A W2VV++ Case Study with Automated and Interactive Text-to-Video Retrieval

Jakub Lokoč
Tomáš Souček
Patrik Veselý
František Mejzlík
SIRET research group, Department of Software Engineering, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic
lokoc@ksi.mff.cuni.cz

Jiaqi Ji
Chaoxi Xu
Xirong Li
AI and Media Computing Group, Key Lab of Data Engineering and Knowledge Engineering, School of Information, Renmin University of China, China
xirong@ruc.edu.cn

ABSTRACT

As reported by respected evaluation campaigns focusing both on automated and interactive video search approaches, deep learning started to dominate the video retrieval area. However, the results are still not satisfactory for many types of search tasks focusing on high recall. To report on this challenging problem, we present two orthogonal task-based performance studies centered around the state-of-the-art W2VV++ query representation learning model for video retrieval. First, an ablation study is presented to investigate which components of the model are effective in two types of benchmark tasks focusing on high recall. Second, interactive search scenarios from the Video Browser Showdown are analyzed for two winning prototype systems implementing a selected variant of the model and providing additional querying and visualization components. The analysis of collected logs demonstrates that even with the state-of-the-art text search video retrieval model, it is still auspicious to integrate users into the search process for task types, where high recall is essential.

CCS CONCEPTS

• **Information systems → Video search**.

KEYWORDS

datasets, neural networks, ad-hoc search, known-item search, representation learning

ACM Reference Format:
Jakub Lokoč, Tomáš Souček, Patrik Veselý, František Mejzlík, Jiaqi Ji, Chaoxi Xu, and Xirong Li. 2020. A W2VV++ Case Study with Automated and Interactive Text-to-Video Retrieval. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3414002

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3414002

1 INTRODUCTION

During the last decade, successful multimedia retrieval systems were established for search scenarios, where high precision of a searched class of images/videos is satisfactory for first few pages. However, many real world problems require also high recall, which is reflected by respected evaluation campaigns like TRECVID [4], Video Browser Showdown (VBS) [24], or Lifelog Search Challenge (LSC) [12]. More specifically, TRECVID focuses on Ad-hoc search (AVS) tasks to foster development of models for retrieval of all scenes matching a text description, while VBS and LSC focuse also on known-item search (KIS) where users are asked to find one specific scene in a given large dataset (e.g., V3C1 [33]), either based on visual memories or by a provided text description.

To aid with AVS and KIS tasks, various multimedia retrieval models are designed and tuned with available training datasets. Whereas traditional design processes involved "manual" feature modeling followed by model training, nowadays, end-to-end deep learning approaches [11] have become the mainstream. The approaches rely on deep architectures composed of building blocks that are jointly trained to learn both feature extraction and task-specific decisions. This paper investigates a state-of-the-art end-to-end deep learning approach W2VV++ [18] for text-to-video retrieval that won AVS task evaluations at TRECVID in 2018 [17] and second place in 2019 [19]. The approach tackles AVS tasks with cross-modality learning. Specifically, a model for free-form text search in unlabeled videos is trained using an architecture involving visual features from convolutional neural networks and language features from linguistic models with the goal to maximize the similarity between a short video and its text description. As the model involves many components, our first contribution is an ablation study providing new insights of the architecture performance and their comparison with a new W2VV++ variant based on RoBERTa-BASE [22] features.

Although state-of-the-art text-to-video retrieval models improved significantly search effectiveness, both AVS and KIS tasks are still far from being solved, as demonstrated in Section 3. The cross-modal similarity models, even though sufficiently trained on current data, do not necessarily generalize to handle novel queries. Meanwhile, users can face difficulties to provide descriptive/precise queries due to memory limitations, especially for known-item search tasks. For such cases, multimedia systems need to integrate also interactive search approaches [35, 37] that enable users to influence the search

process. Reported results from respected evaluation campaigns (e.g. VBS [24, 25]) support the hypothesis that a responsive interface enabling interactive query reformulation, iterative combinations of different modalities, informative visualizations and convenient browsing helps to reach the searched items in a limited time. Since current text-to-video retrieval models are tested mostly with automatic benchmark evaluations, many aspects of "real searching" are not considered. For example, users can overlook a correct image, or users can recognize that a query was not proper and reformulate it. Therefore, our second contribution is an interactive search study complementing presented automatic benchmark evaluations. Note that preparing an interactive search tool for a successful participation at VBS takes months for a whole team. The subsequent analysis and processing of logs is also non-trivial. This is probably a reason why this type of analysis is rare in the current literature. We believe that detailed joint presentation of inf AP like results and results from an interactive search campaign including a log analysis showing interesting statistics from real searches, represents a novel and unique contribution for the multimedia community.

In summary, the key contributions are twofold:

- To provide new insights to text-to-video retrieval with the W2VV++ model, in Section 3 we present a thorough ablation study comprising 18 trained/tested variants of the model and two different types of benchmarks. The study is further extended with a new W2VV++ variant (introduced in this paper) that outperforms the original variants in our known-item search benchmark. Consequently, we identify an easy to deploy competitive variant for interactive search.
- We have successfully participated at the most recent installment of VBS organized at MMM 2020 with two interactive search system prototypes incorporating a variant of the W2VV++ model. In Sections 4 and 5, we present results as well as a detailed log analysis demonstrating that the combination of query representation learning with other search features constitute a strong and effective state-of-the-art interactive known-item search approach.

2 RELATED WORK

2.1 Text-to-video retrieval

In order to search for unlabeled videos by free-form text, both the text modality and the video modality need to be represented in a shared common space for text-to-video similarity matching. Earlier efforts aim to use automatically detected concepts for cross-modal representation [27, 36]. In [27] for instance, concepts deemed to be relevant with respect to a given query are heuristically selected by word-to-word matching. By contrast, recent studies emphasize concept-free deep representation learning on both query and video sides [9, 18, 21, 29, 38]. For video representation, [9, 18, 38] adopt 2D ResNeXt / ResNet models pre-trained on ImageNet to extract visual CNN features from video frames, while multi-modal features regarding visual appearance, motion and audio are jointly exploited in [21]. For query representation learning, the state-of-the-art in light of the TRECVID AVS benchmarks [2, 3] relies on multi-scale text encoding, either by running multiple text encoders independently with their output concatenated later [18] or by applying the encoders in a stacked manner where the output of a previous encoder is used as the input of a following encoder [9, 38]. In this work, we opt for the open-sourced W2VV++ model [18], as it is computationally light, which is crucial for real-time user interaction. Moreover, compared to the stacked alternatives [9, 38], the parallel architecture of W2VV++ allows us to remove a specific encoder with ease, and thus reveal its importance for answering varied types of queries.

