TEMPORAL AGGREGATION FOR LARGE-SCALE QUERY-BY-IMAGE VIDEO RETRIEVAL

André Araujo, Jason Chaves, Roland Angst, Bernd Girod

Stanford University, CA

ABSTRACT

We address the challenge of using image queries to retrieve video clips from a large database. Using binarized Fisher Vectors as global signatures, we present three novel contributions. First, an asymmetric comparison scheme for binarized Fisher Vectors is shown to boost retrieval performance by 0.27 mean Average Precision, exploiting the fact that query images contain much less clutter than database videos. Second, aggregation of frame-based local features over shots is shown to achieve retrieval performance comparable to aggregation of those local features over single frames, while reducing retrieval latency and memory requirements by more than 3X. Several shot aggregation strategies are compared and results indicate that most perform equally well. Third, aggregation over scenes, in combination with shot signatures, is shown to achieve one order of magnitude faster retrieval at comparable performance. Scene aggregation also outperforms the recently proposed aggregation in random groups.

***Index Terms*— ** image-based retrieval, temporal aggregation, video indexing, video search

1. INTRODUCTION AND RELATED WORK

This paper addresses the problem of retrieving relevant videos from a large database, using an image as the query. This is an important retrieval problem, enabling applications such as: searching video lectures using a slide, organizing video archives, advertisement monitoring, or content linking, where an image in a webpage could link to an online video. Several aspects distinguish this problem from the common problem of searching an image database using image queries. First, the database size: if videos are treated simply as a collection of individual and unrelated frames, searching a large video database might require extremely large memory footprint. Second, there is a clear asymmetry between the query, which is an image, and the database items, which are videos. Third, the visual information in the database is redundant, since videos are temporally coherent.

A first solution to this problem, Video Google as proposed by Sivic and Zisserman [1], was inspired by text retrieval systems based on the Bag-of-Words (BoW) model: low-level image features such as SIFT descriptors [2] are quantized into visual words and inserted in an inverted index. Query-by-image video retrieval has gained increased interest thanks to the TRECVID Instance Search challenge [3], where a query is composed of a small set of frames, together with region-of-interest masks. More recently, aggregating local features into global image signatures has shown great success in image retrieval [4, 5, 6]. Hence, in this work, we employ the state-of-the-art Scalable Compressed Fisher Vectors (SCFV) [6] as our aggregation method. Our paper presents three contributions, summarized in the following paragraphs.

Asymmetric comparison: A query image often only covers a fraction of its best-matching item in the database. That item therefore contains a fair amount of 'clutter', which leads to an asymmetry between the query and the entries in the database. For example, in query-by-image video retrieval, the Fisher Vector (FV) of a video segment might capture local features from several frames – making it highly likely that many of those features are not present in the query image. [7] has empirically verified this asymmetry in the context of the BoW model and proposed an asymmetric query-adaptive comparison. Analyzing a large number of bounding boxes in each database image is another recent strategy to handle this asymmetry [8], at the price of much increased complexity. As our first contribution, we instead propose an asymmetric comparison scheme for binarized Fisher Vectors in Sec. 3, which is simple to implement, accelerates retrieval, and provides substantial retrieval performance boost.

Evaluation of shot aggregation schemes: Aggregation of frame-based local features over shots is a well-established technique for improving video retrieval [9, 10]. In this work, we are interested in finding the best way to aggregate these features over shots – an evaluation which is performed using a large-scale database. For example, can we improve performance by tracking local features and collapsing features from the same track into a single unit before aggregation into a shot signature? A similar approach was explored in [11], but yielded poor results. We address this issue in Sec. 3 by comparing three different aggregation schemes which compute signatures from local features extracted from multiple frames in the same shot. Retrieval performance surprisingly hardly varies between those schemes – however, retrieval latency and memory requirements can be reduced by a factor of three, compared to a frame-based aggregation baseline. Note that such shot aggregation is different from the aggregation that is done in [1, 12], where local descriptors are aggregated over tracks, but each frame is indexed independently.

