Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM'16), Miami, FL, USA, Springer LNCS vol. 9517, pp. 394–399, Jan. 2016.
Author's accepted version. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-27674-8_39

VERGE: A Multimodal Interactive Search Engine for Video Browsing and Retrieval

Anastasia Moumtzidou[1], Theodoros Mironidis[1], Evlampios Apostolidis[1], Foteini Markatopoulou[1,2], Anastasia Ioannidou[1], Ilias Gialampoukidis[1], Konstantinos Avgerinakis[1], Stefanos Vrochidis[1], Vasileios Mezaris[1], Ioannis Kompatsiaris[1], Ioannis Patras[2]

[1]Information Technologies Institute/Centre for Research and Technology Hellas,
6th Km. Charilaou - Thermi Road, 57001 Thermi-Thessaloniki, Greece
{moumtzid, mironidis, apostolid, markatopoulou, ioananas, heliasgj, koafgeri, stefanos, bmezaris, ikom}@iti.gr
[2]School of Electronic Engineering and Computer Science, QMUL, UK
i.patras@qmul.ac.uk

Abstract. This paper presents VERGE interactive search engine, which is capable of browsing and searching into video content. The system integrates content-based analysis and retrieval modules such as video shot segmentation, concept detection, clustering, as well as visual similarity and object-based search.

1 Introduction

This paper describes VERGE interactive video search engine[1], which is capable of browsing and retrieving video collections by integrating multimodal indexing and retrieval modules. VERGE supports Known Item Search task, which requires the incorporation of browsing, exploration and retrieval capabilities in a video collection.

Evaluation of earlier versions of VERGE search engine was performed with participation in video retrieval related conferences and showcases such as TRECVID, VideOlympics and Video Browser Showdown (VBS). Specifically, ITI-CERTH participated in several TRECVID tasks such as the Known Item Search (KIS) task, the Instance Search (INS) task, the Surveillance Event Detection (SED) task, in the VideOlympics event, and finally in VBS 2014 and VSS 2015. The proposed version of VERGE aims at participating to the KIS task of the Video Browser Showdown [1].

2 Video Retrieval System

VERGE[2] is an interactive retrieval system that combines advanced browsing and retrieval functionalities with a user-friendly interface, and supports the submission of

[1] More information and demos of VERGE are available at: http://mklab.iti.gr/verge/
[2] Latest VERGE system is available at: http://mklab-services.iti.gr/trec2015_v1/

Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM'16), Miami, FL, USA, Springer LNCS vol. 9517, pp. 394–399, Jan. 2016.
Author's accepted version. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-27674-8_39

queries and the accumulation of relevant retrieval results. The following indexing and retrieval modules are integrated in the developed search application: a) Visual Similarity Search Module; b) Object-based Visual Search, c) High Level Concept Detection; and d) Hierarchical Clustering.

The aforementioned modules allow the user to search through a collection of images and/or video keyframes. However, in the case of a video collection, it is essential that the videos are pre-processed in order to be indexed in smaller segments and semantic information should be extracted. The module that is applied for segmenting videos is shot segmentation.

Figure 1 depicts the general framework realized by VERGE in case of video collection, which contains all the aforementioned segmenting and indexing modules.

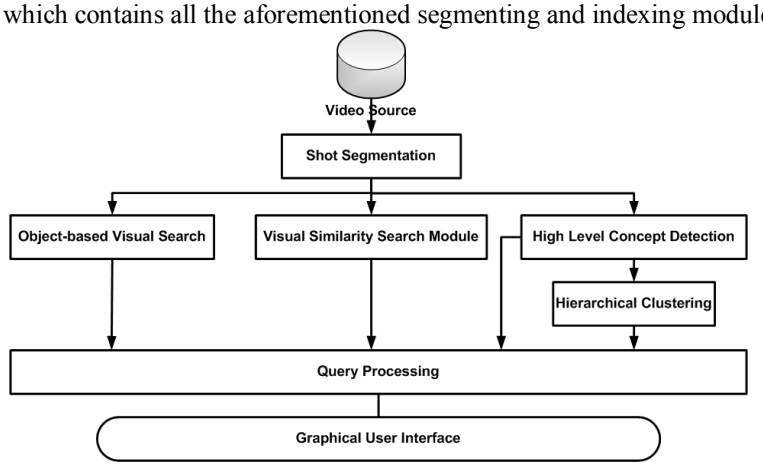

Fig. 1. Screenshot of VERGE video retrieval engine.