2.2 Interactive search systems

Engaging user decisions in the retrieval process can be realized with various interactive search strategies [35, 37]. However, a survey of possible systems is beyond the scope of this paper, and so we summarize basic features of just several selected recent systems that performed competitively at evaluation campaigns. Furthermore, two successful systems VIRET [26] and SOMHunter [14] are both described in more detail in Section 4.2.

The vitrivr system [34] is a long term champion of VBS and LSC competitions, where its stack of video retrieval models proved to be effective for interactive/iterative querying and result set browsing. Beside various sketch based search approaches, the system integrates many state-of-the-art image/video annotation models that produce meta-data for text search. A machine learning based approach was recently tested by the Exquisitor system [13] to localize searched items with positive and negative examples. The system interactively learns a linear SVM model during the search session, suggesting in each iteration the furthest multimedia objects from the actually trained hyper-plane. Interactive searching can be also based on visual browsing, where a candidate set of images can be organized into an exploration structure. Such strategy was integrated into HTW [6] and diveXplore [16] systems that used an image sorting approach to organize images to a 2D grid. On top of such grid, a hierarchical structure was constructed, enabling browsing with different levels of granularity. All the mentioned search strategies can benefit from the W2VV++ model that provides both a text-image mapping as well as a trained image representation space for visual search.

3 ABLATION STUDY ON W2VV++

While the effectiveness of the W2VV++ model for ad-hoc search is verified by its top performance in the TRECVID 2018 AVS task [18], we note that an ablation study on its components with respect to text and video embeddings is largely missing. As the inclusion of a specific component often means a substantial increase in computational resource, response time and memory footprint, such a study is important when integrating the model into an interactive system. In what follows, we provide a high-level description of the model in Section 3.1, followed by an ablation study in the context of ad-hoc search in Section 3.2 and known-item search in Section 3.3, respectively.

3.1 W2VV++ Overview

3.1.1 The original model. The W2VV++ model is developed for computing cross-modality similarity between a given textual query and an unlabeled video clip [18]. Conceptually, the model consists of a text embedding network and a video embedding network that

projects the query and the video into a learned common space, respectively. In particular, the text embedding network uses three text encoders in parallel, *i.e.,* bag-of-words (bow), word2vec (w2v) [28] and gated recurrent unit (gru) [7], that encodes the given query into three real-valued feature vectors. The three vectors are concatenated and projected into the common space via a fully connected (FC) layer. As for the video embedding network, it takes a video feature vector as input, which is transformed into the same common space by an FC layer. The cross-modal similarity is computed as the cosine similarity in the common space. Videos to be retrieved are sorted in descending order in light of their cosine similarities with respect to a given query. In order to construct the common space optimal for video retrieval, the model learns from many video-sentence pairs using an improved marginal ranking loss [10].

3.1.2 Proposed extension. In addition to the original W2VV++ model components [18], we extend the evaluations with a new W2VV++ version utilizing more powerful transformer-based query encoder RoBERTa-BASE [22] instead of the GRU unit. This version outperforms the original W2VV++ in known-item search tasks. We hypothesize that long rich known-item query sentences can benefit from more advanced encoding, while TRECVID AVS queries are rather simple phrases where it is not as important to understand the sentence structure.

The *bow-w2v-bert* model encodes text query in addition to Bag-of-Words and word2vec by 768-dimensional vector obtained by averaging RoBERTa representations of query tokens. We initialize RoBERTa model by publicly available weights obtained from language modeling task, fully-connected projection layers of W2VV++ are initialized randomly by Glorot initializer. The whole model including RoBERTa is trained by Adam optimizer with a learning rate of 10^{-5} and linear warm-up for the half of the first epoch. The model is trained by the same loss as the original W2VV++ and the batch size is set to 96 due to GPU memory constrains. Note it is essential to fine-tune RoBERTa weights in order to achieve superior performance over GRU based W2VV++ version. Further, we would like to emphasize that this new version was "discovered" recently and thus was not considered at all for interactive search evaluations at VBS 2020 presented in Section 4.

3.2 W2VV++ for Ad-hoc Search

3.2.1 Evaluation protocol. We conduct experiments on the TRECVID AVS benchmarks in the previous four years, *i.e.,* TV16 / TV17 / TV18 / TV19. Each year the benchmark organizers provide 30 test queries, each expressed exclusively by a sentence that contains 7.1 words on average. We follow the setup of [19], using MSR-VTT [39] and TGIF [20] for training, and the training set of the TV16 VTT task [5] for validation. The test set for TV16 / TV17 / TV18 is IACC.3 [5], which contains nearly 336k video clips, while TV19 is tested on V3C1 [33] having one million video clips in total. We use public video features[1], where each video is represented by a 2,048 resnext-101 feature and a resnet-152 feature of the same length. Their concatenation is referred to as resnext-resnet.

W2VV++ variants. Note that the original implementation of the W2VV++ model [18] uses all the three text encoders, *i.e.,* bow, w2v and gru, and resnext-resnet as the video feature. To reveal the influence of the individual components, we re-implement W2VV++ with varied choices of text encoders and video features. In total, this results in 18 variants plus one new tested RoBERTa based variant, see Table 1. For each variant, we train the corresponding model three times and average their performance scores, *i.e.,* inferred Average Precision (infAP), as the result of this variant.

3.2.2 Results. Table 1 shows the performance of all W2VV++ variants on the TRECVID AVS task. For the ease of comparison, we have sorted these models in ascending order in terms of their mean performance. We observe two patterns from Table 1. First, concerning the choice of video features, resnext-resnet outperforms resnext-101, followed by resnet. Second, using bow alone turns out to be better than the combined alternatives or the new RoBERTa based variant. Our explanation is that while the AVS queries are written in sentences, many of them appear to be keywords based. For such cases, the bow encoder is adequate.

Table 1: Performance of W2VV++ with varied setups in the TRECVID AVS tasks, sorted in ascending order by the mean performance. Performance metric: infAP.

