ARTEMIS-UBIMEDIA at TRECVid 2011: Instance Search

Andrei Bursuc[1,2], Titus Zaharia[1], Olivier Martinot[2]

[1]Institut Télécom ; Télécom SudParis, ARTEMIS Department, UMR CNRS 8145 MAP5,
9 rue Charles Fourier, 91011 Evry Cedex, France
{Andrei.Bursuc, Titus.Zaharia}@it-sudparis.eu
[2] Alcatel-Lucent Bell Labs France, route de Villejust 91620 Nozay, France
Olivier.Martinot@alcatel-lucent.com

Abstract. This paper describes the approach proposed by ARTEMIS-UBIMEDIA team at TRECVID 2011, Instance Search (INS) task. The method is based on a semi-global image representation relying on an over-segmentation of the keyframes. An aggregation mechanism was then applied in order to group a set of sub-regions into an object similar to the query, under a global similarity criterion.

1 Introduction

Object retrieval in videos is among the most challenging tasks up to date in computer vision. In the last years an increasing number of solutions have provided a variety of satisfying results for concept detection in videos [1]. Yet, retrieving different instances of the same object in video sequences still remains an open issue. The main difficulty is related to the specification of semi-global image representations that need to be considered, together with the elaboration of efficient partial matching strategies. In addition, variations in visual appearance and object's pose have to be taken into account appropriately. Existing methods for object indexing and retrieval such as the discriminatively trained deformable part-based models [2] or the bag-of-words representations [3] cannot be applied successfully in all cases as they rely on classification and machine learning methods. While for specific object category retrieval the results of such techniques are encouraging, it is difficult to train an algorithm for any object the user might want to search for. This relatively recent topic of research has been considered in the TRECVID 2010 [4] evaluation campaign, under the so-called instance search task, and TRECVID continued it in the 2011 edition.

This paper describes the work of ARTEMIS-Ubimedia in the Instance Search Task of the TRECVID 2011 campaign. In the following sections we will present in detail the applied algorithms and the evaluation for the runs we performed.

2 Instance Search Task Presentation

Instance Search (INS) is a pilot task introduced in the TRECVid 2010 campaign and continued in the 2011 campaign. Given a collection of test video clips and a collection of queries that delimit a person, object, or location in some example video, participant applications have to locate for each query up to 1000 clips most likely to contain a recognizable instance of the entity. A number of 25 queries have been specified, each consisting of a set of 2 to 6 example frame images drawn at interval from a video containing the item of interest. The BBC rushes dataset was proposed for this task with a total of 20982 short clips. Different transformations were applied to some random test clips in order to increase the difficulty of the task.

The main objective was to explore task definition and evaluation issues. Thus only a rough estimate of searched instances locations was asked. Participants had only to find the clips where the instance appeared, but not the precise location and time stamp of the instance in the video clips.

3 Approach Overview

For our approach, we have considered a limited number of keyframes per shot (up to 4). We have then over-segmented each such keyframes in order to obtain a semi-global image representation. An aggregation mechanism was then applied in order to group a set of sub-regions into an object similar to the query, under a global similarity criterion. Our strategy relies on a greedy dynamic region construction method.

The main aspects of our approach are presented in detail below. Let us start by detailing the color-based representation used.

3.1 DCD Representation

The object search process is performed uniquely upon the obtained key-frames in order to reduce the computational complexity. Each key-frame is segmented by applying the Mean Shift technique proposed in [5]. Let us mention that other segmentation methods can be used as well. Each region (or segment) determined is described by a unique, homogeneous color, defined as the mean value of the pixels of the given region. The set of colors, together with their percentage of occupation in the image (*i.e.*, the associated color histogram) are regrouped into a visual representation, which is similar to the MPEG-7 Dominant Color Descriptor (DCD). More precisely, let $C_I = \{c_1^I, c_2^I, \ldots c_{N_I}^I\}$ be the set of N_I colors obtained for image I, and $H_I = (p_1^I, p_2^I, \ldots p_{N_I}^I)$ the associated color histogram vector. The visual image representation is defined as the couple (C_I, H_I). An arbitrary number of dominant colors is supported, in contrast with the MPEG-7 DCD, where the maximal number of colors is limited to eight. In our experiments, we have used up to 250 dominant colors for each frame (Fig.1). We can observe that despite the inherent loss in accuracy, the image content can still be visually recognized from the segmented images.

