Proc. 26th Int. Conf. on Multimedia Modeling (MMM 2020), Springer LNCS vol. 11962. Author's accepted version.
The final publication is available at Springer via https://doi.org/10.1007/978-3-030-37734-2_69

VERGE IN VBS 2020

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

Information Technologies Institute/Centre for Research & Technology Hellas,
Thessaloniki, Greece
{andreadisst, moumtzid, kapost, gountakos, dgalanop, michem, heliasgj, stefanos, bmezaris, ikom}@iti.gr

Abstract. This paper demonstrates VERGE, an interactive video retrieval engine for browsing a collection of images or videos and searching for specific content. The engine integrates a multitude of retrieval methodologies that include visual and textual searches and further capabilities such as fusion and reranking. All search options and results appear in a web application that aims at a friendly user experience.

1 Introduction

VERGE is an interactive video retrieval system that provides users with efficient browsing and various search capabilities inside a set of video collections. For more than ten years, VERGE has been participating in numerous video retrieval related conferences and showcases, including TRECVID [1] and Video Browser Showdown (VBS) [2], thus the system is adapted to support the Known Item Search (KIS), Instance Search (INS) and Ad-Hoc Video Search (AVS) tasks. Experience from previous participation drove this year's selection of mature solutions (Section 2.1), the improvement of old modalities (Sections 2.2, 2.6), the integration of new (Sections 2.3, 2.4, 2.5) and also any advances regarding the user experience.

2 Video Retrieval System

VERGE serves as a video search engine with user-friendly browsing and a variety of modules to retrieve an image or a video from a collection. Furthermore, different search functionalities can be fused to create a combined query or they can be used consecutively to rerank the top results. A detailed description of the implemented indexing and retrieval modules follows in the next subsections, while the general architecture of VERGE can be seen in Figure 1. It should be noted here that all shot-based algorithms are based on the keyframes that derived from the provided V3C1 segmentation.

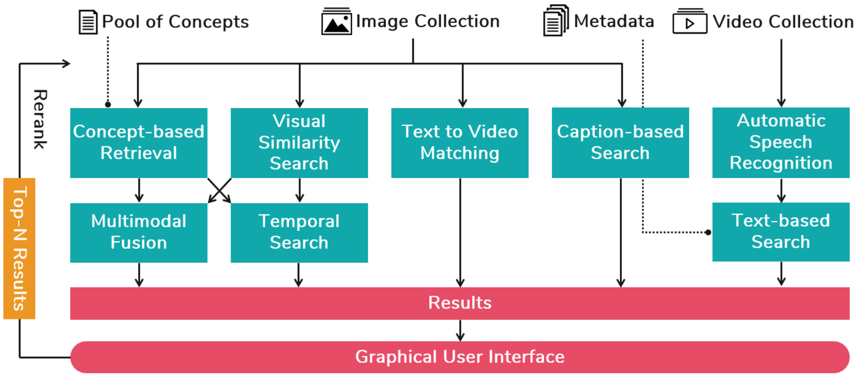

Fig. 1. VERGE System Architecture.

2.1 Visual Similarity Search

This module performs visual-based retrieval similarity of relevant content using convolutional neural networks (CNNs) upon a deep hashing architecture. A deep hashing approach will be followed in order to represent the visual information into a few bits (12, 24, 32, 48)[3]. Then, the retrieval framework will retrieve the relevant visual content by comparing the hamming distance of the generated binary vectors between the gallery images and the query. The backbone convolutional network will be an architecture similar to AlexNet or VGG16. Eventually, an IVFADC index database vector will be created for fast binary indexing and K-Nearest Neighbors will be computed for the query image [4].