Video feature	Text encoder	TV16	TV17	TV18	TV19	Mean
resnext-resnet	gru	0.124	0.165	0.082	0.075	0.112
resnet-152	w2v	0.122	0.163	0.089	0.101	0.119
resnet-152	gru	0.133	0.166	0.086	0.094	0.120
resnet-152	bow-w2v-gru	0.139	0.165	0.083	0.109	0.124
resnet-152	bow-gru	0.136	0.185	0.087	0.113	0.130
resnet-152	bow	0.125	0.193	0.091	0.115	0.131
resnext-101	w2v	0.123	0.181	**0.112**	0.115	0.133
resnet-152	bow-w2v	0.140	0.183	0.098	0.119	0.135
resnext-101	bow-w2v-gru	0.137	0.194	0.090	0.120	0.135
resnext-101	gru	0.142	0.195	0.096	0.121	0.138
resnext-resnet	w2v	0.131	0.190	**0.113**	0.126	0.140
resnext-101	bow-gru	0.144	0.194	0.101	0.133	0.143
resnext-101	bow-w2v	0.140	0.193	**0.112**	0.135	0.145
resnext-101	bow	0.148	0.200	0.109	0.140	0.149
resnext-resnet	bow-gru	**0.154**	0.214	0.101	0.136	0.151
resnext-resnet	bow-w2v	0.146	0.213	**0.111**	0.138	0.152
resnext-resnet	bow-w2v-gru	**0.155**	0.215	0.107	0.138	0.154
resnext-resnet	bow	**0.156**	0.218	**0.110**	**0.151**	**0.159**
resnext-resnet	bow-w2v-bert	0.146	**0.241**	0.102	0.112	0.150

3.3 W2VV++ for Known-item Search

3.3.1 Evaluation protocol. We now test all the W2VV++ variants trained in the previous experiment for the KIS task scenario. In that regard, we created a benchmark test set for a collection S of 20k selected video frames from the V3C1 collection [33]. The test set consists of 202 pairs $\langle q, o \rangle$, where $o \in S$ is a randomly selected frame and q is its free form text (query) description provided by a user. The text consists of one detailed sentence about the frame.

3.3.2 Results. The performance of the multiple models is summarized in Table 2. Similar to the AVS experiments, the combined resnext-resnet is again better then two individual features. However, unlike AVS where using bow alone performs the best, for KIS the joint use of bow and advanced text encoders now tops the performance table, see the last rows. Compared with the AVS queries, the KIS queries are more detailed, with an average number of 12.5

[1]Video features are available at https://github.com/li-xirong/avs

Table 2: Performance of W2VV++ with varied setups in the KIS task, measured in terms of Recall at k=1, 20, 40. Runs are sorted in ascending order by the mean performance.

Video feature	Text encoder	R1	R20	R40	Mean
resnet-152	w2v	0.043	0.248	0.345	0.212
resnet-152	bow	0.045	0.256	0.347	0.216
resnet-152	gru	0.038	0.282	0.356	0.225
resnext-resnet	gru	0.057	0.269	0.393	0.240
resnet-152	bow-w2v	0.053	0.292	0.389	0.245
resnet-152	bow-w2v-gru	0.058	0.299	0.394	0.250
resnext-101	w2v	0.059	0.310	0.394	0.254
resnet-152	bow-gru	0.059	0.317	0.388	0.255
resnext-101	bow-gru	0.079	0.304	0.403	0.262
resnext-101	bow	0.089	0.299	0.404	0.264
resnext-resnet	w2v	0.069	0.323	0.421	0.271
resnext-101	bow-w2v	0.079	0.327	0.418	0.275
resnext-101	bow-w2v-gru	0.087	0.330	0.431	0.283
resnext-101	gru	0.077	0.330	0.442	0.283
resnext-resnet	bow	0.092	0.350	0.428	0.290
resnext-resnet	bow-gru	0.071	0.351	0.449	0.290
resnext-resnet	bow-w2v	0.089	0.376	0.449	0.305
resnext-resnet	bow-w2v-gru	0.081	0.371	0.467	0.306
resnext-resnet	bow-w2v-bert	**0.096**	**0.394**	**0.515**	**0.335**

words per query sentence (for AVS it is 7.0). On average, the KIS queries contain three times as many adjectives and adpositions as well as two more nouns per query compared to the AVS queries as computed by `nltk` POS tagger. So for better handling complex queries, W2VV++ with combined text encoders is appropriate.

The above ablation study on W2VV++ shows that for both AVS and KIS, the gru text encoder can be removed. For scenarios, *e.g.,* interactive search, where portability (especially query encoding module) is preferred to the best accuracy, W2VV++ could be applied with the bow text encoder and the resnext-101 & resnet-152 video features extracted in the preprocessing phase.

4 INTERACTIVE SEARCH WITH W2VV++

Indeed, analyzing ranking effectiveness for a set of benchmark tasks is a standard way to present a text search model to the community. However, real search use-cases differ from the laboratory evaluation setting and so the observed performance could differ too. For example, it might happen that real users overlook the searched item in the result list due to too many items on a display page [26]. On the other hand, if first pages of the result list are not satisfactory, users can try to reformulate the query to localize searched items with a different set of keywords and still solve a task in a given time limit. Hence, we provide a task based analysis of search performance with two different interactive search prototypes incorporating the W2VV++ model for text search. This section details the W2VV++ variant employed by the prototype systems, a description of used prototypes, and the results achieved at VBS 2020.

4.1 Easy to deploy W2VV++ variant

When designing an interactive video search system based on various ranking models, aspects like effectiveness, efficiency, and software project maintainability play an important role. Therefore, before VBS 2020 we selected a variant of W2VV++ that is not the most effective, but enables easy implementation and integration to other platforms. Figure 1 presents the effects of our choices on a tested sequence of models using the known-item search 20K benchmark with query-image pairs $\langle q_i, o_i \rangle$. The graph presents the proportion of searched images o_i that appeared up to a given rank for corresponding benchmark text queries q_i (i.e., average recall at a given rank, having just one relevant image o_i for each query q_i). All models use the combined resnext-resnet features.

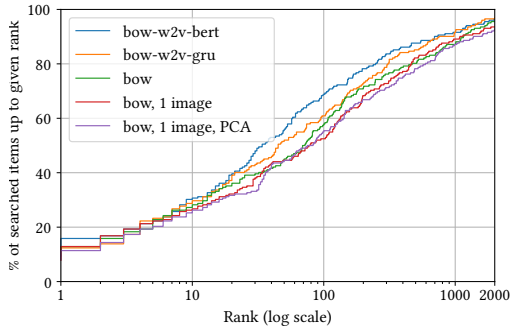

Figure 1: Sequence of W2VV++ model variants compared to the reference model *BoW-W2V-GRU*. For comparison, the new variant with RoBERTa-BASE is included as well.

