IRIM at TRECVID 2012: Semantic Indexing and Instance Search

Nicolas Ballas[1], Benjamin Labbé[1], Aymen Shabou[1], Hervé Le Borgne[1], Philippe Gosselin[2], Miriam Redi[3], Bernard Mérialdo[3], Hervé Jégou[4], Jonathan Delhumeau[4], Rémi Vieux[5], Boris Mansencal[5], Jenny Benois-Pineau[5], Stéphane Ayache[6], Abdelkader Haadi[7], Bahjat Safadi[7], Franck Thollard[7], Nadia Derbas[7], Georges Quénot[7], Hervé Bredin[8], Matthieu Cord[9], Boyang Gao[10], Chao Zhu[10], Yuxing tang[10], Emmanuel Dellandreav[10], Charles-Edmond Bichot[10], Liming Chen[10], Alexandre Benoît[11], Patrick Lambert[11], Tiberius Strat[11], Joseph Razik[12], Sébastion Paris[12], Hervé Glotin[12,13], Tran Ngoc Trung[14], Dijana Petrovska[14], Géerard Chollet[15], Andrei Stoian[16], and Michel Crucianu[16]

[1]CEA, LIST, Laboratory of Vision and Content Engineering, Gif-sur-Yvettes, France.
[2]ETIS UMR 8051, ENSEA / Université Cergy-Pontoise / CNRS, Cergy-Pontoise Cedex, F-95014 France
[3]EURECOM, Sophia Antipolis, 2229 route des crêtes, Sophia-Antipolis, France
[4]INRIA Rennes / IRISA UMR 6074 / TEXMEX project-team / 35042 Rennes Cedex
[5]LABRI UMR 5800, Université Bordeaux 1 / Université Bordeaux 2 / CNRS / ENSEIRB, Talence Cedex, France
[6]LIF UMR 6166, CNRS / Université de la Méditerranée / Université de Provence, F-13288 Marseille Cedex 9, France
[7]UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble INP / CNRS, LIG UMR 5217, Grenoble, F-38041, France
[8]Spoken Language Processing Group, LIMSI-CNRS, BP 133, 91403 Orsay Cedex, France
[9]LIP6 UMR 7606, UPMC - Sorbonne Universités / CNRS, Paris, F-75005 France
[10]LIRIS, UMR 5205 CNRS / INSA de Lyon / Université Lyon 1 / Université Lyon 2 / École Centrale de Lyon, France
[11]LISTIC, Domaine Universitaire, BP 80439, 74944 Annecy le vieux Cedex, France
[12]Dyni Team, LSIS UMR CNRS 7296 & Université Sud Toulon-Var, BP20132-83957 La Garde CEDEX-France
[13]Institut Iniversitaire de France, 103, bd Saint-Michel, 75005 Paris, France
[14]Institut Mines-Télécom, Télécom SudParis, 9 rue Charles Fourier, 91011 Evry Cedex, France
[15]Télécom ParisTech, 46 rue Barrault F-75634 Paris Cedex 13, France
[16]CEDRIC, Conservatoire National des Arts et Métiers, Paris, France

Abstract

The IRIM group is a consortium of French teams working on Multimedia Indexing and Retrieval. This paper describes its participation to the TRECVID 2012 semantic indexing and instance search tasks. For the semantic indexing task, our approach uses a six-stages processing pipelines for computing scores for the likelihood of a video shot to contain a target concept. These scores are then used for producing a ranked list of images or shots that are the most likely to contain the target concept. The pipeline is composed of the following steps: descriptor extraction, descriptor optimization, classification, fusion of descriptor variants, higher-level fusion, and re-ranking. We evaluated a number of different descriptors and tried different fusion strategies. The best IRIM run has a Mean Inferred Average Precision of 0.2378, which ranked us 4th out of 16 participants.

For the instance search task, our approach uses two steps. First individual methods of participants are used to compute similrity between an example image of instance and keyframes of a video clip. Then a two-step fusion method is used to combine these individual results and obtain a score for the likelihood of an instance to appear in a video clip. These scores are used to obtain a ranked list of clips the most likely to contain the queried instance. The best IRIM run has a MAP of 0.1192, which ranked us 29th on 79 fully automatic runs.