Scene aggregation schemes: Theoretically, when aggregating more and more local features, the resulting Fisher Vector should become less discriminative and eventually converge to the zero vector. We explore this behavior in Sec. 4 by aggregating frame-based local features over long video segments. Our experimental evaluation shows that signatures aggregated over scenes, even if they include many features, are still discriminative. These scene signatures can be combined with shot signatures in a retrieval system that achieves similar performance to frame-based schemes, but reduces computational cost by more than 10X. The aggregation of global signatures from several images into a single group signature has recently also been explored in the context of image retrieval [13]. That paper proposed a group testing framework, where the image collection is partitioned into random groups and a signature is computed for each group. We implement this method in the context of our problem, assigning shot signatures to random groups. Our experiments show that our proposed scene signatures outperform such random grouping.

In the remainder of this paper, we first present the experimental evaluation procedure common to all our experiments, and then present the individual contributions in detail.

2. EXPERIMENTAL SETUP

Our experiments make use of the Stanford I2V dataset [14]. The light version of the dataset is used for experimentation with different techniques and parameters throughout the paper, and we provide results for selected methods on the full dataset. The Stanford I2V dataset is much larger than similar-purpose ones: the full (light) version contains 3,801 (1,035) hours of video in the database, spread across 84,443 (23,437) clips, and 229 (78) queries. Using a 1 fps frame rate, we obtain 13,966,820 (3,808,760) video frames. In this dataset, each video clip corresponds to a 'scene' [15, 16, 17], i.e., a concise segment of video that contains interrelated shots and represents a semantic unit for the given type of content. In this case, the scenes correspond to news stories.

We consider the problem of *Scene Retrieval*: given an image query, the system is expected to retrieve the relevant scenes in the database. In the following sections, we experiment with global signatures based on shots, scenes, and individual frames. Although these signatures can be quite different, the retrieval process is the same: the image query signature is compared to every signature in the database. Performance is evaluated using mean Average Precision (mAP) over a ranked list of the top 100 retrieved scenes. If frame (or shot) signatures are used, a scene score is defined as the best score among its frames (or shots).

We use SIFT detector and descriptors [2] in all experiments. The 128D SIFT descriptors are reduced to 32D with a PCA step. Those dimensionality-reduced local descriptors are then aggregated into global signatures using the state-of-the-art Scalable Compressed Fisher Vector (SCFV) framework [6], which has been selected by the MPEG Compact Descriptors for Visual Search (CDVS) subgroup for adoption into the CDVS Test Model. The retrieval procedure can be performed efficiently, since bitwise comparisons can be evaluated with dedicated CPU instructions. Note that additional speed-up could be obtained for any of the techniques presented in this work by using complementary techniques, such as an inverted index [6], multi-round scoring [5] or parallelization. Similarly, we do not make use of reranking strategies such as geometric consistency checks based on local features, which could also improve performance for all of the methods presented in this paper.

In our experiments, we consider retrieval performance as a function of the required number of bitwise comparisons, which is the number of bits in all database signatures that are used during retrieval. For example, the use of 1 fps frame-based SCFV signatures with 512 Gaussians on the light dataset would require 6.2×10^{10} operations. For further reference, in this baseline configuration, our system takes on average 15 seconds per query on a single thread on an Intel Xeon 2.4GHz processor.

3. SHOT AGGREGATION

In this section, we evaluate the aggregation of frame-based local features over shots. Shot boundary detection is performed by comparing HSV histograms using L1 distances, which finds 1,185,227 shots in our database, with an average duration of 3.42 seconds. A scene contains 50.57 shots on average. We experiment with 4 different aggregation modes. Our emphasis here is on a large-scale evaluation of shot aggregation schemes.

3.1. Aggregation modes

Local feature aggregation (LOC): All features from a selected number of frames per shot are aggregated into a single SCFV signature per shot.