The figure shows the effects of a sequence of changes to the reference model based on all *BoW-W2V-GRU* approaches. First, we considered just the *BoW* text embedding, which approximates the reference model well and at the same time allows easy implementation of text query embedding in both interactive search prototypes. To speed up feature extraction for large datasets (e.g., V3C1 [33] has 1000 hours of video), we used only one single image region for visual feature extraction, which still provides a reasonable effectiveness. Finally, a PCA transformation to a 128-dimensional space was applied on normalized vectors from the (learned) joint space. All these choices constitute a compromise variant (with respect to *BoW-W2V-GRU*) for the competition, where the "lightweight" variant *BoW, single-image, PCA* was successfully tested in both interactive prototypes. On the other hand, the new introduced variant *BoW-W2V-BERT* shows impressive recall improvement at ranks 20-300. So we plan to test this model in future for interactive search, where users are encouraged to write more complex textual queries.

4.2 Tested interactive search prototypes

Two different interactive search prototypes were tested for a set of KIS tasks at VBS 2020 – SOMHunter [14] and VIRET [26]. Both of them use the same set I of selected representative video frames (sorted for each video by time) and provide a text search mode for the frames using the selected W2VV++ variant with an option to formulate a temporal query [26]. For the similarity of two images, the prototypes used the same descriptor universe $\mathbb{R}^{128}$ designed for the W2VV++ model.

4.2.1 Ranking model for temporal text queries. Temporal text queries were found to be very important for the good performance of both tested prototypes. Hence, we remind details of the ranking

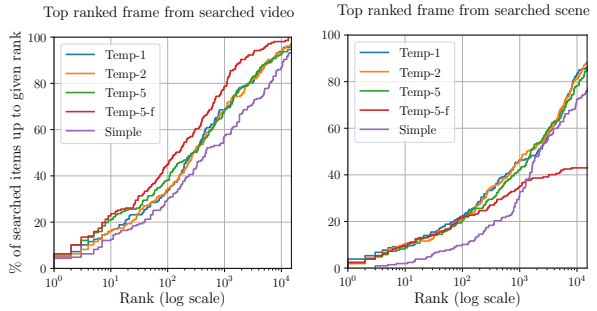

Figure 2: Ranking effectiveness for logged temporal query text pairs from VBS 2020 KIS sessions, $|I| \approx$ 1.15M frames.

model for the cosine similarity σ_{cos}. For each selected representative frame $f_i \in I$, we applied the visual embedding function of W2VV++ followed by PCA, *i.e.*, $t_v : I \rightarrow \mathbb{R}^{128}$, to represent the frame as a visual vector $v_i \in \mathbb{R}^{128}$. Similarly, let us denote $t_t : Q \rightarrow \mathbb{R}^{128}$ to be the query embedding function. Given a temporal query $\langle Q^{first}, Q^{second} \rangle$ describing a sequence of two images, each frame $f_i \in I$ receives two basic scores $\langle s_i^{first}, s_i^{second} \rangle$, where $s_i^j = 1 + \sigma_{cos}(t_v(f_i), t_t(Q^j))$. The overall score of a frame f_i with respect to a temporal query and for sorted I is then defined as:

$$score_{f_i} = s_i^{first} \cdot \max_{c=1\ldots x} s_{i+c}^{second},$$

where video boundary cases are handled separately[2]. Figure 2 shows the effectiveness of the temporal scoring with temporal queries collected during the competition[3]. For each temporal query, we considered its first part Q^{first} alone (denoted as *Simple*) and compared it to the temporal version (denoted as *Temp-x*) matching Q^{second} to consecutive frames of f_i. We test different sizes of the temporal context $x \in \{1, 2, 5\}$. The left graph focuses only on the top ranked frame from the (whole) searched video for a query during a VBS task. For each evaluated ranking model variant, a cumulative distribution of the top searched frame ranks for all queries is presented. For example, when searching up to the rank 100, users would encounter the first correct video frame in 40% cases when using the temporal context of size 5. In addition, a *Temp-5-f* variant is included (red line) with presentation filtering considering just top 3 frames from one video and top 1 frame from one shot in each result set (this setting was used at VBS 2020). We may observe that the filtering improves the chance to find a searched video frame when inspecting top ranked 100 frames, reaching almost 50% average recall. For the left graph, we would like to emphasize that the top ranked video frame does not have to visually match the particular searched scene from a VBS task. On the other hand, many videos in the V3C1 collection contain similar contents and so the top ranked video frame may represent a promising clue to solve a KIS task.

The right graph in Figure 2 focuses on the top ranked (selected) frame from a particular 20s long VBS task scene for a given query. As expected, the performance drops compared to the left graph as the queries match different parts of correct videos and only some of these matches are from the correct scene. Anyway, the average recall at the rank 100 for all temporal variants is twice higher compared to the tested simple query variant (please note that simple non-temporal text queries used at VBS are not included). On average, one in five temporal queries bring a searched scene frame to top ranked 100 frames (out of about 1.15M!). Assuming that users usually scan just top ranked frames (up to top 100-200), the variant with presentation filters is acceptable as it is still competitive for these ranks. The graph shows an expected behavior that for searched scene frames deeper in the ranked result set the presentation filters are usually too strong.

4.2.2 Tested Interactive Search Prototype 1 – SOMHunter. The system enables users to start with a simple or temporal text query and browse the ranked result list. From this list, users can select positive examples and update maintained relevance scores for each selected frame with a Bayesian relevance feedback approach [8]. A new form can be displayed with results sorted based on the new scores, or a self-organizing map can be used to select a new display with more diverse (yet relevant) frames. Users can open also two additional exploitation displays – video summary and top-K nearest frames for a selected frame in the result set. Both the relevance score approach and self organizing maps work with PCA-transformed vector features obtained by the W2VV++ model. The cosine distance δ_{cos} is used by the system. However, according to our post-VBS evaluations (for the same test set as is used in Figure 2), the distance based approach turned out to be slightly less effective for temporal text query scoring using $s_i^j = \delta_{cos}(x, y)$. SOMHunter for VBS 2020 had also a small issue in the implementation of the query embedding function t_t (in query point shifting right before PCA), nevertheless, having just a minor effect on the performance. For more details about SOMHunter, we refer to [14] or the project repository at https://github.com/siret/somhunter.