1 Semantic Indexing

1.1 Introduction

The TRECVID 2012 semantic indexing task is described in the TRECVID 2012 overview paper [1, 2]. Automatic assignment of semantic tags representing high-level features or concepts to video segments can be fundamental technology for filtering, categorization, browsing, search, and other video exploitation. New technical issues to be addressed include methods needed/possible as collection size and diversity increase, when the number of features increases, and when features are related by an ontology. The task is defined as follows: "Given the test collection, master shot reference, and concept/feature definitions, return for each feature a list of at most 2000 shot IDs from the test collection ranked according to the possibility of detecting the feature." 346 concepts have been selected for the TRECVID 2012 semantic indexing task. Annotations on the development part of the collections were provided in the context of a collaborative annotation effort [16].

Fifteen French groups (CEA-LIST, CNAM, ETIS, EURECOM, INRIA, LABRI, LIF, LIG, LIMSI, LIP6, LIRIS, LISTIC, LSIS, Mines Télécom and Télécom ParisTech) collaborated to participate to the TRECVID 2012 semantic indexing task.

The IRIM approach uses a six-stages processing pipeline that compute scores reflecting the likelihood of a video shot to contain a target concept. These scores are then used for producing a ranked list of images or shots that are the most likely to contain the target concept. The pipeline is composed of the following steps:

1. Descriptor extraction. A variety of audio, image and motion descriptors have been produced by the participants (section 1.2).

2. Descriptor optimization. A post-processing of the descriptors allows to simultaneaously improve their performance and to reduce their size (section 1.3).

3. Classification. Two types of classifiers are used as well as their fusion (section 1.4).

4. Fusion of descriptor variants. We fuse here variations of the same descriptor, e.g. bag of word histograms with different sizes or associated to different image decompositions (section 1.6).

5. Higher-level fusion. We fuse here descriptors of different types, e.g. color, texture, interest points, motion (section 1.7).

6. Re-ranking. We post-process here the scores using the fact that videos statistically have an homogeneous content, at least locally (section 1.8).

This approach is quite similar to the one used by the IRIM group last year [15]. The main novelties are the inclusion of new descriptors, by some improvements in the pre-processins step and iprovements in the automatic fusion methods.

1.2 Descriptors

Thirteen IRIM participants (CEA-LIST, ETIS/LIP6, EURECOM, GIPSA, INRIA, LABRI, LIF, LIG, LSIS, LIRIS, Mines Télécom and Télécom ParisTech) provided a total of 128 descriptors, including variants of a same descriptors. These descriptors do not cover all types and variants but they include a significant number of different approaches including state of the art ones and more exploratory ones. The relative performance of these descriptors has been separately evaluated using a combination of LIG classifiers (see section 1.5). Here is a description of these descriptors (some of the descriptors used in 2011 are described in more details ih The TRECVid IRIM 2011 paper [15]):

CEALIST/tlep: texture local edge pattern [3] + color histogram ⇝ 576 dimensions.

CEALIST/bov_dsiftSC_8192: : bag of visterm[31]. Dense SIFT are extracted every 6 pixels. The codebook of size 1024 is built with K-means. The bag are generated with soft coding and max pooling. The final signature result from a three levels spatial pyramid $\rightsquigarrow 1024 \times (1+2\times 2+3\times 1) = 8192$ dimensions: see [17] for details.

ETIS/global_<feature>[<type>]x<size>: (concatenated) histogram features[4], where:

<feature> is chosen among lab and qw:

- **lab:** CIE L*a*b* colors
- **qw:** quaternionic wavelets (3 scales, 3 orientations)

<type> can be:

- **m1x1:** histogram computed on the whole image
- **m1x3:** histogram for 3 vertical parts
- **m2x2:** histogram on 4 image parts

<size> is the dictionary size, sometimes different from the final feature vector dimension.

For instance, with <type>=m1x3 and <size>=32, the final feature vector has $3 \times 32 = 96$ dimensions.

