# VERGE: A Multimodal Interactive Video Search Engine

Anastasia Moumtzidou[1], Konstantinos Avgerinakis[1], Evlampios Apostolidis[1], Fotini Markatopoulou[1,2], Konstantinos Apostolidis[1], Theodoros Mironidis[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, koafgeri, apostolid, markatopoulou, kapost, mironidis, stefanos, bmezaris, ikom}@iti.gr
[2]School of Electronic Engineering and Computer Science, QMUL, UK
i.patras@eecs.qmul.ac.uk

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

## 1 Introduction

This paper describes VERGE interactive video search engine[1], which is capable of retrieving and browsing video collections by integrating multimodal indexing and retrieval modules. VERGE supports Known Item Search task, which requires the incorporation of browsing, exploration, or navigation capabilities in 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). The proposed version of VERGE aims at participating to the KIS task of the Video Search Showcase (VSS) Competition 2015 which was formerly known as Video Browser Showdown [1].

## 2 Video Retrieval System

VERGE is an interactive retrieval system that combines advanced retrieval functionalities with a user-friendly interface, and supports the submission of 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) High Level Concept Detection; and c) 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 se-

[1] More information and demos of VERGE are available at: http://mklab.iti.gr/verge/

mantic information should be extracted. The modules that are applied for segmenting videos are: a) Shot Segmentation; and b) Scene Segmentation;

Thus, the general framework realized by VERGE in case of video collection is depicted in Figure 1.

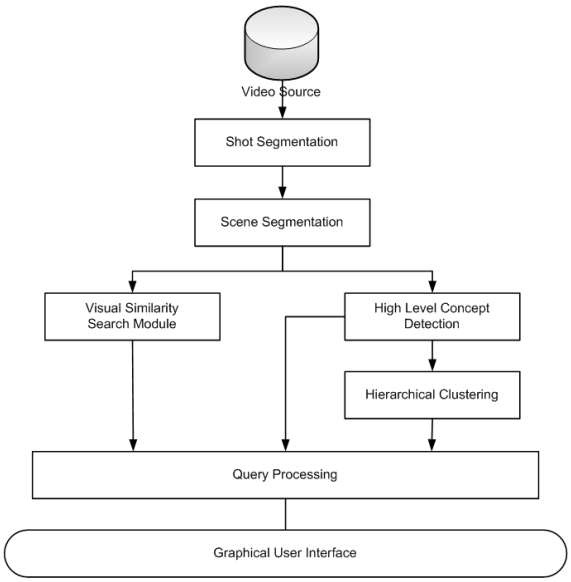


**Fig. 1.** Framework of VERGE

### 2.1 Shot segmentation

The video temporal decomposition module defines the shot segments of the video, i.e., video fragments composed by consecutive frames captured uninterruptedly from a single camera, based on a variation of the algorithm proposed in [2]. The utilized technique represents the visual content of the each frame by extracting an HSV histogram and a set of ORB (Oriented FAST and Rotated BRIEF) descriptors (introduced in [3]), being able to detect the differences between a pair of frames, both in color distribution and at a more fine-grained structure level. Then both abrupt and gradual transitions are detected by quantifying the change in the content of successive or neighboring frames of the video, and comparing it against experimentally specified thresholds that indicate the existence of abrupt and gradual shot transitions. Erroneously detected abrupt transitions are removed by applying a flash detector, while false alarms are filtered out after re-evaluating the defined gradual transitions with the help of a dissolve and a wipe detector that rely on the algorithms introduced in [4] and [5] respectively. Finally, a simple fusion approach (i.e. taking the union of the detected abrupt and gradual transitions) is used for forming the output of the algorithm.

### 2.2 Scene segmentation

Drawing input from the analysis in section 2.1, the scene segmentation algorithm of [6] defines the story-telling parts of the video, i.e., temporal segments covering either a single event or several related events taking place in parallel, by grouping shots into sets that correspond to individual scenes of the video. For this, content similarity (i.e., visual similarity assessed by comparing HSV histograms extracted from the keyframes of each shot) and temporal consistency among shots are jointly considered during the grouping of the shots into scenes, with the help of two extensions of the well-known Scene Transition Graph (STG) algorithm [7]. The first one reduces the computational cost of STG-based shot grouping by considering shot linking transitivity and the fact that scenes are by definition convex sets of shots, while the second one builds on the former to construct a probabilistic framework that alleviates the need for manual STG parameter selection. Based on these extensions, and as reported in [6], the employed technique can identify the scene-level structure of videos belonging to different genres, and provide results that match well the human expectations, while the needed time for processing is a very small fraction (<3%) of the video's duration.

