IMOTION – Searching for Video Sequences using Multi-Shot Sketch Queries

Luca Rossetto[1], Ivan Giangreco[1], Silvan Heller[1], Claudiu Tănase[1], Heiko Schuldt[1], Stéphane Dupont[2], Omar Seddati[2], Metin Sezgin[3], Ozan Can Altıok[3], and Yusuf Sahillioğlu[3]

[1] Databases and Information Systems Research Group, Department of Mathematics and Computer Science, University of Basel, Switzerland
{luca.rossetto|ivan.giangreco|c.tanase|silvan.heller|heiko.schuldt}@unibas.ch
[2] Research Center in Information Technologies, Université de Mons, Belgium
{stephane.dupont|omar.seddati}@umons.ac.be
[3] Intelligent User Interfaces Lab, Koç University, Turkey
{oaltiok15|mtsezgin|ysahillioglu}@ku.edu.tr

Abstract. This paper presents the second version of the IMOTION system, a sketch-based video retrieval engine supporting multiple query paradigms. Ever since, IMOTION has supported the search for video sequences on the basis of still images, user-provided sketches, or the specification of motion via flow fields. For the second version, the functionality and the usability of the system have been improved. It now supports multiple input images (such as sketches or still frames) per query, as well as the specification of objects to be present within the target sequence. The results are either grouped by video or by sequence and the support for selective and collaborative retrieval has been improved. Special features have been added to encapsulate semantic similarity.

1 Introduction

In this paper we introduce the improvements made to the IMOTION system to adapt to the changed rules of the 2016 Video Browser Showdown [6] and to improve the system performance. With this version, we address the shortcomings of the 2015 edition of our system [4], especially in the textual challenges. We briefly discuss the architecture and implementation of the IMOTION system in Section 2 and elaborate on the changes made in this version in Section 3.

2 The IMOTION system

2.1 Architecture

The IMOTION system can be divided into a front-end and a back-end part. The back-end is based on the Cinest content-based video retrieval engine [3] which uses a multitude of different features in parallel to perform retrieval.

The front-end is browser-based. It communicates with the back-end through a web server which serves as a proxy for the retrieval engine while also serving static content such as preview images and videos.

In [4], we provide a more in-depth discussion of the architecture.

2.2 Implementation

The retrieval engine is written in Java and uses a customized version of PostgreSQL for storing all the feature data and meta-data. The adapted database provides various indexing techniques to index the feature data and by that decreases the retrieval time.

For object, scene, and action recognition, we train Convolutional Neural Networks (ConvNets) using the publicly available Torch toolbox [1].

3 New Functionality and User Interaction

This section outlines the various improvements we made to the system in comparison to the version which has participated to the 2015 VBS.

3.1 Multi-Shot Queries

An important new feature is the possibility to search for multiple shots in a single query. While the 2015 edition of the IMOTION system allowed only for one shot per query, the current version enables users to search for an arbitrary amount of (succeeding) shots. This greatly increases the overall expressiveness of a single query, especially when searching for heterogeneous video sequences, i.e., sequences which span several subsequent shots. In this case, separate query sketches can be provided for the different shots. Figure 1 shows a screen-shot illustrating a multi-shot query and Figure 2 shows the corresponding results.

3.2 Object recognition and retrieval

To augment the visual queries with semantic information, we use an object recognition system which is trained to recognize several hundred commonly seen objects in the video. For each shot of a video, all recognized objects and their positions within the frame are stored.

Even though the focus of the visual query specification lies on sketching, we decided against using sketch recognition [7] for query specification because of time constraints during the competition. Instead, the users will be presented with a list of clip-arts representing the recognizable objects which they can add to the query image via drag and drop.