ETIS/vlat_<desc type>_dict<dict size>_<size>: compact Vectors of Locally Aggregated Tensors (VLAT [6]). <desc type> = low-level descriptors, for instance hog6s8 = dense histograms of gradient every 6 pixels, 88 pixels cells. <dict size> = size of the low-level descriptors dictionary. <size> = size of feature for one frame. Note: these features can be truncated. These features must be normalized to be efficient (e.g. L_2 unit length).

EUR/sm462: The Saliency Moments (SM) feature [5] is a holistic descriptor that embeds some locally-parsed information, namely the shape of the salient region, in a holistic representation of the scene, structurally similar to [7]. More details in [15].

INRIA/dense_sift_<k>: Bag of SIFT computed by INRIA with k-bin histograms ⇝ k dimensions with k = 128, 256, 512, 1024, 2048, 4096 and 8192.

LABRI/faceTracks: OpenCV+median temporal filtering, assembled in tracks, projected on keyframe with temporal and spatial weighting and quantized on image divided in 16 × 16 blocks ⇝ 256 dimensions.

LIF/percepts_<x>_<y>_1_15: 15 mid-level concepts detection scores computed on x × y grid blocks in each key frames with (x,y) = (20,13), (16,6), (5,3), (2,2) and (1,1), ⇝ 15 × x × y dimensions.

LIG/opp_sift_<method>[_unc]_1000: bag of word, opponent sift, generated using Koen Van de Sande's software[8] ⇝ 1000 dimensions (384 dimensions per detected point before clustering; clustering on 535117 points coming from 1000 randomly chosen images). **<method>** method is related to the way by which SIFT points are selected: **har** corresponds to a filtering via a Harris-Laplace detector and **dense** corresponds to a dense sampling; the versions with **_unc** correspond to the same with fuzziness introduced in the histogram computation.

LIG/stip_<method>_<k>: bag of word, STIP local descriptor, generated using Ivan Laptev's software [9], **<method>** may be either histograms of oriented (spatial) gradient (**hog**) or histograms of optical flow (**hof**), ⇝ k dimensions with k = 256 or 1000.

LIG/concepts: detection scores on the 346 TRECVID 2011 SIN concepts using the best available fusion with the other descriptors, ⇝ 346 dimensions.

LIRIS/OCLPB_DS_4096: Dense sampling OCLBP [33] bag-of-words descriptor with 4096 k-means clusters. We extract orthogonal combination of local binary pattern (OCLBP) to reduce original LBP histogram size and at the same time preserve information on all neighboring pixels. Instead of encoding local patterns on 8 neighbors, we perform encoding on two sets of 4 orthogonal neighbors, resulting two independent codes. Concatenating and accumulating two codes leads to a final 32 dimensional LBP histogram, compared with original 256 dimensions. The 4096 bag-of-words descriptors are finally generated by the pre-trained dictionary.

LIRIS/MFCC_4096: MFCC bag-of-words descriptor with 4096 k-means clusters. To reserves video's sequential information, we keep 2 seconds audio wave around the key frame, 1 second before and after. 39 dimensional MFCC descriptors with delta and delta delta are extracted with 20ms window length and 10ms window shift. The 4096 bag-of-words descriptors are finally generated by the pre-trained dictionary.

LISTIC/SIFT_*: Bio-inspired retinal preprocessing strategies (retinal model from [10]) applied before extracting *Bag of Words* of *Opponent SIFT* features. Features extracted on a 6 pixel sampling dense grid. K-means clusters 1024 or 2048 visual words. L2 distance is used for matching. The proposed descriptors are similar to those from [11]. We highlight the following: *SIFT_L2_retina_k*: at keyframe level only, OppSIFT features are extracted on the enhanced details channel of the retina. *SIFT_L2_retinaMasking_k*: an interval of 30 frames centered around the keyframe is considered. The retinal transient/motion output channel is used to select blob-shaped areas containing potential areas of interest and motion in each frame. OppSIFT features are extracted within these Blobs, from all of the 30 frames, from the retinal detail channel. *SIFT_L2_MultiChannels_retinaMasking_k*: similar to the previous descriptor. Additionally, for each feature, its descriptor is concatenated with a SIFT descriptor extracted from the retina transient channel frame, at the same location allowing motion data to be considered.