2.1 Video Temporal Segmentation

This module is applied for decomposing a video into elementary temporal segments by applying shot segmentation. The shots of the video, i.e., groups of consecutive frames captured without interruption from a single camera, are defined based on a variation of the algorithm described in [2]. According to the utilized method, the visual content of each video frame is represented by extracting an HSV histogram and a set of ORB (Oriented FAST and Rotated BRIEF) descriptors (introduced in [3]), allowing the algorithm to detect differences between a pair of frames, both in color distribution and at a more fine-grained structure level. An image matching strategy is then applied using the extracted descriptors, for assessing the visual similarity between successive or neighboring frames of the video. The computed similarity scores and their pattern over short sequences of frames are then compared against experimentally pre-specified thresholds and models that indicate the existence of abrupt and gradual shot transitions. The defined transitions are re-evaluated with the help of a flash detector which removes erroneously detected abrupt transitions due to camera flashes, and a pair of dissolve and wipe detectors (based on the methods from [4] and [5] respectively) that filter out wrongly identified gradual transitions due to camera

Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM'16), Miami, FL, USA, Springer LNCS vol. 9517, pp. 394–399, Jan. 2016.
Author's accepted version. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-27674-8_39

and/or object movement. Finally, the union of the resulting sets of detected abrupt and gradual transitions forms the output of the applied technique.

2.2 Visual Similarity Search

The visual similarity search module performs content-based retrieval based on local information. Specifically, simple SURF and color extension of SURF features are extracted. Then, we apply K-Means clustering on the database vectors in order to acquire the visual vocabulary and VLAD encoding for representing the images [6].

For Nearest Neighbour search, the best performing approach between the query and database vectors described in [6, 7] is applied. This approach involves the construction of an IVFADC index for database vectors and then the computation of K-Nearest Neighbours from the query file. Search is realized by combining an inverted file system with the Asymmetric Distance Computation (ADC). Finally, a web service is implemented in order to accelerate the querying process.

2.3 Object-based Visual Search

This module performs instance-based object retrieval and is based on the Bag-Of-Words (BoW) model. Initially, the fast Hessian detector and SIFT descriptor are applied for local feature extraction from the keyframes. Then, the detected features are randomly sampled and afterwards clustered using Repeated Bisecting K-Means [8] and a 2-layer visual vocabulary is constructed. Vocabularies of various sizes up to 150K are explored and the final representation of each frame is the result of a hard assignment. An inverted index is built using the open-source Apache Lucene software for fast online search of the image database BoW vectors. For querying the system, the whole keyframe (containing the object of interest) or any object/cropped part of the image can be selected. The similarity score between a query image and a video frame is obtained based on Lucene's scoring function (which exploits the tf-idf weighting scheme) and the ranking position of the frame in the retrieved list, i.e. Borda Counts is computed for all frames in the list in order to form the final ranking.

2.4 High Level Concept Retrieval Module

This module indexes the video shots based on 346 high level concepts (e.g. water, aircraft). To build concept detectors a two-layer concept detection system is employed [9]. The first layer builds multiple independent concept detectors. The video stream is initially sampled, generating for instance one keyframe per shot by shot segmentation. Subsequently, each sample is represented using one or more types of features based on three different pre-trained convolutional neural networks (CNN): i) The 16-layer deep ConvNet network [10], ii) the 22-layer GoogLeNet network [11] and iii) the 8-layer CaffeNet network [12]. Each of these networks is applied on the keyframes and the output of one or more layers is used as a feature. The CNN-based feature vectors are served as input to Support Vector Machine (SVN) classifiers. Specifically, one SVM is trained per concept and per feature type. The output of the classifiers trained

Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM'16), Miami, FL, USA, Springer LNCS vol. 9517, pp. 394–399, Jan. 2016.
Author's accepted version. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-27674-8_39

for the same concept is combined using a cascade of classifiers [13]. In the second layer of the stacking architecture, the fused scores from the first layer are aggregated in model vectors and refined by two different approaches. The first approach uses a multi-label learning algorithm that incorporates concept correlations [9]. The second approach is a temporal re-ranking method that re-evaluates the detection scores based on video segments as proposed in [14].