Fig. 2. Video frames (left) and their segmentations (right). A number of up to 250 segmented regions provides in this case a "recognizable" image.

The query is, by definition, an object of arbitrary shape and is processed in the same manner in order to derive its visual representation. The advantage of the DCD representation comes from the fact that objects with arbitrary numbers of colors can be efficiently compared by using, for example, the Quadratic Form Distance Measure introduced in [6], which can be re-written for arbitrary length representations as described by the following equation:

$$D_h^2(H_Q, H_I) = \sum_{i=1}^{N_Q}\sum_{k=1}^{N_Q} a(c_i^Q, c_k^Q)\, p_i^Q p_k^Q + \sum_{j=1}^{N_I}\sum_{l=1}^{N_I} a(c_j^I, c_l^I)\, p_j^I p_l^I - \sum_{i=1}^{N_Q}\sum_{j=1}^{N_I} a(c_i^Q, c_j^I)\, p_i^Q p_j^I \quad , \tag{1}$$

where $H_Q = (p_1^Q, p_2^Q, \ldots p_{N_Q}^Q)$ and $H_I = (p_1^I, p_2^I, \ldots p_{N_I}^I)$ respectively denote the DCD histogram vectors of length N_Q, and N_I respectively associated to the query (Q) and candidate (I) images. The function a, describe the similarity between two colors c_i and c_j and is defined as*:*

$$\mathrm{a}(c_i, c_j) = 1 - \frac{d(c_i, c_j)}{d_{max}} \tag{2}$$

where d is the Euclidean distance between colors c_i and c_j and d_{max} is the maximum Euclidean distance between any 2 colors in the considered color space (*e.g.*, for the RGB color space $d_{max} \cong 442$).

Let us note that each color region in a candidate image has a specific contribution to the global distance. Thus, the contribution of color c_j^I in an image I to the global distance between image I and query Q is defined as:

$$C(c_j^I, Q) = \sum_{l=1}^{N_I} a(c_j^I, c_l^I)\, p_j^I p_l^I - \sum_{i=1}^{N_Q} a(c_i^Q, c_j^I)\, p_i^Q p_j^I \tag{3}$$

The above-defined distance is used as a global criterion in the matching stage. Here, the objective is to determine, in each key-frame of the considered video sequence, candidate regions visually similar with the query.

3.2 Offline processing

Our method does not use any training or classification method and each query is searched within all the keyframes in the dataset. In order to improve the execution time of our methods, we have reduced the number of keyframes to be stored in the indexed database. The work flow of the offline processing of the videos is illustrated in Fig. 2.

Most of the clips from the BBC Rushes dataset consisted of sequences containing usually one or two shots. We have extracted from these clips up to 4 frames at equal intervals of time. Furthermore, we have filtered out the near-duplicate frames by using a color histogram distance computed between all the frames extracted from a video clip. A total of 34614 distinct keyframes were retained and then segmented. We have resized the videos to 352×288 pixels, which corresponds to the resolution of a majority of movies among the video clips in the data set. We have left unchanged the frames with lower resolutions and modified the segmentation parameters to generate a proper number of color regions.

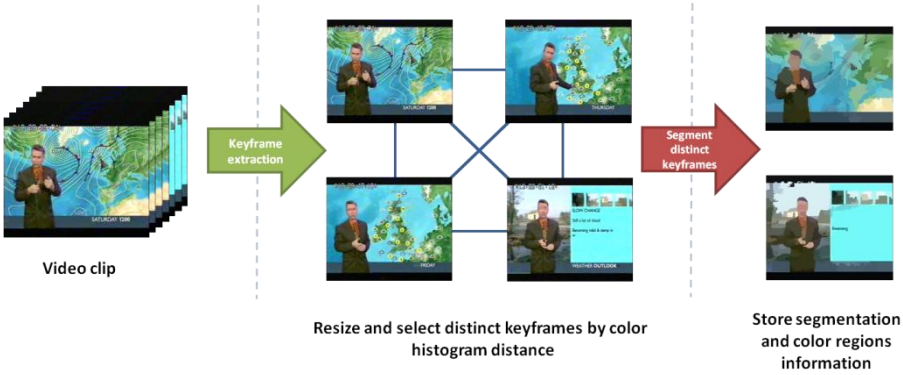

Fig. 2. Offline processing flow.