LSIS/mlhmslbp_spyr_<k>: three kinds of parameters based on a Multi-Level Histogram of Multi-Scale features including spatial pyramid technique (MLHMS) [12]. More details in [15].

MTPT/superpixel_color_sift_k1064: this visual feature is extracted based on superpixel segmentation [32]; it is an histogram combination of color and texture over superpixels of a given image; these histograms are computed basically using trained codebooks as bag-of-words.

1.3 Descriptor optimization

The descriptor optimization consists of two steps: power transformation and principal component analysis (PCA) reduction [16].

Power transformation: The goal of the power transformation is to normalize the distributions of the values, especially in the case of histogram components. It simply consists in applying an $x \leftarrow x^{\alpha}$ ($x \leftarrow -(-x)^{\alpha}$ if $x < 0$) tranformation on all components individually. The optimal value of α can be optimized by cross-validation and is often close to 0.5 for histogram-based descriptors.

Principal component analysis: The goal of PCA reduction is both to reduce the size (number of dimensions) of the descriptors and to improve performance by removing noisy components.

The optimization of the value of the α coefficient and of the number of components kept in the PCA reduction is optimized by two-fold cross-validation within the development set. In practice, it is done with the LIG_KNNB classifier only (see section 1.4), since it is much faster when a large number of concepts (346 here) has to be considered and since it involves a large number of combinations to be evaluated. Trials with a restricted number of varied descriptors indicated that the optimal values for the kNN based classifier are close to the ones for the multi-SVM based one. Moreover, the overall performance is not very sensitive to the precise values for these hyper-parameters.

1.4 Classification

The LIG participant ran two types of classifiers on the contributed descriptors as well as their combination. CEA LIST also run a classifier on its descriptors.

LIG_KNNB: The first classifier is kNN-based. It is directly designed for simultaneously classifying multiple concepts with a single nearest neighbor search. A score is computed for each concept and each test sample as a linear combinations of 1's for positive training samples and of 0's for negative training samples with weights chosen as a decreasing function of the distance between the test sample and the reference sample. As the nearest neighbor search is done only once for all concepts, this classifier is quite fast for the classification of a large number of concepts. It usually gives lower classification rates than the SVM-based one but is much faster.

LIG_MSVM: The second one is based on a multiple learner approach with SVMs. The multiple learner approach is well suited for the imbalanced data set problem [13], which is the typical case in the TRECVID SIN task in which the ratio between the numbers of negative and positive training sample is generally higher than 100:1.

CEALIST_LSVM: The third one is a linear SVM classifier and was applied by CEALIST to its own high-dimensional descriptors.

LIG_ALLC: Fusion between all available classifiers. The fusion is simply done by averaging the classification scores produced by the two classifiers. Their output is naturally (or by construction) normalized in the the [0:1] range. kNN computation is done using the KNNLSB package [14]. Even though the LIG_MSVM classifier is often significantly better than the LIG_KNNB one, the fusion is most often even better, probably because they are very different in term of information type capture.

1.5 Evaluation of classifier-descriptors combinations

We evaluated a number of image descriptors for the indexing of the 346 TRECVID 2011 concepts. This has been done with two-fold cross-validation within the development set. We used the annotations provided by the TRECVID 2012 collaborative annotation organized by LIG and LIF [18]. The performance is measured by the inferred Mean Average Precision (MAP) computed on the 346 concepts. Results are presented for the two classifiers used, as well as for their fusion. Results are presented only for the best combinations of the descriptor optimization hyper-parameters.

Table 1 shows the two-fold cross-validation performance (trec_eval MAP) within the development set and the performance (sample_eval MAP) on the test set for most the descriptors (some variants are skipped) with the LIG_ALLC classifier combination; dim is the original number of dimensions of the descriptor vector, exp is the optimal value of the α coefficient, Pdim is the number of dimensions of the descriptor vector kept after PCA reduction.

At the time of the writing, not all descriptor × classifier combinations were computed. For some descriptors, only the scores from the KNN classifier was available and are displayed in the table (KNN only) and/or som are not available on the test set. Some more were not available at the time of run submissions and not all of them were included in the fusion or some less performing versions were used.