2.5 Hierarchical Clustering

This module clusters the video keyframes in a hierarchical agglomerative manner, so as to provide a structured hierarchical view of the video keyframes. We employ hierarchical agglomerative clustering [15], in which each leaf of the generated dendrogram represents a group of keyframes of similar content. We further elaborate the classic hierarchical agglomerative clustering method, in terms of speed and scalability, using skewed-split k-d trees [16]. In the clustering process, we also incorporate the responses of the concept detectors for each video shot, in order to increase the efficiency of the hierarchical representation.

3 VERGE Interface and Interaction Modes

The modules described in section 2 are incorporated into a friendly user interface (Figure 2) in order to aid the user to interact with the system. The existence of a friendly and smartly designed graphical interface (GUI) plays a vital role in the procedure. The interface features a fast and effective search functionality, partially thanks to the transition from MySQL to MongoDB. Comparing to the previous system, the technique of endless scrolling has been enabled for faster browsing. A RESTful API has also been developed in order to achieve asynchronous data calls. The main results area extends to the 90% of the screen and the video search toolbar is fixed to the top and covers about 10% of screen height in order to give the user the opportunity to have instant access to the core search modules. All the other components are collapsible. The interface comprises of three main components: a) the central component, b) the left side and c) the lower panel. We have incorporated the aforementioned modules inside these components. Below we describe briefly the three main components of the VERGE system and then present a simple usage scenario.

The central component of the interface includes a shot-based representation of the video in a grid-like interface. When the user hovers over a shot keyframe, three options appear on its corners (Figure 3). A selection tool that lets the user select the current keyframe, a cross tool that expands a view of the temporarily related shots, and a magnifier tool that opens up a frame that contains a larger preview of the image and gives the user the opportunity to crop an object from the image in order to make an object based search. All the shots from the main results area are also draggable. When a user starts to drag an image, the Visual Similarity Module area slides up and he can drop the image on this area in order to make a Visual Similarity Search with one or more images. On the left side of the interface resides the search history, as well

Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM'16), Miami, FL, USA, Springer LNCS vol. 9517, pp. 394–399, Jan. 2016.
Author's accepted version. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-27674-8_39

as additional search and browsing options (that include a high level visual concepts list and the hierarchical clustering controls). Lastly, the lower panel is a storage structure that holds the shots selected by the user.

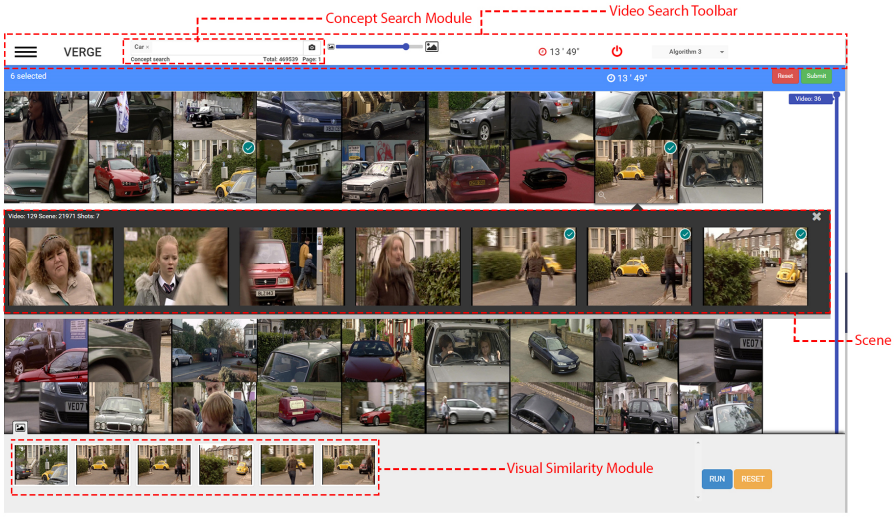

Fig. 2. Screenshot of VERGE video retrieval engine.