4.2.3 Tested Interactive Search Prototype 2 – VIRET. Our study involves also another interactive search system that was very successful at the Video Browser Showdown during the last three years. Specifically, part of the study is conducted on the basis of VIRET [26], but with its original text search model[4] replaced by the selected W2VV++ model variant. Unlike SOMHunter, VIRET uses the cosine similarity. In VIRET, users can enter multi-modal temporal queries in the left panel, while top ranked results are displayed in a grid panel on the right with an option to easily inspect the temporal (video) context of displayed frames. The query panel provides an option to enter a set of supported keywords[5]. In addition, users can draw a color sketch consisting of memorized color regions, two types of localized objects in the sketch, or select example images from the result set. For additional details on the sketch search models and multi-modal temporal fusion, we refer to [26].

4.3 VBS 2020 competition: Settings and results

Sharing the same large-scale dataset[6] and evaluating each task simultaneously in one room, Video Browser Showdown [24, 25, 32]

[2]For Figure 2, the last video frame is multiplied by 1 for temporal queries.
[3]We consider just logged text pairs, but the collected queries could be used in a different way at the competition (e.g., the second query could be set as primary in VIRET).
[4]VIRET used keyword search using class labels and (transformed) confidence scores assigned by a deep convolutional neural network.
[5]Matching keywords are "visually" prompted with their top matching images.
[6]Currently, the V3C1 dataset with 1000 hours of videos is used [33].

provides a reputable platform for comparative evaluations of various models and interactive search strategies. Each team participates with two tools/users cooperating on solving the currently evaluated task. At VBS 2020, visual and textual known-item search tasks were provided/presented by playing a short "known" video clip on a data projector (visual KIS) or by showing a gradually extended "known" scene description (textual KIS). For each task, there was a time limit[7] and penalty for wrong submissions. The competition was organized to expert visual/textual KIS sessions and novice visual KIS sessions. The novice users were not familiar with the systems as they were randomly selected from the audience and asked to solve several visual KIS tasks, after observing how expert users use the systems in the expert sessions. Furthermore, the novice users were "rotated" among the teams after three performed tasks.

Table 3 presents the number of solved tasks by participating scoring tools at VBS 2020. The numbers demonstrate that known-item search is still a challenging task for state-of-the-art interactive search systems. The presented W2VV++ model BoW variant was used by the first two teams that solved the highest number of known-item search tasks and achieved the first two places in the overall scoring. In the following section, we analyze result logs and demonstrate that typing one text query is mostly not sufficient to solve a challenging KIS task.

Table 3: Number of solved visual and textual KIS tasks during VBS 2020. Overall, SOMHunter and VIRET were ranked first and second at VBS 2020.

tool/team name	T-KIS	V-KIS	V-KIS N	$\sum$
SOMHunter [14]	8	5	2	15
VIRET [23]	8	4	2	14
vitrivr [34]	8	3	1	12
VIREO [30]	4	5	2	11
Exquisitor [13]	5	4	1	10
AAU [16]	7	2	0	9
IVIST [31]	5	3	1	9
ITEC	5	2	1	8
VERGE [1]	3	3	1	7
VNU [15]	1	2	0	3
number of tasks	10	6	6	22

5 RESULT LOG ANALYSIS

Both SOMHunter and VIRET implement logging of top results every time a ranked list is computed and presented to users[8]. Hence, given the knowledge of the searched scenes obtained after the competition, it is possible to reconstruct in part the search history of each task, revealing the current position of the searched scene video (or frame if available) in the logged result list. Specifically, logged results between a task start and end are considered, where the task ends for a team with a correct submission or with a time limit. These logs can reveal the effectiveness of text queries during the search process and whether additional implemented query formulation features in the prototypes were used/helpful during the competition. In the analysis oriented on the W2VV++ performance, we consider three possible types of result logs based on the utilized query:

- Simple text query (denoted as **t**) representing a sequence of words provided to the W2VV++ model. No other model is used, except presentation filters limiting the number of displayed top ranked frames from a shot and video.
- Temporal text query (denoted as **T**) extending the first type to a sequence of simple text queries, each targeting a different (consecutive) shot in the searched video scene.
- Other query approaches (denoted as **O**) comprising additional models used by the prototypes, but mostly used in a combination with the first two types **t** and **T**. For example, a color sketch or example image search combined with text search in VIRET, or frame scores from a temporal text query, further updated with a relevance feedback model based on selected positive examples in SOMHunter.

5.1 Used query types

Figure 3 shows a summary of the types of collected result logs for each tool, task, and instance (each team competed with two tools). Symbol "!" and bold font mark the query type used right before a correct submission. The time of the submission is presented as well on the bottom line for each instance with light gray background. The values in the tables show that in most cases users of both tools did not consider just simple text queries, but also temporal queries and other supported querying/search models during the competition. Except one case (T8, SOMHunter), a simple text query alone is present only in situations, where the second member of the team solved the task earlier. Only in one case, the task was solved right after a simple text query. Otherwise, a temporal query and/or other search models were used before a correct submission. Please note that in several cases the task was (quickly) solved after one temporal text query. The tables reveal different numbers of issued queries (i.e., result logs) during each task, ranging from units to dozens of query reformulations. However, to solve a task maximally twelve query interactions were used. The large white blocks (3 x 3 cells) correspond to novice user sessions where only one novice user was selected for each team. The high number of interactions (more than thirty) in several novice sessions demonstrate that novice users can search differently with the interfaces than expert users. For example, the first novice user of the VIRET prototype relied often on the semantic/color sketch canvas. Let us note that the user was really close to solve the first task, the user needed just a few more seconds.

5.2 Query type transitions

Figure 4 presents transition diagrams between the three types of logs, showing statistics of transitions at the directed edges between the types. Specifically, each edge is labeled with three numbers for query transitions where the position of searched video was improved, worsened, and the overall number of transitions. The diagrams show also nodes for search start and correct submission, showing the numbers of observed transitions to/from query

[7] Five minutes for visual KIS tasks and eight minutes for textual KIS tasks.

[8] The log analysis does not consider switching between different display types for one ranked list, or two already evaluated and cached result lists. In addition, data cleaning was applied for pairs of the same result lists occurring in a short time in a row due to a technical issue in logging.