1.6 Performance improvement by fusion of descriptor variants and classifier variants

As in previous years, we started by fusing classification scores from different variants of a same descriptor and from different classifiers of a same variant of a same descriptor. This is done as first levels of hierarchical late fusion, the last ones being done using dedicated methods as described in section 1.7. Three levels are considered when applicable: fusions of different classifiers of a same variant of a same descriptor, fusion of different variants of a same descriptor according to a dictionary size, and fusion of different variants of a

Table 1: Performance of the classifier and descriptor combinations

Descriptor	dim	exp	Pdim	MAP dev	MAP test
CEALIST/tlep	576	0.350	128	0.1118	0.1064
CEALIST/bov_dsiftSC_8192	8192	0.800	256	0.1417	0.1774
CEALIST/2012_motion1000_tshot	1000	0.500	512	0.0540	
ETIS/global_labm1x1x512	512	0.350	192	0.1070	0.0991
ETIS/global_labm1x3x512	1536	0.350	256	0.1196	0.1230
ETIS/global_labm2x2x512	2048	0.350	384	0.1174	0.1166
ETIS/global_qwm1x1x512	512	0.450	192	0.0959	0.0892
ETIS/global_qwm1x3x512	1536	0.450	256	0.1114	0.1133
ETIS/global_qwm2x2x512	2048	0.450	384	0.1079	0.1036
ETIS/vlat_hog6s8_dict16	4096	0.900	512	0.1312	
ETIS/vlat_hog6s8_dict64	4096	0.900	512	0.1343	
ETIS/vlat_hog3s4-6-8-10_dict64	4096	0.900	512	0.1530	0.1984
EUR/sm462	462	0.150	128	0.1148	0.1183
INRIA/dense_sift_k128	128	0.400	96	0.1002	0.1025
INRIA/dense_sift_k256	256	0.400	128	0.1076	0.1190
INRIA/dense_sift_k512	512	0.450	181	0.1174	0.1361
INRIA/dense_sift_k1024	1024	0.450	256	0.1241	0.1493
INRIA/dense_sift_k2048	2048	0.450	320	0.1307	0.1623
INRIA/dense_sift_k4096	4096	0.450	400	0.1354	0.1720
INRIA/dense_sift_k8192	4096	0.450	512	0.1243	0.1801
INRIA/vlad_10240	10240	0.500	640	0.1589	
INRIA/vlad_20480	20480	0.500	640	0.1623	
INRIA/vlad_32768	32768	0.500	640	0.1448	
LABRI/faceTracks16x16	256	0.400	192	0.0157	0.0191
LABRI/faceTracks16x16_B	256	0.400	192	0.0159	
LIF/percepts_1_1_1_15 (KNN only)	15	0.400	15	0.0729	0.0432
LIF/percepts_2_2_1_15 (KNN only)	60	0.600	50	0.0873	0.0680
LIF/percepts_5_3_1_15 (KNN only)	225	0.700	150	0.0968	0.0881
LIF/percepts_10_6_1_15 (KNN only)	900	0.450	250	0.1002	0.0902
LIF/percepts_20_13_1_15 (KNN only)	3900	0.400	300	0.1017	0.0907
LIG/opp_sift_har_1000	1000	0.500	250	0.1131	0.1274
LIG/opp_sift_dense_1000	1000	0.400	250	0.1260	0.1410
LIG/opp_sift_har_unc_1000	1000	0.500	250	0.1247	0.1490
LIG/opp_sift_dense_unc_1000	1000	0.300	250	0.1303	0.1465
LIG/stip_hof_256	256	0.400	128	0.0555	
LIG/stip_hog_256	256	0.500	128	0.0783	
LIG/stip_hof_1000	1000	0.500	256	0.0617	
LIG/stip_hog_1000	1000	0.500	256	0.0819	
LIG/faces (KNN only)	15	1.000	15	0.0071	0.0176
LIG/concepts (KNN only)	346	1.800	192	0.1361	0.1928
LIRIS/MFCC_4096 (KNN only)	4096	0.600	512	0.0446	0.0216
LIRIS/OCLBP_DS_4096 (KNN only)	4096	0.600	512	0.0368	0.0289
LISTIC/SIFT_L2_1024 (KNN only)	1024	0.500	256	0.0757	0.0557
LISTIC/SIFT_L2_2048 (KNN only)	2048	0.500	384	0.0756	0.0595
LISTIC/SIFT_L2_BOR_2048bows_2048 (KNN only)	2048	0.400	384	0.0473	0.0283
LISTIC/SIFT_L2_MultiChannels_retinaMasking_1024 (KNN only)	1024	0.700	256	0.0753	0.0457
LISTIC/SIFT_L2_retina_1024 (KNN only)	1024	0.700	256	0.0809	0.0583
LISTIC/SIFT_L2_retinaMasking_1024 (KNN only)	1024	0.600	256	0.0712	0.0454
LSIS/mlhmslbp_spyr_10240	10240	0.500	768	0.1380	
LSIS/mlhmslbp_spyr_26624	26624	0.700	768	0.1322	
MTPT/superpixel_color_sift_k1064	1064	0.500	256	0.1271	