3.3 Dynamic Region Construction

First, we filter out the far-off colors, based on a color similarity criterion. A permissive threshold is here used, the goal being to reject highly improbable colors but in the same time to keep a sufficiently large variety of candidate regions. We then label the rest of the regions in connected components and consider each of them as a candidate for our query.

Next, the objective is to develop a dynamic region construction algorithm. This represents the core of the proposed methodology. In this stage, the candidate object is iteratively refined by removing and adding individual segments until the global matching distance is minimized. To achieve this goal we have tested a recursive greedy based optimization method [7]. An overview of the proposed approach is illustrated in Fig.3.

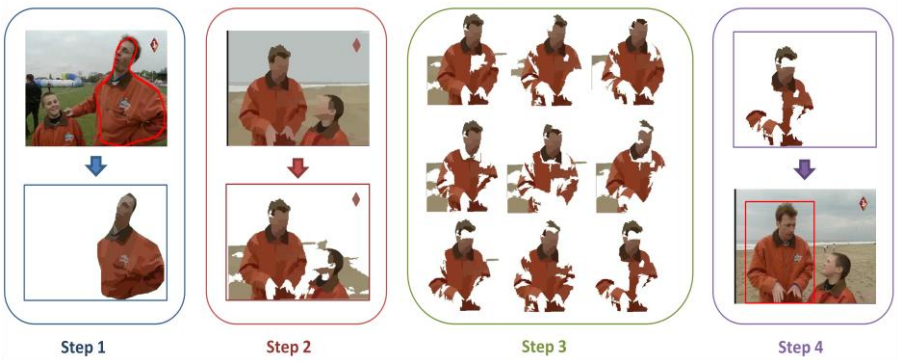

Fig. 3. Overview of the algorithm: (1): The query mask is used to crop out the corresponding segmented regions of the query; (2): Regions with colors highly different from the query are filtered out; (3): Different configurations of candidate objects are generated by adding/removing color segments; (4): The object with the minimal score is selected and displayed in its bounding box.

3.3.1 Relaxed Greedy Scheme

The algorithm starts from the initial set of regions obtained after the filtering stage described in the previous section. At each stage, we consider the current candidate object in image I and attempt to improve the current similarity measure between query and candidate objects. More precisely, we recursively eliminate the color segment which provides the highest contribution to the global distance (equation 3). We then check if the global distance is decreasing or not. If yes, we eliminate the corresponding region, update the color frequency vector H_I, and re-iterate the algorithm on the new candidate object obtained. If not, the region is maintained and the algorithm successively tries to eliminate the following regions (sorted by decreasing order of their contribution to the global distance). Each time an attempt to eliminate a segment is performed, the region connectivity needs to be re-calculated in order to determine the eventually newly created connected components. Each connected component is then treated separately.

Concerning the exit condition, we constrain the algorithm to stop generating configurations when the current distance is "considerably" higher than the previous one. We consider that if the current distance is with δ % higher than the previous obtained one, the candidate object has a low probability of reaching a configuration with a better score. The algorithm should in this case stop and return the current best distance. Otherwise, it should continue removing the regions with the highest contributions to the score as it could find another minimum after this "uphill" configuration. In our submission, we have considered values of 10% and 20% for δ, which provide a good trade-off between the number of generated configurations and the computational time [7].

The strategy of recursively eliminating the highest contributor to the global score increases the speed of the algorithm, by pruning the search space. However, the main limitation of the greedy-based approach is that it does not ensure the retrieval of an

optimal solution. In order to achieve asymptotic optimality, we have adopted a simulated annealing matching strategy, described in the next section.