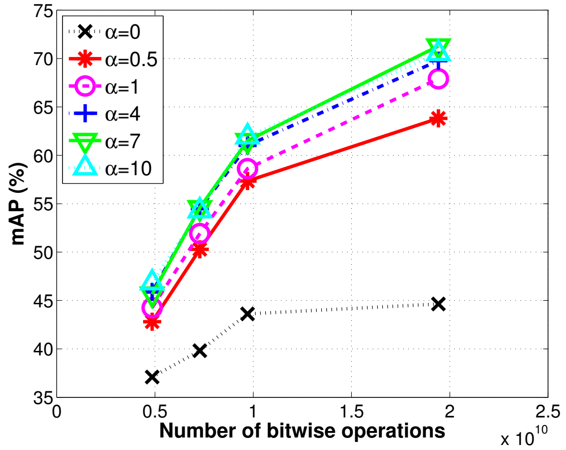

Fig. 1: Retrieval performance on the light dataset as a function of the required number of bitwise comparisons and α, using shot aggregation with LOC mode and 10 frames per shot. For each curve, we vary the number of Gaussians used in the global signature within $\{128, 192, 256, 512\}$.

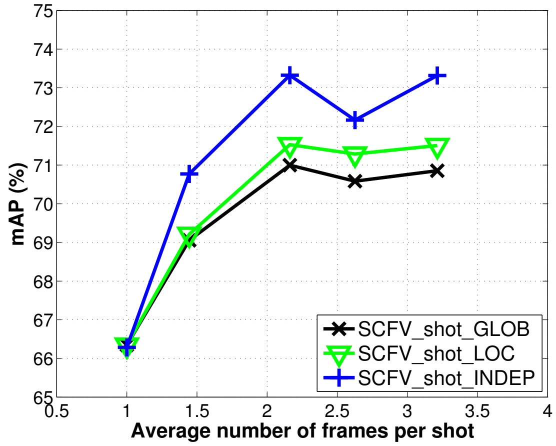

Fig. 2: Retrieval performance on the light dataset as a function of the average number of used frames per shot, using 512 Gaussians, for different shot aggregation modes. For each curve, we vary the selected number of frames per shot within $\{1, 2, 5, 10, all\}$.

Global signature aggregation (GLOB): First, we extract FV signatures for a selected number of frames per shot. These FV signatures are summed and the resulting floating-point vector is binarized based on each dimension's sign – similar to SCFV's binarization.

Tracking-based aggregation (TRACK): First, we find tracks within each scene, by grouping extracted keypoints in consecutive frames if they are similar and nearby. Second, the local descriptors within a track are averaged within each shot. Finally, the averaged track descriptors in a shot are aggregated into a shot SCFV signature.

Independent frame aggregation (INDEP): SCFV signatures for a selected number of frames per shot are used. In this case, no shot aggregation is performed. This mode is used to test whether shot-based signatures have an advantage over simply keeping some frame-based signatures per shot.

For modes LOC, GLOB and INDEP, we use a selected number of frames per shot. In this case, frames are selected by subsampling the 1 fps stream in regular intervals within the shot. In case there are fewer frames in a shot than the selected number of frames, we simply use all frames from the given shot.

3.2. Asymmetric Comparisons

In query-by-image video retrieval, a query and its matching database entry often have a 'containment relationship': for example, a large

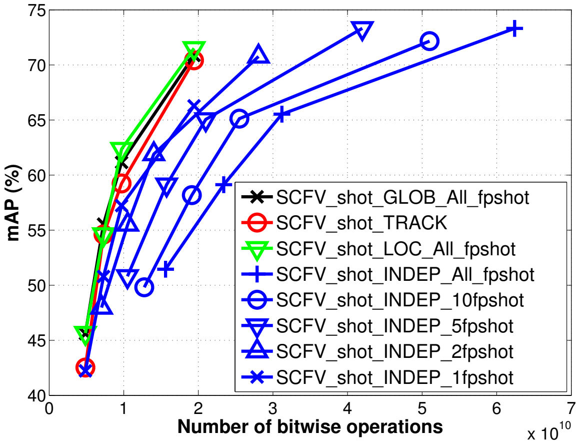

Fig. 3: Retrieval performance on the light dataset as a function of the required number of bitwise comparisons, for different shot aggregation modes. For each curve, we vary the number of Gaussians used in the global signature within $\{128, 192, 256, 512\}$.