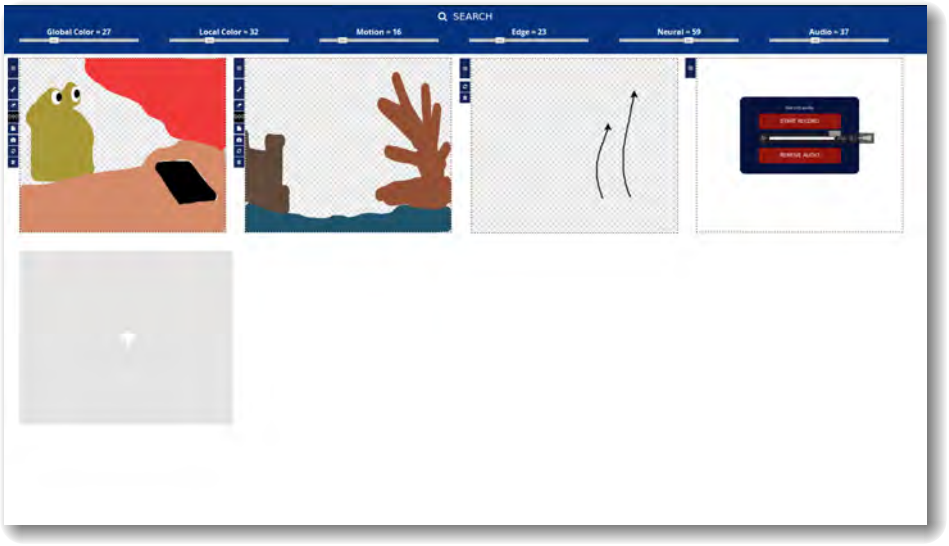

Fig. 1. Screen-shot of the IMOTION 2016 prototype UI

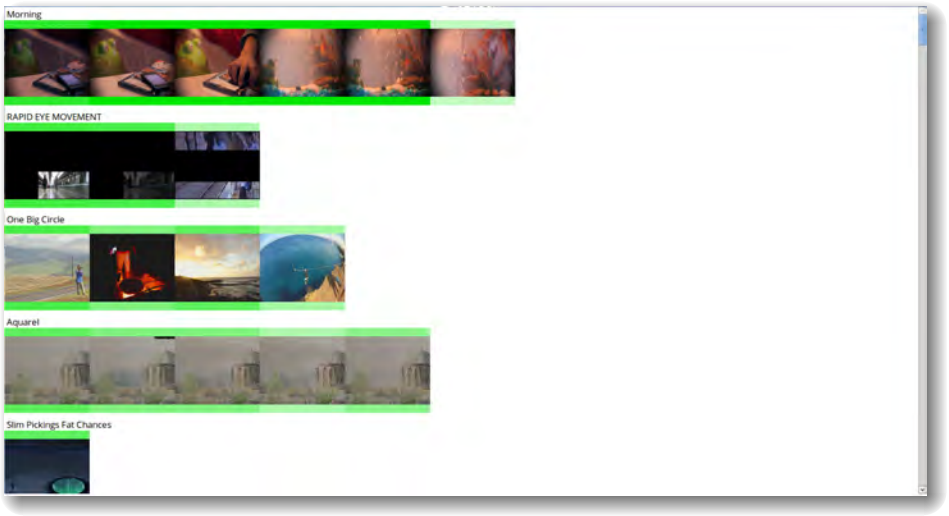

Fig. 2. Screen-shot of the corresponding result page

3.3 Result limitation and collaborative search

When refining a query, it is important to be able to limit the displayed results to a selected few or even one single video. The 2016 edition of IMOTION does not only support such selection based on retrieved results but also provides means to efficiently specify the relevant videos. The latter is important to support collaborative search which is actively supported in this version of the system. In case one user is sure to have found the correct video but not necessarily the correct sequence, other users can limit their search efforts to this video. This is achieved by representing the video designation to which the results should be limited in an efficient alphanumerical encoding.

3.4 Result presentation and browsing

The way retrieval results are displayed has been made more intuitive. Rather than showing isolated shots grouped by similarity measure, we now show the results grouped by video. The shots within a video are ordered chronologically and their score is indicated both with an overlay and by the color of their border. The videos are sorted based on the maximum score of their shots. It is also possible to perform a sequence segmentation of the results, breaking videos into multiple sequences with multiple shots each. This is particularly useful in cases where a query matches more than one sequence of the same video.

Additionally, improvements have been made to the way results are transferred from the back-end to the front-end. Results can now be streamed as they are generated which reduces the time required for the first results to appear. The front-end can display retrieved sequences as they come and re-order them to reflect their appropriate position in the growing set of results.