same descriptor according to a pyramidal decomposition. While the last levels of fusion attempt to improve the overall performance by fusing information of different types (e.g;. color, texture, percepts or SIFT), the first fusion levels attempt to improve the robustness of the classification from a given type. More details on this approach can be found in the previous TRECVid IRIM papers [19, 15].

1.7 Final fusion

Two IRIM participants (LISTIC and LIMSI) worked on the automatic fusion of the classification results. The fusion started with the original classification scores and/or with the results of previous fusions of descriptor variants and/or classifier variants as described in the previous section. A comparison of the LISTIC and LIMSI automatic fusion methods, along with another fusion method tried in the context of the Quaero group using some of the same classification results, and an arithmetic mean and the best attribute per concept, is given in [30].

1.7.1 LISTIC fusion

We will call an "attribute" the set of scores obtained by applying any of the KNNB, MSVM or ALLC classifiers on a descriptor. The inputs for this fusion approach are the KNNB, MSVM and ALLC scores obtained from various descriptors. We normally use the ALLC scores, but if they are not available for a particular descriptor, the KNNB and/or MSVM scores for that descriptor are used instead.

As explained in [30], this fusion approach treats each semantic concept independently. It automatically filters out irrelevant attributes for that concept, then it automatically groups highly-correlated attributes in an iterative manner. The method does not take into account the type of descriptor or the type of classifier for generating groupings, instead it groups based on the correlation of output scores. Being an automatic fusion method, it can easily be extended to large sets of available attributes, without having to manually specify groupings. The method consists in the following steps:

1. determine the individual relevance of each attribute for the target concept. The relevance is taken as the average precision α for that concept on the training dataset, normalized by the proportion of true positives in the training dataset.

2. retain only attributes with a relevance higher than 1 (better than random classification). Additionally, the attributes must have at least 1/8th of the relevance of the best one, so as not to "pollute" the good attributes with bad ones.

3. Some of the retained attributes are highly correlated, so we look for the pair with the maximum correlation and fuse it into a single attribute through arithmetic mean. We keep track of how many original attributes have already been fused into each of the members of the pair, in order to balance the members of the pair correctly. After replacing the pair with the new attribute, we update the correlation between the resulting attribute and the remaining ones.

4. The previous step is repeated many times, until a sufficiently correlated pair can no longer be found. This has a dimensionality-reduction effect and also helps to reduce the classification "noise".

The correlation measure used is the correlation coefficient of the raw (unnormalized) classification scores. We consider a pair of attributes as correlated if *(a)* the correlation coefficient for all video shots is at least 0.75, to ensure that the two attributes give similar information on a global scale, and *(b)*, the correlation between the scores for just the positive shots must be at least 0.65, to ensure that the positives tend to be classified in the same way. We add the second constraint because in TRECVID, most of the target concepts have very few positives, and otherwise, the classification scores on the negatives would dominate the correlation. Now, the resulting attributes are again filtered based on their average precision on the training set, using the same criteria as before. Afterwards, the individual relevances of each remaining attribute are used as weights for a weighted arithmetic mean, thus obtaining the final classification score. In the end, because all the previous steps consisted of selections and averaging, without any normalization operations, this approach is in fact a weighted arithmetic mean, with a more elaborate way of choosing the weights.

