Comparing our method's best results with other research groups, using TRECVID2022 and TRECVID2023 AVS main task query set.

	mean xinfAP	
Team	2022 Query set	2023 Query set
WHU_NERCMS [3]	-	**0.292**
WasedaMeiseiSoftbank [13]	**0.282**	0.285
RUCMM [6]	0.262	0.272
VIREO [15]	0.142	0.268
ITI_CERTH [2]	0.210	0.240
NII_UIT [11]	-	0.166
Ours	0.236	0.248

Table 4: Evaluate our best method (CLIP+BEiT-3 for feature extraction, query images+text query for retrieval) using mAP and Average Recall metrics. Utilize query images from 3 image search engines: Google Images, Bing Images, and Yahoo Images.

	2022 Query Set			**2023 Query Set**		
CLIP+BEiT3	mean xinfAP	mAP	Avg.Recall	mean xinfAP	mAP	Avg.Recall
Bing Images	0.2265	0.4411	0.4224	**0.2481**	0.4578	**0.3838**
Google Images	0.2243	0.4392	0.4235	0.2468	**0.4580**	0.3759
Yahoo Images	**0.2356**	**0.4498**	**0.4348**	0.2469	0.4522	0.3771

This difference in the number of ground truth video segments per query can influence the observed Recall values.

Considering the 2023 query dataset, specifically query ID 743 with the content: *"A man is talking in a small window located in the lower corner of the screen"*, when using image search engines to aggregate relevant images, the results largely do not satisfy the query's description, shown in Fig. 3. This could be explained by the complexity of the text description, leading to image search engines not fully grasping the meaning of the sentence and returning irrelevant results.

The resulting video segments (represented by the corresponding keyframe) returned for some queries by the best-performing approach (combining CLIP and BEiT-3 for feature extraction, and integrating query images and text query for retrieval) are shown in Fig. 4, 5. The vast majority of representative keyframes for the returned video segments strikingly illustrate description of the query and align with the provided ground truth by TRECVID.

We conducted additional experiments to evaluate the improvement potential when using more language-image models. We incorporated the pre-trained BLIP model (BLIP ViT-L Image-Text Retrieval COCO) into the BEiT-3 and CLIP models, and the results are shown in Table 5. It is evident that the query results were significantly improved, especially with the TRECVID2023 query set, where performance increased from 0.2481 (CLIP + BEiT-3) to 0.2708 (CLIP + BEiT-3 + BLIP). However, when considering Table 5, the evaluation results for

Table 5: The results obtained below are with the following weight settings: Only text query: CLIP=0.25, BLIP=0.25, BEiT-3=0.5; Only web images: CLIP=0.1, BLIP=0.3, BEiT-3=0.6; Both web images and text query: CLIP=0.1, BLIP=0.3, BEiT-3=0.6, web images weight=0.3 (for 2022 query set), CLIP=0.1, BLIP=0.4, BEiT-3=0.5, web images weight=0.1 (for 2023 query set).

		mean xinfAP	
		2022 Query	2023 Query
Only text query		0.2254	0.2656
Only web images	Bing Images	0.1703	0.1305
	Google Images	0.1674	0.1497
	Yahoo Images	0.1710	0.1189
Web images + text query	Bing Images	0.2385	**0.2708**
	Google Images	0.2389	0.2702
	Yahoo Images	**0.2436**	0.2676