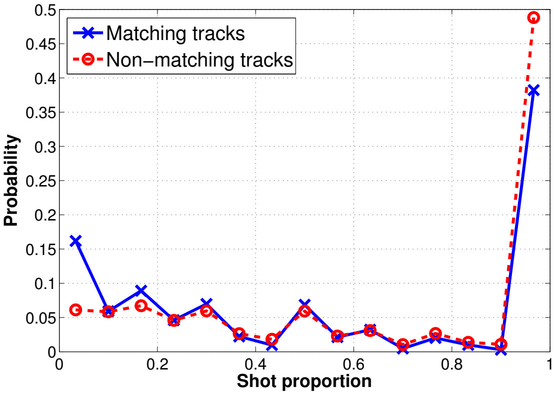

Fig. 4: Normalized histogram of track lengths, as a proportion of shot lengths, on the Stanford I2V light dataset [14], for matching and non-matching database tracks.

part of a query image may be contained in a relevant database shot, while the reverse is not true. This asymmetry can be exploited to boost retrieval performance. Specifically, the presence of elements in a query signature but absence in a database signature should be penalized more heavily than the absence of elements in a query signature and presence in a database signature. This was explored in [7], in the context of retrieval using the BoW framework. We propose an asymmetric comparison scheme for state-of-the-art FV-based methods. Note that we use FVs that make use of gradients only with respect to the mean, which is the most common setting [4].

Consider the d-dimensional gradient with respect to the i-th Gaussian in the model, denoted by $\mathcal{G}_i$ (in our case, $d = 32$). These d components in the *query* signature are taken into account if and only if $\|\mathcal{G}_i\|_1 > \alpha$, where α is a parameter we set empirically – whose optimal value might depend on the type of aggregation. If $\|\mathcal{G}_i\|_1 \leq \alpha$, this indicates that information about local descriptor patterns considered by the i-th Gaussian is not strongly present in the query – thus, we simply ignore those d components, not taking them into account when comparing query and database signatures. Fig. 1 presents results using LOC mode with 10 frames per shot, varying α and the number of Gaussians. Note that using $\alpha = 0$ corresponds to not using asymmetric comparisons. Substantial performance improvements are obtained, of up to 0.27 mAP. This result might be surprising considering that we are discarding information from some Gaussians in the model, i.e., using a smaller bitrate, yet boosting retrieval performance.

We decide to use $\alpha = 7$ in all shot-based experiments – this value provides near-optimal mAP for all modes. Similar improvements can be obtained for frame-based and scene-based signatures. We use $\alpha_{frame} = 8$ for frame-based signatures and $\alpha_{scene} = 6$ for scene-based signatures in all experiments (unless otherwise specified), as these values provide best mAP for these methods. Another benefit of this asymmetric comparison technique is that the number of required bitwise comparisons slightly decreases. Note that we have decided not to reflect this decrease in our plots, since the gain is query-dependent and makes the interpretation of the plots more complex.

3.3. Comparison of different modes

For modes LOC, GLOB and INDEP, we experiment with selecting 1, 2, 5, 10 or *all* frames within a shot. Note that INDEP with *all* frames per shot is exactly the same as indexing each frame in the database with a frame-based signature. Fig. 2 presents results for these modes as we vary the selected number of frames per shot, using 512 Gaussians in the global signatures. The actual average number of frames per shot that ends up being used is much smaller than the selected number, since many shots are very short. In general, significant gains can be obtained by using multiple frames per shot, of up to 0.07 mAP. This gain agrees with [18], which reports improved performance when more frames are used.