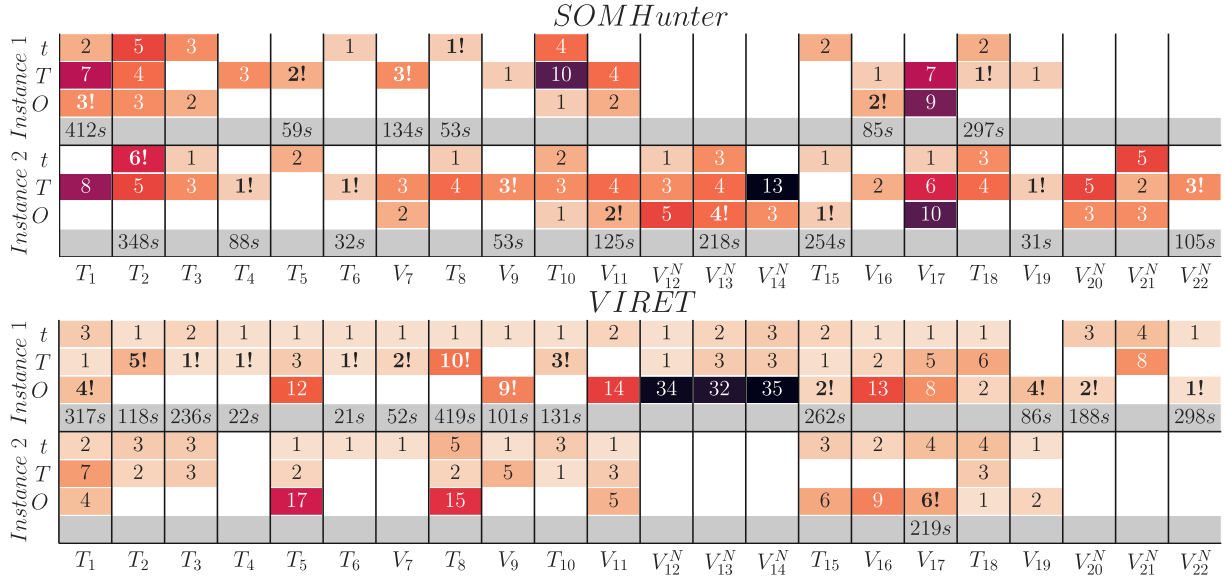

Figure 3: Statistics of used querying approaches logged by the two prototypes during VBS 2020 known-item search tasks. Exclamation mark indicates a query type used right before a correct submission.

types. The diagrams reveal that most of the searches[9] started with a W2VV++ text query, where SOMHunter supports also to start directly with a temporal text query, while VIRET supports speech meta-data (included in node O). The numbers in brackets for edges from the start node show the number of cases, where the searched video did not appear in the logged result set for the initial query. For most directed edges, we may observe that query reformulation had a more positive than negative effect on the position of a top ranked frame from the searched video (here we do not consider the searched scene). For both prototypes, the task was solved most often after (or using) a temporal text query.

5.3 Selected searches

In order to highlight interactive KIS challenges, we detail logs of several selected tasks from VBS 2020 in Figure 5. The searches are presented as diagrams, where x axis represents the time from the task start and y axis (log scale) shows the position of the searched video (solid line) or scene frame (dots). If the searched video ID or frame ID are not in the current result log, the line is not continuous and only a symbol is depicted in the top border line. Each diagram shows actions of both members of a team, distinguished with a color (orange and blue). The submission attempts are depicted as arrows; green for correct, red for incorrect submissions. The actions are depicted either as symbols {**t**, **T**} introduced in the beginning of this section, or with additional geometric shapes representing other supported search models by the prototypes:

- Rectangle shows that a relevance feedback model was used.
- Rhombus represents top-k nearest neighbors to an example.
- Circle denotes the usage of the color sketch search model.
- Triangle shows filtering with a localized object.

[9]The two initial queries to node O are actually also text searches in ASR data.

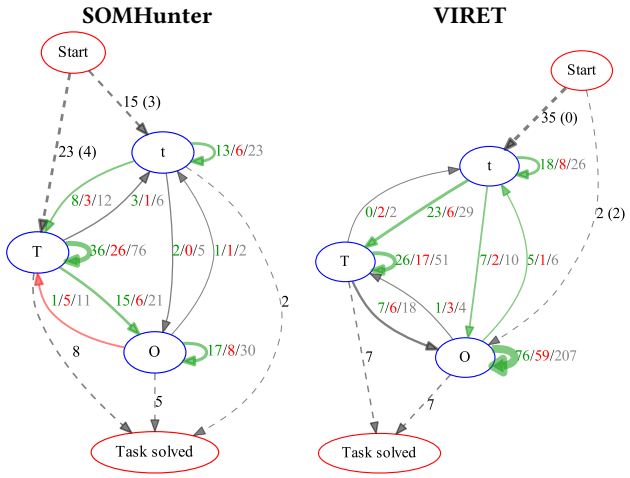

Figure 4: Transitions between query log types collected during VBS 2020 tasks. Edge labels between t, T, O denote how often a transition improved/worsened the logged position of a top frame from searched videos, while the overall number of transitions is presented with grey color. Result logs without searched video frames are included.

To highlight an actually updated model, the cyan color is used for member one, while the red color is used for member two. To further support statements about the necessity of interactive search, two panels below show also browsing actions logged by the systems (e.g., scrolling, or temporal context inspection). Please note that the panels do not show a heatmap based density. A line mark is painted at a given time position if there is a record in the corresponding browsing interaction log.

Figure 5: Selected search logs of SOMHunter and VIRET prototypes at VBS 2020. The x axis shows the time from the task start, with y axis (log scale) indicating the position of the searched video (solid line) or scene frame (dots).

The diagrams of real user searches in Figure 5 confirm that even with the state-of-the-art W2VV++ model one text query (even temporal) does not have to be effective enough to bring the searched shot to the first page of a result set. Hence, users often need to interactively reformulate the query in combination with frequent result set browsing. In several situations, the users were able to recognize the searched video from a non-relevant but visually similar video frame. The diagrams show examples, where the positions/ranks of the searched video/shot frames are not converging to better values with user actions. Furthermore, there are cases where users overlook the searched frame on the first page and then lose the frame from "sight" with another query reformulation.