2.2 Concept-Based Retrieval

This module annotates each keyframe with a pool of concepts, which comprises 1000 ImageNet concepts, 345 concepts of the TRECVID SIN task [5], 500 event-related concepts, 80 action-related, 365 scene classification concepts, 580 object labels and 30 style-related concepts. For performing the annotation, each keyframe was split to 9 equally-sized regions by applying a 3 × 3 grid, and each region, as well as the whole image, were processed separately so as to incorporate coarse localization information to the annotations. To obtain the annotation scores for the 1000 ImageNet concepts, we used an ensemble method, averaging the concept scores from four pre-trained models that employ different DCNN architectures, namely the VGG16, InceptionV3, InceptionResNetV2, as well as a hybrid model that combines the ImageNet and Places365 concept pools [6]. To obtain scores for the 345 TRECVID SIN concepts, we used the deep learning framework of [7]. For the event-related concepts we used the pre-trained model of EventNet [8] while for the action-related concepts we used a model trained on the AVA dataset [9]. Regarding the extraction of the scene-related concepts, we utilized the publicly available VGG16 model fine-tuned on the Places365 dataset.

Proc. 26th Int. Conf. on Multimedia Modeling (MMM 2020), Springer LNCS vol. 11962. Author's accepted version.
The final publication is available at Springer via https://doi.org/10.1007/978-3-030-37734-2_69

Object detection scores were extracted using models pre-trained on the established MS COCO and Open Images V4 datasets, with 80 and 500 detectable objects, respectively, and the bounding box information for each detected object was used for assigning the detection to one of the 9 considered keyframe regions. Finally, for the style-related concepts we employed the pre-trained models of [10].

2.3 Text to Video Matching module

This module compares a complex free-text query with a set of keyframes and returns a ranked list with the most correlated keyframes. Following the method proposed in [11], we use an architecture that learns to represent a textual instance (e.g. a sentence) and a visual instance (i.e. a keyframe) into a common feature space. Therefore, the correlation between a given text S_i and an image Im_j is directly comparable in the common space. For this, a dual encoding deep neural network that projects a natural language sentence and a shot keyframe into the common feature space is used. The network performs multi-level encoding in parallel, for both sentence and keyframes. A pre-trained Resnet-152 model is used for the initial keyframe representation, whereas each word sentence is initially encoded as a bag-of-words vector. Then, both the sentence and the keyframe representations go through three different encoders (i.e. mean-pooling, bi-GRU-based sequential model [12], and biGRU-CNN [13]). To train this module, we followed the approach of [14], and in terms of training data we combined two datasets, TGIF [15] and MSR-VTT [16]. The TGIF dataset contains approx. 100k short animated GIFs with one short description per each, while MSR-VTT consists of 10k short video clips, each accompanied by 20 short descriptions.

2.4 Automatic Speech Recognition

Acoustic content from videos is also exploited, by extracting audio channels and applying Automatic Speech Recognition (ASR) on them, in order to produce speech transcriptions for the whole collection. The basis for ASR is the open source framework CMU Sphinx-4 [17], a widely used, portable and flexible ASR system. The main components of the CMU Sphinx-4 Transcriber are a) a phonetic dictionary, which contains a mapping from words to phones, which are the basic units of speech, b) an acoustic model, which contains acoustic properties for each unit of speech, and c) a language model, which provides word-level language structure, by defining which word could follow previously recognized words and significantly restricting the matching process by stripping words that are not probable. Existing open source language and acoustic models are used in the context of VERGE platform. A priori extracted transcriptions and provided metadata are then fed into a text-based search module that uses Apache Lucene and enables the identification of a video by using words from the plot.

2.5 Video Captioning - Caption-Based Search

This module describes each video by a sentence/caption that is constructed from words included in a vocabulary, and thus the user can retrieve videos by simple text search. Video captioning approaches comprise two separate components: i) a feature extractor that typically extracts the features of a video by sampling among the frames using a fixed number as a step, and ii) an encoder-decoder that encodes the content and subsequently assigns it to words. To address this, an RNN-based neural network is used similar to [18]. The model is pre-trained on MSR-VTT [16], a widely-known dataset in video captioning domain. Finally, an approach based on [19] using reinforcement learning is implemented.