3.4 Run and Fusion Methods

We have submitted for evaluation 4 runs: ARTEMIS-UBIMEDIA_RG20A (1), ARTEMIS-UBIMEDIA_RG20M (2), ARTEMIS-UBIMEDIA_RG10A (3), ARTEMIS-UBIMEDIA_RG10M (4).

Runs (1) and (2) are based on the Relaxed Greedy scheme with the δ tolerance coefficient of 20%, while runs (2) and (4) use a δ of 10%.

For all runs, we have launched individual queries for each example of the given topics and then performed a fusion the returned rank lists. In the case of runs (1) and (3), we have computed for each clip the average of its scores obtained from each query belonging to the same topic. The newly obtained rank list was then re-ordered and the first 1000 clips were returned. For runs (2) and (4) we have selected the best score obtained by each shot for all queries belonging to the same topic. The first 1000 clips were then written to the rank list.

4 Results and Discussion

The results for the queries and the mean MAPs of the query types are illustrated in Fig. 4.

We can notice that in most of the cases, runs (2) and (4) provided better results. Examples from the same topic can vary in size, pose, lighting and color (*e.g.* TV presenters, actors with different clothing) and thus the scores of each clip can vary accordingly. Therefore, for the runs that compute the mean of all scores, the best score can fade during the averaging, while for the runs selecting only the best score, the chances of pushing up in the rank list the good matches are higher.

The size of the query examples plays an important role as the LOCATION type queries (*e.g.* inside the windmill) covering almost an entire frame, returned the best results in our runs. On the other hand queries of small sizes with little color information (*e.g.* lantern, all-yellow balloon) were nearly not found.

A considerable number of positive clips were not retrieved as the queried object could not be found within the considered keyframes. We think that a higher frequency keyframe sampling (at least one frame every 2-3 seconds) could improve the accuracy of the method. Moreover, we have noticed that some of the query objects (especially the small scale ones) could not be found within the positive frames due to the considered segmentation. For future improvements, we consider using higher resolution images and configure the segmentation to return up to 300 color regions per frame.

Although we have set different segmentation configurations for the small resolution videos (down to 176×144), our region based method could not distinguish properly the relevant objects. In this case the segmentation based method should be complemented by finer descriptors (*e.g.* interest point descriptors)