In Fig. 3, we compare retrieval performance of the 4 modes in terms of the number of bitwise operations in the retrieval process, by varying the number of Gaussians. For GLOB and LOC modes, we only show results selecting *all* frames per shot, since this selection is among the best, as in Fig. 2. Note that the number of bitwise operations for modes GLOB and LOC is independent of the selected number of frames per shot. Performance is similar for modes GLOB, LOC and TRACK. INDEP mode can achieve similar performance, but at the cost of more comparisons during the retrieval process. More specifically, using 512 Gaussians, INDEP-*all* achieves 0.02 mAP improvement over LOC-*all*, at 3.21X slower retrieval. Overall, our experiments indicate that shot-based aggregation might be preferred over frame-based aggregation, since the former achieves comparable retrieval performance with much smaller computational complexity and memory footprint.

The fact that TRACK mode works similarly to GLOB and LOC modes can be understood by considering Fig. 4, which presents histograms of database track lengths, as a proportion of shot lengths. In this plot, we distinguish between matching and non-matching database tracks, i.e., tracks whose local features can be successfully mapped to a query local feature or not – this is found by a matching process between query images and ground-truth video frames, based on local features and using geometric consistency checks, as in [2]. Given the assumption that long tracks are more often non-matching tracks, the use of TRACK mode could reduce their influence such that the global signatures would be less influenced by clutter. However, the statistics of matching and non-matching tracks are very similar, and we cannot expect much difference in retrieval performance.

4. SCENE AGGREGATION

Shot aggregation allows 3X retrieval speed-up, while achieving high mAP. In this section, we consider ways to reduce retrieval complexity even further, by developing scene signatures. These allow a very fast first pass over the database to select a small number of candidate scenes that will be post-processed. First, we consider grouping shot signatures for faster retrieval. Then, we investigate different design choices for scene signatures.

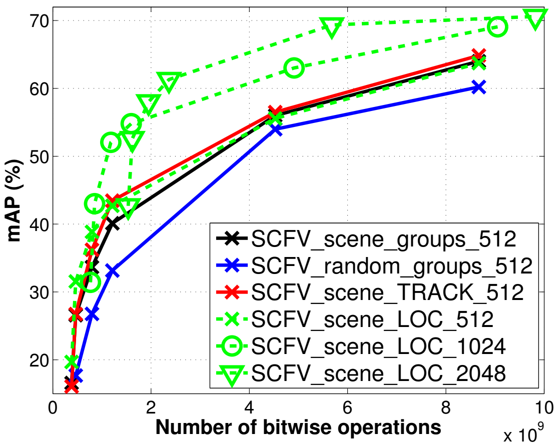

Fig. 5: Retrieval performance on the light dataset as a function of the required number of bitwise comparisons, for different group aggregation modes. For each curve, we vary the number of groups that are re-ranked within $\{0, 100, 500, 1000, 5000, 10000\}$.

4.1. Grouping shot signatures

In recent work, Shi et al [13] present a group testing framework to speed up image retrieval in large databases. Their approach works by grouping database images at random, and summing up their FVs to construct a single signature for each group of images. At querying time, the exact similarity scores for images in top-ranked groups are computed, and these images are ranked according to their similarity.

We consider a variant of this set-up, where we want to group shot signatures to obtain faster retrieval. In our case, however, we can take advantage of the underlying structure from the video database: We propose to assign each shot signature to a group constituted by all shots from its scene. In other words, each group corresponds to a scene, and its group signature is constructed from the signatures of each shot belonging to that scene.