6 CONCLUSIONS AND BEYOND

This paper presents an ablation study investigating various configurations of the W2VV++ model and interactive search evaluations with real users at the respected Video Browser Showdown competition. A significant part of the evaluations focus on known-item search that is still considered to be a very challenging search scenario. The evaluations reveal promising as well as easy to deploy model configurations that can be conveniently integrated to multimedia search engines. To conclude our work and analysis, the development of more effective text-image search models optimized with benchmark evaluations should be complemented with the design of more effective interactive search models. Especially for known-item search tasks with an inherent 100% recall requirement, effective visualization models for faster result set inspection and query building models providing a better convergence of the search process represent an important research direction to improve the overall effectiveness of interactive video search systems.

Based on our study and observations, we formulate two additional challenging questions for future inspection.

What is the true potential of advanced text encoders for text-to-video retrieval? According to our evaluation, the architecture with more advanced text encoders, *e.g.,* RoBERTa-BASE, starts to outperform the simple BoW encoder for queries comprising also adjectives and adpositions. Indeed, these words help to distinguish frequent objects in the database. However, it is necessary to conveniently learn mapping of sentences with these clarifying words to suitable cross-modal representations with no need of immense train data.

What is the optimal querying strategy for challenging search tasks? Using different visualizations and views of logged results, we show that one entered text query was not sufficient in most cases to solve a KIS task at VBS 2020. Users either entered a new text query or combined the query with another search mode. The presented statistics are in favor of query reformulation instead of long-lasting sequential browsing after a first query. Yet, it is still unclear whether other search modes pay off higher complexity of the user interface, especially for novice users. Therefore, more studies focusing on the comparison of various subsets of available querying models are in demand. Since these studies are highly time consuming, it is also desired to design simulation frameworks to automate these tests and provide reliable estimates of expected performance.

ACKNOWLEDGMENTS

This research has been supported by Czech Science Foundation (GAČR) project 19-22071Y, Charles University grant SVV-260451, National Natural Science Foundation of China (No. 61672523), Beijing Natural Science Foundation (No. 4202033), and the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (No. 18XNLG19). We would also like to thank Gregor Kovalčík for his help with VIRET and Miroslav Kratochvíl for his help with SOMHunter.

Corresponding authors: Xirong Li (for automated evaluation) and Jakub Lokoč (for interactive evaluation).

REFERENCES

[1] Stelios Andreadis, Anastasia Moumtzidou, Konstantinos Apostolidis, Konstantinos Gkountakos, Damianos Galanopoulos, Emmanouil Michail, Ilias Gialampoukidis, Stefanos Vrochidis, Vasileios Mezaris, and Ioannis Kompatsiaris. 2020.

VERGE in VBS 2020. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 778–783.

[2] George Awad, Asad Butt, Keith Curtis, Jonathan Fiscus, Afzal Godil, Alan F. Smeaton, Yvette Graham, Wessel Kraaij, Georges Quénot, Joao Magalhaes, David Semedo, and Saverio Blasi. 2018. TRECVID 2018: Benchmarking Video Activity Detection, Video Captioning and Matching, Video Storytelling Linking and Video Search. In *Proceedings of TRECVID 2018*. NIST, USA.

[3] George Awad, Asad Butt, Keith Curtis, Yooyoung Lee, Jonathan Fiscus, Afzal Godil, Andrew Delgado, Alan F. Smeaton, Yvette Graham, Wessel Kraaij, and Georges Quénot. 2019. TRECVID 2019: An evaluation campaign to benchmark Video Activity Detection, Video Captioning and Matching, and Video Search & retrieval. In *Proceedings of TRECVID 2019*. NIST, USA.

[4] George Awad, Asad Butt, Jonathan Fiscus, Martial Michel, David Joy, Wessel Kraaij, Alan F. Smeaton, Georges Quénot, Maria Eskevich, Roeland Ordelman, Gareth J. F. Jones, and Benoit Huet. 2017. TRECVID 2017: Evaluating Ad-hoc and Instance Video Search, Events Detection, Video Captioning and Hyperlinking. In *Proceedings of TRECVID 2017*. NIST, USA.

[5] G. Awad, J. Fiscus, D. Joy, M. Michel, A. Smeaton, W. Kraaij, G. Quénot, M. Eskevich, R. Aly, R. Ordelman, G. Jones, B. Huet, and M. Larson. 2016. TRECVID 2016: Evaluating Video Search, Video Event Detection, Localization, and Hyperlinking. In *TRECVID*.

[6] Kai Uwe Barthel, Nico Hezel, and Radek Mackowiak. 2015. ImageMap - Visually Browsing Millions of Images. In *MultiMedia Modeling - 21st International Conference, MMM 2015, Sydney, NSW, Australia, January 5-7, 2015, Proceedings, Part II*. 287–290. https://doi.org/10.1007/978-3-319-14442-9_30

[7] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *EMNLP*.

[8] Ingemar J Cox, Matthew L Miller, Thomas P Minka, Thomas V Papathomas, and Peter N Yianilos. 2000. The Bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments. *IEEE transactions on image processing* 9, 1 (2000), 20–37.

[9] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, Yuan He, Gang Yang, and Xun Wang. 2019. Dual Encoding for Zero-Example Video Retrieval. In *CVPR*.

[10] F. Faghri, D. J Fleet, J. R. Kiros, and S. Fidler. 2018. VSE++: Improved visual-semantic embeddings. In *BMVC*.

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

[12] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Andreas Leibetseder, Liting Zhou, Aaron Duane, Duc-Tien Dang-Nguyen, Michael Riegler, Luca Piras, Minh-Triet Tran, Jakub Lokoč, and Wolfgang Hürst. 2019. [Invited papers] Comparing Approaches to Interactive Lifelog Search at the Lifelog Search Challenge (LSC2018). *ITE Transactions on Media Technology and Applications* 7, 2 (2019), 46–59. https://doi.org/10.3169/mta.7.46

[13] Björn Pór Jónsson, Omar Shahbaz Khan, Dennis C. Koelma, Stevan Rudinac, Marcel Worring, and Jan Zahálka. 2020. Exquisitor at the Video Browser Showdown 2020. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 796–802.

[14] Miroslav Kratochvíl, Patrik Veselý, František Mejzlík, and Jakub Lokoč. 2020. SOM-Hunter: Video Browsing with Relevance-to-SOM Feedback Loop. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 790–795.