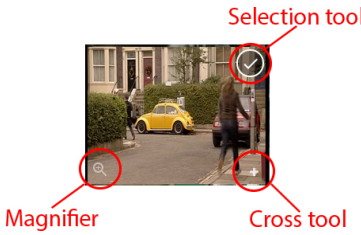

Fig. 3. Shot options.

Regarding the usage scenario for the KIS task, we suppose that a user is interested in finding a clip containing 'a yellow VW beetle with roofrack' (Figure 2). Given that there is a high level concept called "car", the user can initiate a search from it. Then, the visual similarity module can be used, if a relative image is retrieved during the first step or if an image that possibly matches the query is found; the temporally adjacent shots can be browsed and retrieved from the desired clip. Finally, the user can select the desirable shots, which will also be available in a basket structure.

4 Future Work

Future work includes the capability of querying the video collection with either one or more colors found in specific place of the shot or with a rough sketch of objects found in the query image. However, it should be noted that both require knowledge of the

Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM'16), Miami, FL, USA, Springer LNCS vol. 9517, pp. 394–399, Jan. 2016.
Author's accepted version. The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-27674-8_39

query image, i.e. the location of the color(s) in the image for the first case and the objects found inside an image in the second case.

Acknowledgements This work was supported by the European Commission under contracts FP7-600826 ForgetIT, FP7-610411 MULTISENSOR and FP7-312388 HOMER.

References

1. Schoeffmann, K.: A User-Centric Media Retrieval Competition: The Video Browser Showdown 2012-2014. IEEE Multimedia, vol. 21, no. 4, pp. 8-13 (2014)
2. Apostolidis, E., Mezaris, V.: Fast shot segmentation combining global and local visual descriptors. 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6583-6587 (2014)
3. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An efficient alternative to SIFT or SURF. 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2564-2571 (2011)
4. Su, C.-W., Liao, H.-Y.M., Tyan, H.-R., Fan, K.-C., Chen, L.-H.: A motion-tolerant dissolve detection algorithm. IEEE Transactions on Multimedia, vol. 7, pp.1106-1113 (2005)
5. Seo, K.-D., Park, S., Jung, S.-H.: Wipe scene-change detector based on visual rhythm spectrum. IEEE Transactions on Consumer Electronics, vol. 55, no. 2, pp. 831-838 (2009)
6. Jegou, H., Douze, M., Schmid, C., Perez. P.: Aggregating local descriptors into a compact image representation. In Proc. CVPR (2010)
7. Jegou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 117-128 (2011)
8. Steinbach, M., Karypis, G., and Kumar, V.: A comparison of document clustering techniques. In KDD Workshop on Text Mining (2000)
9. Markatopoulou, F., Mezaris, V., Pittaras, N., Patras, I.: Local Features and a Two-Layer Stacking Architecture for Semantic Concept Detection in Video. IEEE Transactions in Emerging Topics in Computing, vol.3, no.2, pp.193-204 (2015)
10. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv technical report (2014)
11. Szegedy, C., et al.: Going deeper with convolutions. In: CVPR 2015 (2015), http://arxiv.org/abs/1409.4842
12. Krizhevsky, A., Ilya, S., Hinton, G.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012)
13. Markatopoulou, F., Mezaris, V., Patras, I.: Cascade of classifiers based on binary, non-binary and deep convolutional network descriptors for video concept detection. In: IEEE Int. Conf. on Image Processing (ICIP 2015). IEEE, Canada (2015)
14. Safadi B., Quénot, G.: Re-ranking by local re-scoring for video indexing and retrieval. 20th ACM Int. Conf. on Information and Knowledge Management, pp. 2081–2084 (2011)
15. Murtagh, F., & Legendre, P.: Ward's hierarchical agglomerative clustering method: Which algorithms implement ward's criterion?. Journal of Classification, vol. 31(3), pp. 274-295 (2014)
16. Gialampoukidis, I., Vrochidis, S., and Kompatsiaris, I.: Fast Visual Vocabulary Construction for Image Retrieval using Skewed-Split k-d trees. Proc. 22nd Int. Conf. on MultiMedia Modeling (MMM16), Miami, USA (2016)