2.6 Multimodal Fusion and Temporal Search

This module fuses the results of two or more search modules, such as the visual descriptors (Section 2.1), the concepts (Section 2.2) and the color features mentioned in Section 3. Similar shots are retrieved by performing center-to-center comparisons among keyframes by using the selected modules. The query is described with multiple features (e.g. a shot, a color and/or concepts) and one of the features is considered by the user as dominant (i.e the most important one). The system returns the top-N relevant shots by considering solely the dominant feature (e.g. color), and then the other features are used for re-ranking the initial list by using a non-linear graph-based fusion method [20]. In order to perform temporal search, a query using multiple features of two adjacent shots is received, the top-N relevant images for one of the query shots are retrieved and finally this list is re-ranked by considering the features of the adjacent shot.

3 VERGE User Interface and Interaction Modes

The VERGE web application (Fig. 2) aims to provide end users with a friendly and effective way to utilise the developed retrieval algorithms, in a modern environment. Since this year we decided to incorporate a large number of modalities to offer more search options, our main goal is to serve them to the user in a non-complex way.

The VERGE user interface consists of three principal components: (i) a dashboard menu on the left, (ii) a results panel that covers most of the screen, and (iii) a filmstrip on the bottom. The menu contains a countdown timer that shows the remaining time to submit during VBS, a slider that adjusts the size of results, a back button that restores outcomes of previous queries and a switch button that defines whether a retrieval module will bring new matching shots or rerank existing results. Next, the various search modules are visualised as boxes that can be expanded or collapsed for reasons of compactness. In detail, *Concepts* and *Filters* present the entire list of visual concepts and filters respectively (Section 2.2), while both provide the option of auto-complete search. The selection of multiple concepts is also supported. *Colors* is a color palette in order to retrieve images of a specific shade. *Text Search* looks for the typed words in the video metadata, in the speech-to-text transcriptions (Section 2.4), and/or in the summaries described in Section 2.5, and it can also map the words to visual concepts and return most relevant shots (Section 2.3). Furthermore, *Combination* allows

Proc. 26th Int. Conf. on Multimedia Modeling (MMM 2020), Springer LNCS vol. 11962. Author's accepted version.
The final publication is available at Springer via https://doi.org/10.1007/978-3-030-37734-2_69

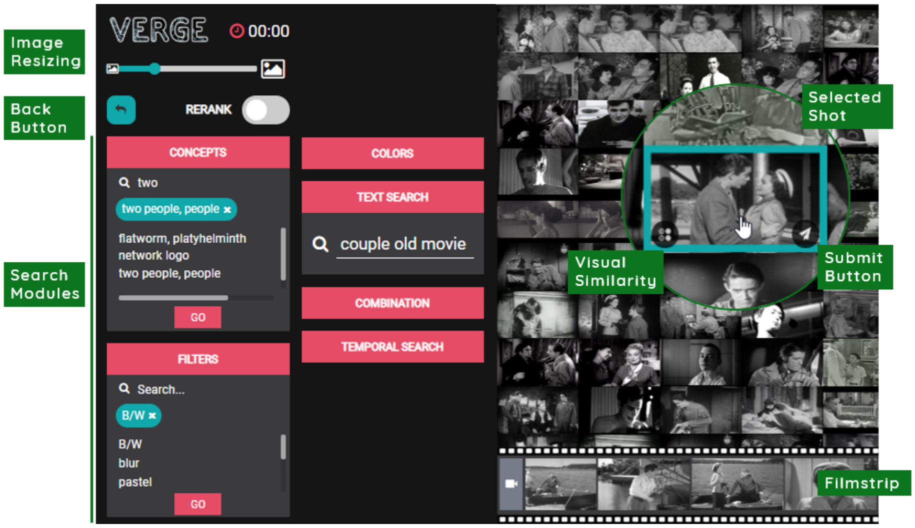

Fig. 2. Screenshot of the VERGE web application.

the fusion of some of the aforementioned modalities and *Temporal Search* offers the option to describe two consecutive shots (Section 2.6). The results of each search module are displayed in the main component, either as single images or groups of images (videos) in a grid view, sorted by highest relevance. Hovering over an image reveals the options to run the *Visual Similarity Search* (Section 2.1) or submit it to the contest. Clicking on the image shows the complete video in the bottom filmstrip, in the form of sequential shots.