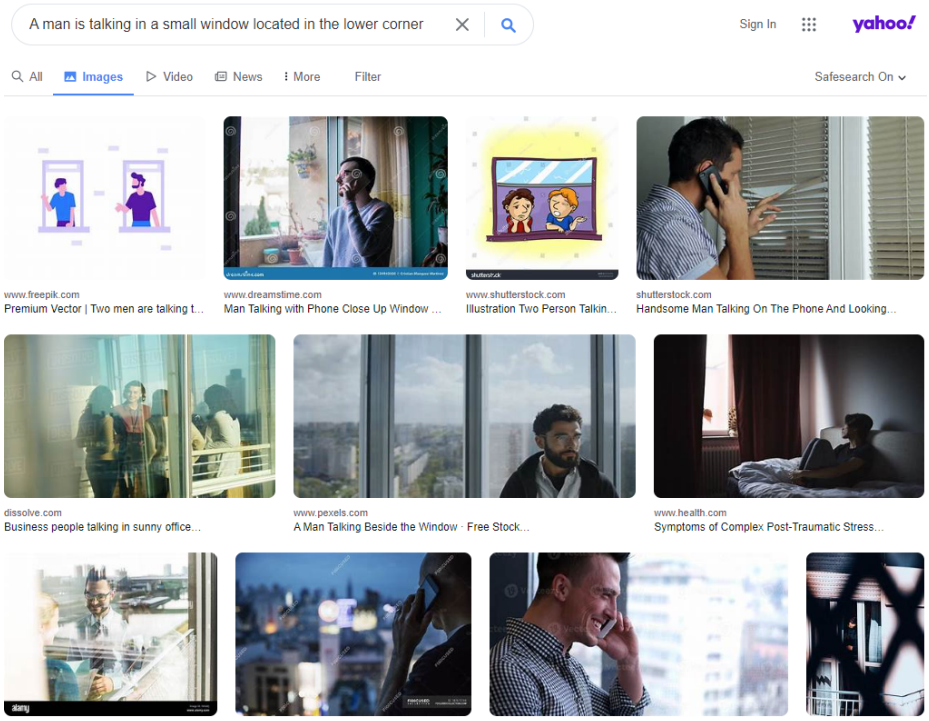

Fig. 3: The results returned by Yahoo Images for the search query *"A man is talking in a small window located in the lower corner of the screen"*.

combining web images and text queries compared to using only text queries did not show a substantial improvement.

When using these online search tools, concerns arise about the latency of commercial search engines and whether the trade-off of employing lightweight

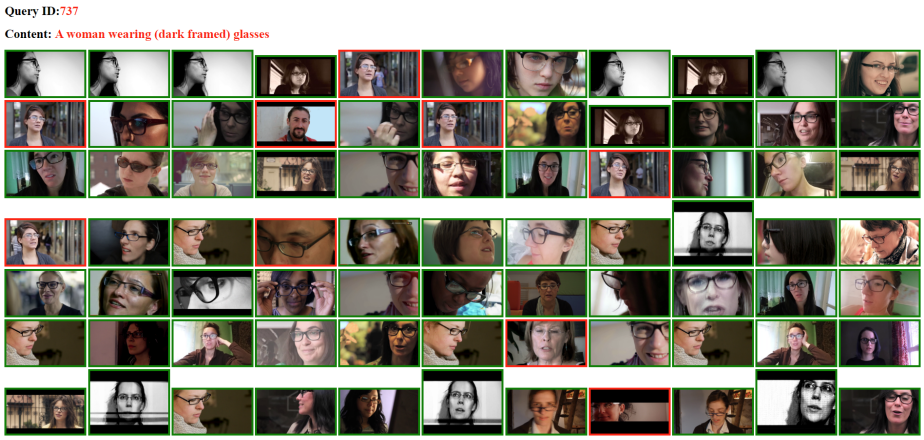

Fig. 4: Visualize query results with keyframes of each video segment returned for query ID 737 in the TRECVID2023 AVS query set. Query content is *"A woman wearing (dark framed) glasses"*. Keyframes with green borders represent video segments that match the ground truth, while those with red borders do not.

Fig. 5: Visualize query results with keyframes of each video segment returned for query ID 745 in the TRECVID2023 AVS query set. Query content is *"A person wearing gloves while biking"*. Keyframes with green borders represent video segments that match the ground truth, while those with red borders do not.

diffusion models is worthwhile. We believe that retrieving images from the Internet is faster than generating images of similar quality using a generative model. Furthermore, the effectiveness of using only images generated by Stable Diffusion, as shown in Table 1 of the WHU-NERCMS [3], with an infAP score of 0.0788 on the 2022 query set, is not good.