[15] Nguyen-Khang Le, Dieu-Hien Nguyen, and Minh-Triet Tran. 2020. An Interactive Video Search Platform for Multi-modal Retrieval with Advanced Concepts. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 766–771.

[16] Andreas Leibetseder, Bernd Münzer, Jürgen Primus, Sabrina Kletz, and Klaus Schoeffmann. 2020. diveXplore 4.0: The ITEC Deep Interactive Video Exploration System at VBS2020. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 753–759.

[17] Xirong Li, Jianfeng Dong, Chaoxi Xu, Jing Cao, Xun Wang, and Gang Yang. 2018. Renmin University of China and Zhejiang Gongshang University at TRECVID 2018: Deep Cross-Modal Embeddings for Video-Text Retrieval. In *TRECVID 2018 Workshop*.

[18] Xirong Li, Chaoxi Xu, Gang Yang, Zhineng Chen, and Jianfeng Dong. 2019. W2VV++: Fully Deep Learning for Ad-hoc Video Search. In *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019*. 1786–1794. https://doi.org/10.1145/3343031.3350906

[19] Xirong Li, Jinde Ye, Chaoxi Xu, Shanjinwen Yun, Leimin Zhang, Xun Wang, Rui Qian, and Jianfeng Dong. 2019. Renmin University of China and Zhejiang Gongshang University at TRECVID 2019: Learn to Search and Describe Videos. In *TRECVID*.

[20] Y. Li, Y. Song, L. Cao, J. Tetreault, L. Goldberg, A. Jaimes, and J. Luo. 2016. TGIF: A New Dataset and Benchmark on Animated GIF Description. In *CVPR*.

[21] Y. Liu, S. Albanie, A. Nagrani, and A. Zisserman. 2019. Use What You Have: Video Retrieval Using Representations From Collaborative Experts. In *BMVC*.

[22] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR* abs/1907.11692 (2019). arXiv:1907.11692 http://arxiv.org/abs/1907.11692

[23] Jakub Lokoč, Gregor Kovalčík, and Tomáš Souček. 2020. VIRET at Video Browser Showdown 2020. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 784–789.

[24] Jakub Lokoč, Werner Bailer, Klaus Schoeffmann, Bernd Münzer, and George Awad. 2018. On Influential Trends in Interactive Video Retrieval: Video Browser Showdown 2015-2017. *IEEE Trans. Multimedia* 20, 12 (2018), 3361–3376. https://doi.org/10.1109/TMM.2018.2830110

[25] Jakub Lokoč, Gregor Kovalčík, Bernd Münzer, Klaus Schöffmann, Werner Bailer, Ralph Gasser, Stefanos Vrochidis, Phuong Anh Nguyen, Sitapa Rujikietgumjorn, and Kai Uwe Barthel. 2019. Interactive Search or Sequential Browsing? A Detailed Analysis of the Video Browser Showdown 2018. *ACM Trans. Multimedia Comput. Commun. Appl.* 15, 1, Article 29 (Feb. 2019), 18 pages. https://doi.org/10.1145/3295663

[26] Jakub Lokoč, Gregor Kovalčík, Tomáš Souček, Jaroslav Moravec, and Přemysl Čech. 2019. A Framework for Effective Known-item Search in Video. In *Proceedings of the 27th ACM International Conference on Multimedia (MM '19)*. ACM, New York, NY, USA, 1777–1785. https://doi.org/10.1145/3343031.3351046

[27] Yi-Jie Lu, Hao Zhang, Maaike de Boer, and Chong-Wah Ngo. 2016. Event Detection with Zero Example: Select the Right and Suppress the Wrong Concepts. In *ICMR*.

[28] T. Mikolov, K. Chen, G. Corrado, and J. Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *ICLR*.

[29] N. Mithun, J. Li, F. Metze, and A. K. Roy-Chowdhury. 2019. Joint Embeddings With Multimodal Cues For Video-text Retrieval. *International Journal of Multimedia Information Retrieval* 8, 1 (2019), 3–18.

[30] Phuong Anh Nguyen, Jiaxin Wu, Chong-Wah Ngo, Danny Francis, and Benoit Huet. 2020. VIREO @ Video Browser Showdown 2020. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 772–777.

[31] Sungjune Park, Jaeyub Song, Minho Park, and Yong Man Ro. 2020. IVIST: Interactive VIdeo Search Tool in VBS 2020. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 809–814.

[32] L. Rossetto, R. Gasser, J. Lokoc, W. Bailer, K. Schoeffmann, B. Muenzer, T. Soucek, P. A. Nguyen, P. Bolettieri, A. Leibetseder, and S. Vrochidis. 2020. Interactive Video Retrieval in the Age of Deep Learning - Detailed Evaluation of VBS 2019. *IEEE Transactions on Multimedia* (2020).

[33] Luca Rossetto, Heiko Schuldt, George Awad, and Asad A. Butt. 2019. V3C - A Research Video Collection. In *MultiMedia Modeling - 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8-11, 2019, Proceedings, Part I*. 349–360. https://doi.org/10.1007/978-3-030-05710-7_29

[34] Loris Sauter, Mahnaz Amiri Parian, Ralph Gasser, Silvan Heller, Luca Rossetto, and Heiko Schuldt. 2020. Combining Boolean and Multimedia Retrieval in vitrivr for Large-Scale Video Search. In *MultiMedia Modeling*, Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve (Eds.). Springer International Publishing, Cham, 760–765.

[35] Klaus Schoeffmann, Marco A. Hudelist, and Jochen Huber. 2015. Video Interaction Tools: A Survey of Recent Work. *ACM Comput. Surv.* 48, 1, Article Article 14 (Sept. 2015), 34 pages. https://doi.org/10.1145/2808796

[36] Cees G. M. Snoek and Marcel Worring. 2009. Concept-Based Video Retrieval. *Found. Trends Inf. Retr.* 2, 4 (2009), 215–322.

[37] Bart Thomee and Michael S. Lew. 2012. Interactive search in image retrieval: a survey. *International Journal of Multimedia Information Retrieval* 1, 2 (01 Jul 2012), 71–86. https://doi.org/10.1007/s13735-012-0014-4

[38] X. Wu, D. Chen, Y. He, H. Xue, M. Song, and F. Mao. 2019. Hybrid Sequence Encoder For Text Based Video Retrieval. In *TRECVID*.

[39] J. Xu, T. Mei, T. Yao, and Y. Rui. 2016. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *CVPR*.