To illustrate the capabilities of VERGE, a simple scenario is described, where users try to find shots of *a couple hugging in a black-and-white movie*. Search can be initiated by applying the “B/W” filter from the available list of filters and then combining it with the concept “two people”. Once a relevant image appears among the results, then visual similarity can be used in order to retrieve more similar shots. An alternative strategy is to look for relevant keywords (e.g., “couple old movie”) inside the metadata, the transcripts and the video summaries.

4 Future Work

Since some of the aforementioned retrieval modalities are introduced to VERGE for the first time, we will evaluate their performance during the VBS contest and we will decide accordingly on their further enhancement or modification.

Acknowledgements This work was supported by the EU’s Horizon 2020 research and innovation programme under grant agreements H2020-779962 V4Design, H2020-786731 CONNEXIONs and H2020-780656 ReTV.

References

1. G. Awad, A. Butt, et al. Trecvid 2018: Benchmarking video activity detection, video captioning and matching, video storytelling linking and video search. 2018.
2. J. Lokoč, G. Kovalčík, et al. Interactive search or sequential browsing? a detailed analysis of the video browser showdown 2018. *ACM TOMM*, 15(1):29, 2019.
3. H.-F. Yang, K. Lin, and C.-S. Chen. Supervised learning of semantics-preserving hash via deep convolutional neural networks. *IEEE Trans. on PAMI*, 40(2):437–451, 2017.
4. H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Trans. on PAMI*, 33(1):117–128, January 2011.
5. F. Markatopoulou, A. Moumtzidou, D. Galanopoulos, et al. ITI-CERTH participation in TRECVID 2017. In *Proc. TRECVID 2017 Workshop*, USA, 2017.
6. B. Zhou, A. Lapedriza, et al. Places: A 10 million image database for scene recognition. *IEEE Trans. on PAMI*, 40(6):1452–1464, 2017.
7. F. Markatopoulou, V. Mezaris, and I. Patras. Implicit and explicit concept relations in deep neural networks for multi-label video/image annotation. *IEEE Trans. on Circuits and Systems for Video Technology*, 2018.
8. Y. Guangnan, Yitong L., Hongliang X., et al. Eventnet: A large scale structured concept library for complex event detection in video. In *Proc. ACM MM*, 2015.
9. C. Gu, C. Sun, D. A Ross, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conf. on CVPR*, pages 6047–6056, 2018.
10. W. R Tan, C. S. Chan, H. E. Aguirre, and Kiyoshi Tanaka. Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification. In *2016 IEEE ICIP*, pages 3703–3707. IEEE, 2016.
11. J. Dong, X. Li, C. Xu, et al. Dual encoding for zero-example video retrieval. In *Proc. of the IEEE Conf. on CVPR*, pages 9346–9355, 2019.
12. K. Cho, Bart Van M., C. Gulcehre, et al. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
13. Y. Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.
14. F. Faghri, D. J Fleet, J. R Kiros, and S. Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. 2018.
15. Y. Li, Y. Song, L. Cao, et al. TGIF: A New Dataset and Benchmark on Animated GIF Description. In *The IEEE Conf. on CVPR*, June 2016.
16. J. Xu, T. Mei, T. Yao, and Y. Rui. MSR-VTT: A large video description dataset for bridging video and language. In *The IEEE Conf. on CVPR*, June 2016.
17. P. Lamere, P. Kwok, E. Gouvea, et al. The CMU SPHINX-4 speech recognition system. In *IEEE ICASSP 2003*, volume 1, pages 2–5, Hong Kong, 2003.
18. S. Venugopalan, M. Rohrbach, J. Donahue, et al. Sequence to sequence-video to text. In *Proceedings of the IEEE ICCV*, pages 4534–4542, 2015.
19. S. Phan, G. E. Henter, Y. Miyao, and S. Satoh. Consensus-based sequence training for video captioning. *arXiv preprint arXiv:1712.09532*, 2017.
20. I. Gialampoukidis, A. Moumtzidou, D. Liparas, et al. A hybrid graph-based and non-linear late fusion approach for multimedia retrieval. In *2016 14th International Workshop on CBMI*, pages 1–6, June 2016.