5 Conclusion

This paper presents a novel approach to Ad-hoc Video Search, demonstrating its feasibility through experimental implementation. Our method is computationally efficient, faster, and simpler than existing methods. By leveraging image search engines to aggregate relevant images, using a single keyframe per video segment, and combining two pre-trained general-purpose models for feature extraction, we achieve competitive results on the V3C2 dataset with TRECVID2023 and TRECVID2022 main task query sets for AVS. However, the reliance on image search engines can lead to poor results when descriptions are overly complex, and using only the middle frame as a keyframe may not always represent the entire video segment's content accurately.

Acknowledgement

This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under grant number C2024-26-06.

References

1. Fan, L., Krishnan, D., Isola, P., Katabi, D., Tian, Y.: Improving clip training with language rewrites. NeurIPS **36** (2024)
2. Galanopoulos, D., Mezaris, V.: Iti-certh participation in avs task of trecvid 2023. In: NIST (ed.) TRECVID 2023, International Workshop on Video Retrieval Evaluation (2023)
3. He, J., Li, R., Guo, J., Zhang, H., Li, M., Wu, Z., Wang, Z., Du, B., Liang, C.: Whunercms at trecvid 2023: Ad-hoc vedio search (avs) and deep video understanding (dvu) tasks. In: NIST (ed.) TRECVID 2023, International Workshop on Video Retrieval Evaluation (2023)
4. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: ICML. pp. 19730–19742. PMLR (2023)
5. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: ICML. pp. 12888–12900. PMLR (2022)
6. Li, X., Hu, F., Zhao, R., Wang, Z., Liu, J., Liu, J., Lan, B., Kou, W., Fu, Y., Kang, Z.: Renmin university of china and tencent at trecvid 2023: Harnessing pre-trained models for ad-hoc video search. In: NIST (ed.) TRECVID 2023, International Workshop on Video Retrieval Evaluation (2023)
7. Mu, N., Kirillov, A., Wagner, D., Xie, S.: Slip: Self-supervision meets language-image pre-training. In: ECCV. pp. 529–544. Springer (2022)
8. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763. PMLR (2021)
9. Rossetto, L., Schoeffmann, K., Bernstein, A.: Insights on the v3c2 dataset. arXiv preprint arXiv:2105.01475 (2021)

10. Rossetto, L., Schuldt, H., Awad, G., Butt, A.A.: V3c – a research video collection. In: MultiMedia Modeling. pp. 349–360 (2019)
11. Thuyen, T.D., Khiem, L., Thua, N., Tien, D., TruongAn, P., Tien-Dung, M., Duy-Dinh, L., Satoh, S.: Nii uit participation in avs and dvu tracks of trecvid 2023. In: NIST (ed.) TRECVID 2023, International Workshop on Video Retrieval Evaluation (2023)
12. Ueki, K., Suzuki, Y., Takushima, H., Okamoto, H., Tanoue, H., Hori, T.: Waseda meisei softbank at trecvid 2022. In: Proceedings of the TRECVid 2022 Workshop. pp. 1–5 (2022)
13. Ueki, K., Suzuki, Y., Takushima, H., Sato, H., Takada, T., Okamoto, H., Tanoue, H., Hori, T., Kumar, A.M.: Waseda meisei softbank at trecvid 2023. In: Proceedings of the TRECVid 2023 Workshop (2023)
14. Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., Aggarwal, K., Mohammed, O.K., Singhal, S., Som, S., Wei, F.: Image as a foreign language: Beit pretraining for vision and vision-language tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19175–19186 (June 2023)
15. Wu, J., Ma, Z., Zhong, S.H., Ngo, C.W.: Vireo @ trecvid 2023 ad-hoc video search. In: NIST (ed.) TRECVID 2023, International Workshop on Video Retrieval Evaluation (2023)