We compare this approach to [13]'s random grouping. Instead of raw FVs, we use SCFV signatures, to allow fast retrieval in a large database. FVs for each shot signature are computed and a group signature is obtained by first summing up the constituent FVs, then binarizing the result based on each component's sign. We assign each shot to one group – as assigning each item to one group shows near-optimal performance in [13]. For a fair comparison, the number of random groups is set to 23,437, the exact number of scenes in our database. During retrieval, we re-rank a certain number of top-ranked groups by obtaining the exact similarity scores for each of its constituent shots, similar to [13]'s procedure. For the shot signatures, we use LOC mode with 10 frames per shot and 512 Gaussians.

Note that scene groups can be scored directly, without re-ranking (i.e., re-ranking 0 groups) – since a ranked list of scenes is directly obtained even without re-ranking in this case. This cannot be done for random groups, in which case it is necessary to obtain shot scores to then be able to rank scenes. Fig. 5 presents curves obtained by varying the number of re-ranked groups. It shows clearly that scene groups outperform random groups, for the setting with 512 Gaussians, by up to 0.09 mAP. This is intuitive, as scene groups might contain several shots in which the query is shown, increasing the chance that a true matching scene will be ranked higher.

4.2. Scene signature design

We consider other scene signature designs. In particular, we experiment with scene signatures constructed directly from frame-based local features. The experimental setup here is the same as in the previous subsection, with the only change being the scene signature that is utilized.

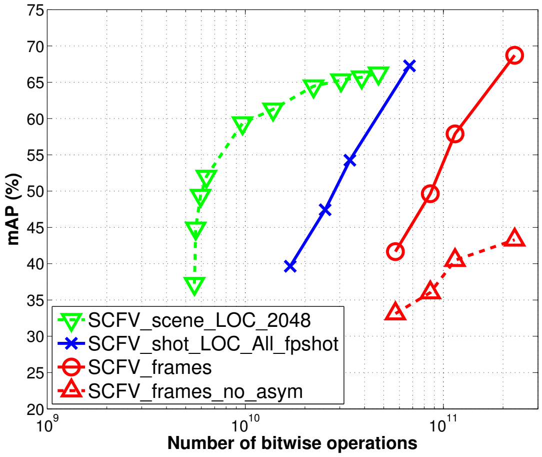

Fig. 6: Retrieval performance on the full dataset as a function of the required number of bitwise comparisons, for the best scene- and shot-based aggregation methods, compared to the baseline frame-based methods with and without asymmetric comparisons. In this case, the re-ranking for the scene-based scheme uses shot signatures with mode LOC and all frames per shot. The number of re-ranked scenes is varied from 0 to 50000. Note the logarithmic scale for the x-axis.

Scene signatures from local features: We experiment with two other scene signature aggregation modes: LOC (in this case, using all frames in the scene), and TRACK. Aggregation happens in the same way as presented in 3.1, except that it is now done over scenes, not shots. In this experiment, we once again use 512 Gaussians, for a fair comparison against scene groups. Fig. 5 shows an overall similar retrieval performance of scene groups, LOC and TRACK. LOC presents a slight advantage when a small number of scenes are re-ranked, of up to 0.05 mAP compared to scene groups.

Scene signatures with higher number of Gaussians: For further performance improvement, we increase the number of Gaussians used in scene signatures to 1024 and 2048, using LOC aggregation mode with all frames per scene. Fig. 5 shows a substantial performance boost, achieving more than 0.10 mAP improvement over LOC using 512 Gaussians.

Comparison of frame-, shot- and scene-based aggregation on the full dataset: Fig. 6 presents a comparison between the baseline frame-based and the best scene-based and shot-based approaches, on the full dataset. Using scene signatures, we are able to achieve 10.35X faster retrieval with a small mAP drop (0.04 mAP), compared to the frame-based method that uses asymmetric comparisons.