### 2.3 Visual Similarity Search

The visual similarity search module performs content-based retrieval based on global and local information. To deal with global information, MPEG-7 descriptors are extracted from each keyframe and they are concatenated into a single feature vector. More specifically, the colour related descriptors Colour Structure (CS), Colour Layout (CL) and Scalable Colour (SC) are used. Regarding the case of local information, SURF features are extracted. Then K-Means clustering is applied on the database vectors in order to acquire the visual vocabulary and VLAD encoding for representing images is realized [8].

For Nearest Neighbour search we implement three different approaches between the query and database vectors that is described in [8]. In each case, an index is first constructed for database vectors and K-Nearest Neighbours are then computed from the query file. An index of lower-dimensional PCA-projected VLAD vectors, an ADC index and an IVFADC index were constructed from the database vectors as in [8]. Exhaustive search is deployed in the first two cases using a Symmetric Distance Computation (SDC) and Asymmetric Distance Computation (ADC) for Nearest Neighbour calculation, while a faster solution is suggested in the third one, where an inverted file system is combined with ADC instead. Based on the experiments realized in [9], the approach that performs the best is IVFADC. Therefore, in our implementation, we will apply this approach but we are going to investigate the other two methods as well. It should be noted that this indexing structure is utilized for both descriptors (i.e. global and local). Finally, a web service is implemented in order to accelerate the querying process.

### 2.4 High Level Concepts 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. 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 appropriate local descriptors (e.g. SIFT, RGB-SIFT, SURF, ORB etc.). The descriptors are extracted in more than one square regions at different scale levels. All the local descriptors are compacted using PCA and are subsequently aggregated using the VLAD encoding. These VLAD vectors are compressed by applying a modification of the random projection matrix [10] and served as input to Logistic Regression (LR) classifiers. Following the bagging methodology of [11] five LR classifiers are trained per concept and per local descriptor (SIFT, RGB-SIFT, SURF, ORB etc.), and their output is combined by means of late fusion (averaging). When different descriptors are combined, again late fusion is performed by averaging of the classifier output scores. 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 [12]. The second approach is a temporal re-ranking method that re-evaluates the detection scores based on video segments as proposed in [13].

### 2.5 Hierarchical Clustering

This module incorporates a generalized agglomerative hierarchical clustering process [14], which provides a structured hierarchical view of the video keyframes. In addition to the feature vectors described in section 2.3, we extract vectors consisting of the responses of the concept detectors for each video shot. The hierarchical clustering is applied to these representations to cluster the keyframes into classes, each of which consists of keyframes of similar content, in line with the concepts provided.

## 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, discover and retrieve the desired video clip. The existence of a friendly and smartly designed graphical interface (GUI) plays a vital role in the procedure. Within this context, the GUI of VERGE has been redesigned in order to improve the user experience. The new interface, similarly to older one, comprises of three main components: a) the central component, b) the left side, c) the lower panel. We have incorporated the aforementioned modules inside these components, in order to allow the user interact with the system and retrieve the desired video clip during known item search tasks. In the sequel, we describe briefly the three main components of the VERGE system and then present a simple usage scenario.