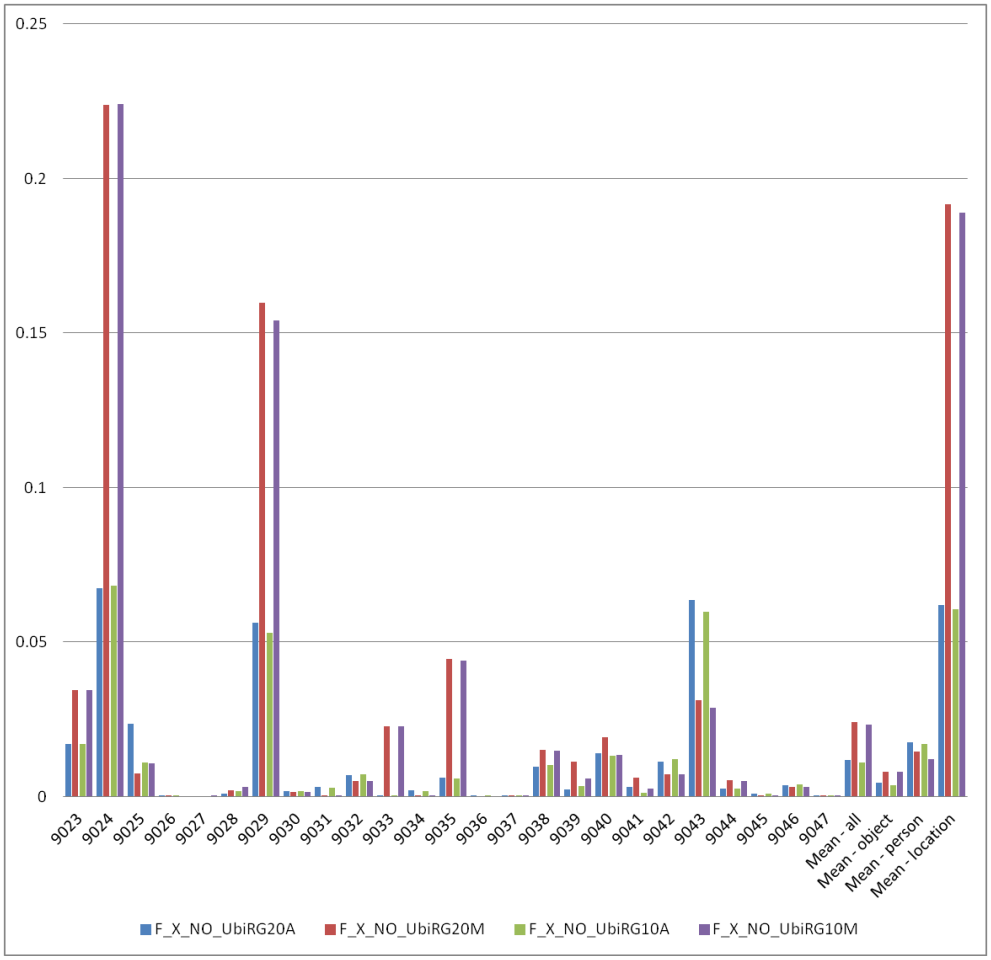

Fig. 4. MAP for the different runs for each of the topics and the mean of topic category and overall

Concerning the difference between the runs using 20% or 10%.values for δ, the 20% one provides slightly better results. A higher number of color regions per frame should widen the gap between the two methods in favor of the 20% one.

We are planning to perform additional experiments using more advanced global optimization methods such as Simulated Annealing [7][8] and also to improve the drawbacks that we have noticed in the first experiments (*e.g.* higher number of keyframes, higher number of color regions per keyframe, different color spaces).

Perspectives of improvement concern the embedding on spatial information in our region based approach. Notably, we are considering an adaptation of the GraphCut technique based on Markovian modeling. We also have to study how to take advantage of the several example images of a given topic and how to fusion the results for queries using different descriptors and different images belonging to the same topic.

5 Conclusion

In this paper we presented our experiments performed in the Instance Search of the TRECVid 2011 campaign. The participation in the TRECVid campaign represented for us a rewarding experience in advancing forward our research and in finding new ideas and research directions in the challenging domain of object-based video retrieval.

Acknowledgments. The current work has been developed within the framework of the UBIMEDIA Common Laboratory established between Institut TELECOM and Alcatel-Lucent Bell Labs France.

References

1. Snoek, C.G.M., Worring, M.: "Concept-Based Video Retrieval," Foundation and Trend in Information Retrieval, Vol.2, No.4 (2008), pp. 215-322.
2. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: "Object Detection with Discriminatively Trained Part Based Models," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, September 2010
3. Sivic, J. and Zisserman, A.: "Video Google: A text retrieval approach to object matching in videos," IEEE International Conf. on Computer Vision (ICCV'03), 2003.
4. Smeaton, A. F., Over, P., and Kraaij, W.: "2006. Evaluation campaigns and TRECVid," In Proc. 8th ACM International Workshop on Multimedia Information Retrieval (USA, October 26 - 27, 2006). MIR '06. ACM Press, New York, NY, pp. 321-330.
5. Comaniciu, D., Meer, P.: "Mean Shift: A Robust Approach Toward Feature Space Analysis," IEEE Tran. on Pattern Analysis and Machine Intelligence, pp. 603-619, May, 2002.
6. Hafner, J., Sawhney, H. S., Equitz, W., Flickner, M., Niblack, W.: "Efficient color histogram indexing for quadratic form distance functions," IEEE Trans. Pattern Anal. Machine Intell., vol. 17, pp. 729–736, July 1995.
7. Bursuc, A., Zaharia, T., Prêteux, F.: "Detection of Multiple Instances of Video Objects," In Proc. 7th ACM/IEEE International Conference on Signal Image Technology and Internet Based Systems (Dijon, France, November 28 – December 1, 2011)
8. Kirkpatrick, S., Gelatt, C.D., Vechi, M.P.: "Optimization by simulated annealing", Science 220 (1983).