5. CONCLUSION

In this work, we introduce new temporal aggregation strategies for query-by-image video retrieval. We demonstrate substantial improvement in retrieval quality for systems based on binarized Fisher Vectors when using an asymmetric comparison technique. Several shot-based aggregation techniques are evaluated and shown to achieve similar retrieval performance to frame-based aggregation schemes, with a 3X speed-up. Scene-based aggregation is introduced and shown to outperform a grouping scheme based on random assignments. Using scene signatures in combination with shot signatures, we design a retrieval system that achieves 0.21 mAP improvement and one order of magnitude smaller computational cost, compared to a baseline frame-based scheme that does not use asymmetric comparisons.

6. REFERENCES

[1] J. Sivic and A. Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos," in *Proc. ICCV*, 2003.

[2] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, 2004.

[3] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quenot, "TRECVID 2014 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics," in *Proc. TRECVID*, 2014.

[4] H. Jégou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, "Aggregating Local Image Descriptors into Compact Codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, 2012.

[5] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, R. Vedantham, R. Grzeszczuk, and B. Girod, "Residual Enhanced Visual Vector as a Compact Signature for Mobile Visual Search," *Signal Processing*, vol. 93, no. 8, 2013.

[6] L. Duan, J. Lin, J. Chen, T. Huang, and W. Gao, "Compact Descriptors for Visual Search," *IEEE Multimedia*, vol. 21, no. 3, 2014.

[7] C.-Z. Zhu, H. Jegou, and S. Satoh, "Query-Adaptive Asymmetrical Dissimilarities for Visual Object Retrieval," in *Proc. ICCV*, 2013.

[8] R. Tao, E. Gavves, C. G. M. Snoek, and A. W. M. Smeulders, "Locality in Generic Instance Search from One Example," in *Proc. CVPR*, 2014.

[9] C.-Z. Zhu, H. Jegou, and S. Satoh, "NII team: Query-Adaptive Asymmetrical Dissimilarities for Instance Search," in *Proc. TRECVID*, 2013.

[10] N. Ballas, M. Redi, A. Hamadi, B. Gao, B. Labbe, B. Merialdo, C. Zhu, L. Chen, B. Safadi, N. Derbas, Y. Tang, A. Benoit, H. Le Borgne, B. Mansencal, E. Dellandrea, J. Benois-Pineau, P. Lambert, P. Gosselin, M. Budnik, T. Strat, D. Picard, S. Ayache, G. Quenot, and C. Bichot, "IRIM at TRECVID 2014: Semantic Indexing and Instance Search," in *Proc. TRECVID*, 2014.

[11] A. Bursuc and T. Zaharia, "ARTEMIS at TRECVID 2013: Instance Search Task," in *Proc. TRECVID*, 2013.

[12] A. Araujo, M. Makar, V. Chandrasekhar, D. Chen, S. Tsai, H. Chen, R. Angst, and B. Girod, "Efficient Video Search Using Image Queries," in *Proc. ICIP*, 2014.

[13] M. Shi, T. Furon, and H. Jégou, "A Group Testing Framework for Similarity Search in High-dimensional Spaces," in *Proc. ACM-MM*, 2014.

[14] A. Araujo, J. Chaves, D. Chen, R. Angst, and B. Girod, "Stanford I2V: A News Video Dataset for Query-by-Image Experiments," in *Proc. ACM Multimedia Systems Conference*, 2015.

[15] M. Yeung, B.-L. Yeo, and B. Liu, "Segmentation of Video by Clustering and Graph Analysis," *Computer Vision and Image Understanding*, vol. 71, no. 1, 1998.

[16] Z. Rasheed and M. Shah, "Scene Detection In Hollywood Movies and TV Shows," in *Proc. CVPR*, 2003.

[17] B. Truong, S. Venkatesh, and C. Dorai, "Scene Extraction in Motion Pictures," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 1, 2003.

[18] C.-Z. Zhu, Y.-H. Huang, and S. Satoh, "Multi-Image Aggregation for Better Visual Object Retrieval," in *Proc. ICASSP*, 2014.