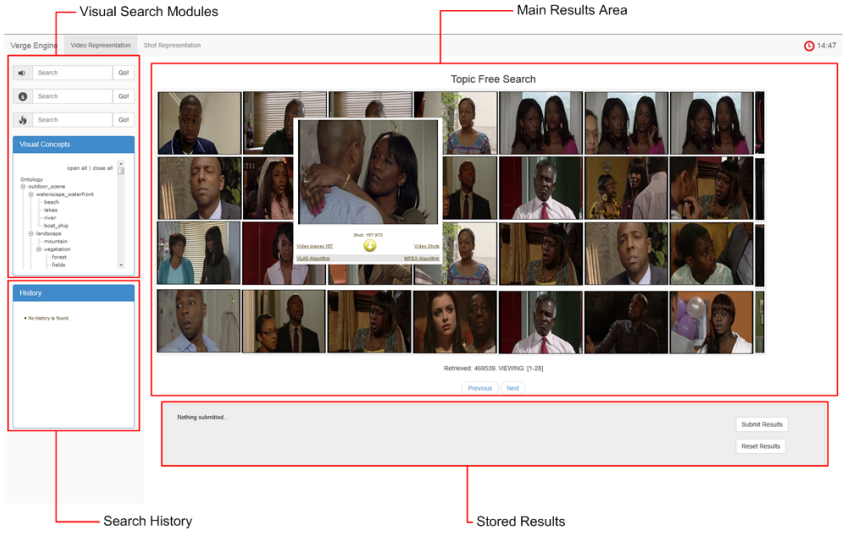


**Fig. 2.** Screenshot of VERGE video retrieval engine

The central component of the interface includes a shot-based or scene-based representation of the video in a grid-like interface. When the user hovers over a shot keyframe, the shot preview is visible by rolling three to five different keyframes that constitute the shot. Moreover, when the user clicks on a shot a pop-up frame appears that contains a larger preview of the image and several links that support her in viewing adjacent shots or all video shots, the frames constituting the shot and in searching for visually similar images. On the left side of the interface, the search history, as well as additional search and browsing options (that include the high level visual concepts and the hierarchical clustering) are displayed. Finally, the lower panel is a storage structure that holds the shots selected by the user.

Regarding the usage scenario for the known-item task, we suppose that a user is interested in finding a clip where 'a man hugs a woman while both of them have dark-skin' (Figure 2). Given that there is a high level concept called "dark-skinned people", the user can initiate her search from it. Then, she can either use the visual similarity module if a relative image is retrieved during the first step or if she finds an image that possibly matches the query; she can browse the temporally adjacent shots and retrieve the desired clip. Finally the user can store the desirable shots in a basket structure.

## 4 Future Work

Future work includes fusion of high level visual concepts in order to allow for retrieval of video shots that can be described equally with more than one concept. Another feature that could be implemented is the capability of querying the video collection

with one or more colors found in specific place of the shot. However, this requires knowledge of the specific location of the color in the image.

**Acknowledgements** This work was supported by the European Commission under contracts FP7-287911 LinkedTV, FP7-600826 ForgetIT, FP7-610411 MULTISENSOR and FP7-312388 HOMER.

## References

1. Schoeffmann, K., Bailer, W.: Video Browser Showdown. ACM SIGMultimedia Records, vol. 4, no. 2, pp. 1-2 (2012)
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. Sidiropoulos, P., Mezaris, V., Kompatsiaris, I., Meinedo, H., Bugalho, M., Trancoso, I.: Temporal video segmentation to scenes using high-level audiovisual features. IEEE Transactions on Circuits and Systems for Video Technology, vol. 21(8), pp. 1163-1177 (2011)
7. Yeung, M., Yeo, B.-L., Liu, B.: Segmentation of video by clustering and graph analysis. Computer Vision and Image Understanding, vol. 71, no. 1, pp. 94-109 (1998)
8. Jegou, H., Douze, M., Schmid, C., Perez. P.: Aggregating local descriptors into a compact image representation. In Proc. CVPR (2010)
9. 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)
10. Mandasari, M.I., McLaren, M., van Leeuwen, D.A, Bingham, E., Mannila, H.: Random projection in dimensionality reduction: Applications to image and text data. 7th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, pp. 245–250 (2001)
11. Markatopoulou, F., Moumtzidou, A., Tzelepis, C., Avgerinakis, K., Gkalelis, N., Vrochidis, S., Mezaris, V., Kompatsiaris, I.: ITI-CERTH participation to TRECVID 2013. In TRECVID 2013 Workshop, Gaithersburg, MD, USA (2013)
12. Markatopoulou, F., Mezaris, V., Kompatsiaris, I.: A comparative study on the use of multi-label classification techniques for concept-based video indexing and annotation. In: C. Gurrin, F. Hopfgartner, W. Hurst, H. Johansen, H. Lee, and N. OConnor (eds.), MultiMedia Modeling, vol. 8325, pp. 1–12 (2014)
13. 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)
14. Johnson, S.C.: Hierarchical Clustering Schemes. Psychometrika, vol. 2, pp. 241-254 (1967)