3.5 Additional and improved video features

To improve the flexibility in query specification, the previously used features have been extended to be able to deal with transparency in query images. This enables the user to have incomplete sketches which focus only on a part of the frame while ignoring the rest. The previous version of the system would not differentiate between an empty and a white area which could lead to unwanted results.

As in our previous system, we use two different ConvNets types for feature extraction. The first one for spatial information and the second one for temporal information. We use the output of neurons in a selected hidden layer as features. This time, however, and in order to speed up similarity search using those features, we reduced the dimensionality of the vectors by adding a bottleneck layer before the final fully connected classification layer. This solution ensures getting a shorter vector of features without degrading accuracy. This has been applied for the spatial ConvNets. No changes were needed for the temporal ConvNets as the classification task it is trained on only covers a much limited number

of classes (about 150 human actions) compared to the spatial on (about 1000 concepts).

For the spatial information, we actually use three different ConvNets enabling to highlight different facets of the content. We still use a ConvNet trained on the ImageNet dataset [5]. The large number of categories helps building good feature extractors. But unfortunately, most categories present in ImageNet are not of great interest for search in generic video databases. To improve on our previous system, we hence used an additional ConvNet trained on images downloaded from the internet and which correspond to the 1000 most frequent synsets of WordNet. In addition to training for object recognition, we use one more ConvNet trained to recognize the context/scene within an image. This ConvNet is trained on the Places dataset [9], containing examples of 205 scene categories and a total of 2.5 million images.

For the temporal feature extractor, we increased the number of recognized categories by merging two action recognition databases, namely HMDB-51 [2] and UCF101 [8]. As before, optical flows are extracted from video shots and used as input to our temporal ConvNet.

Finally, our new system also enables to use the audio channel of the video. The system extracts audio features: MFCC, Chroma and temporal modulation. This enables audio-based similarity search. The formulation of audio queries is also possible as the interface enables the user to record audio (vocal imitations of the sound of the video to be retrieved) using a microphone. We see this as a form of audio sketching, complementary to the image sketching used for specifying the visual content.

As before, depending on the weight that the user gives to the various feature sets, the system returns videos that have similarities according to different facets of the content.

4 Conclusions

The 2015 edition of the IMOTION system has already proven to be highly suitable for the VBS competition, especially for the visual part. With the 2016 edition, several improvements have been added to the functionality of the system, in particular to give a user more flexibility when specifying queries for heterogeneous video sequences, and we have improved the usability of the system.

Acknowlegements

This work was partly supported by the Chist-Era project IMOTION with contributions from the Belgian Fonds de la Recherche Scientifique (FNRS, contract no. R.50.02.14.F), the Scientific and Technological Research Council of Turkey (Tübitak, grant no. 113E325), and the Swiss National Science Foundation (SNSF, contract no. 20CH21_151571).

References

1. Ronan Collobert, Koray Kavukcuoglu, and Clément Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011.
2. Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 2556–2563, 2011.
3. Luca Rossetto, Ivan Giangreco, and Heiko Schuldt. Cineast: A multi-feature sketch-based video retrieval engine. In *Proceedings of the IEEE International Symposium on Multimedia (ISM'2014)*, pages 18–23. IEEE, 2014.
4. Luca Rossetto, Ivan Giangreco, Heiko Schuldt, Stéphane Dupont, Omar Seddati, Metin Sezgin, and Yusuf Sahillioğlu. IMOTION – a content-based video retrieval engine. In *MultiMedia Modeling*, pages 255–260. Springer, 2015.
5. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, pages 1–42, April 2015.
6. Klaus Schoeffmann, David Ahlström, Werner Bailer, Claudiu Cobârzan, Frank Hopfgartner, Kevin McGuinness, Cathal Gurrin, Christian Frisson, Duy-Dinh Le, Manfred Del Fabro, et al. The video browser showdown: a live evaluation of interactive video search tools. *International Journal of Multimedia Information Retrieval*, 3(2):113–127, 2014.
7. Omar Seddati, Stephane Dupont, and Said Mahmoudi. Deepsketch: Deep convolutional neural networks for sketch recognition and similarity search. In *Proceedings of the 13^{t}h International Workshop on Content-Based Multimedia Indexing (CBMI'2015)*, pages 1–6. IEEE, 2015.
8. Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.
9. Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 487–495. Curran Associates, Inc., 2014.