This method has given the TRECVid Semantic Indexing Full Task runs F_A_IRIM1_1 (which also uses a temporal re-ranking of video shots) and F_A_IRIM3_3 (without temporal re-ranking).

1.7.2 LIMSI community-driven hierarchical fusion

Let $\mathtt{K}$ be the number of available classifiers and $\mathtt{N}$ the number of video shots. Each classifier $k \in \{1 \ldots \mathtt{K}\}$ provides scores $\mathbf{x}_k = [x_{k1}, \ldots, x_{k\mathtt{N}}]$ indicating the likelihood for each shot $n \in \{1 \ldots \mathtt{N}\}$ to contain the requested concept. The objective is to find a combination function $\mathbf{f}$ so that the resulting classifier $\mathbf{x} = \mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_\mathtt{K})$ is better than any of its components, and as good as possible.

Graph of classifiers Let us denote ρ_{ij} the Spearman rank correlation coefficient of two classifiers i and j. We then define the agreement A_{ij} between two classifiers i and j as $A_{ij} = \max(0, \rho_{ij})$.

A complete undirected graph $\mathcal{G}$ is constructed with one node per classifier. Each pair of classifiers (i, j) is connected by an undirected edge, whose weight is directly proportional to A_{ij}. Based on this graph $\mathcal{G}$, classifiers can be automatically grouped into communities using the so-called Louvain approach proposed by *Blondel et al.* [21].

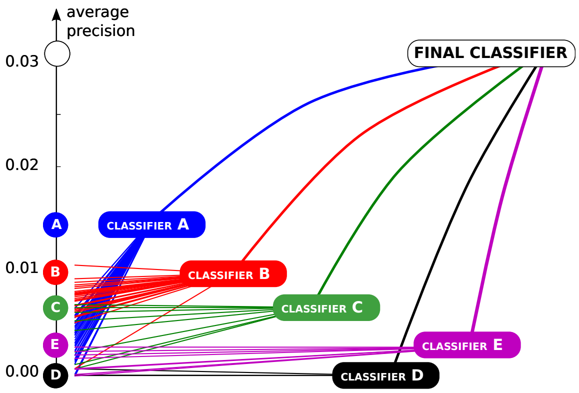

Figure 1: Community-driven hierarchical fusion

F_A_IRIM4_4: Hierarchical fusion It can be divided into three consecutive steps:

Step 1: community detection. Classifiers are automatically grouped into C communities using the *Louvain* method described above;

Step 2: intra-community fusion. Classifiers from each community are combined by simple sum of normalized scores, in order to obtain one new classifier per community (classifiers A to E in Figure 1): $\mathbf{x}_c = \sum_{k=1}^{k=\mathtt{K}} \delta_c(k) \widehat{\mathbf{x}_k}$ with $\delta_c(k) = 1$ if classifier k is part of community c (and 0 otherwise);

Step 3: inter-community fusion. Those new classifiers are then combined using weighted sum fusion of normalized scores: $\mathbf{x} = \sum_{c=1}^{c=\mathtt{C}} \alpha_c \widehat{\mathbf{x}_c}$. To this end, the performance α_c (average precision) of each of these new *community classifiers* needs to be estimated using a development set.

F_A_IRIM2_2: re-ranked version of F_A_IRIM4_4 using the method described in section 1.8.

1.8 Re-ranking

Video retrieval can be done by ranking the samples according to their probability scores that were predicted by classifiers. It is often possible to improve the retrieval performance by re-ranking the samples. *Safadi and Quénot* in [22] propose a re-ranking method that improves the performance of semantic video indexing and retrieval, by re-evaluating the scores of the shots by the homogeneity and the nature of the video they belong to. Compared to previous works, the proposed method provides a framework for the re-ranking via the homogeneous distribution of video shots content in a temporal sequence. The experimental results showed that the proposed re-ranking method was able to improve the system performance by about 18% in average on the TRECVID 2010 semantic indexing task, videos collection with homogeneous contents. For TRECVID 2008, in the case of collections of videos with non-homogeneous contents, the system performance was improved by about 11-13%.

1.9 Evaluation of the submitted runs

IRIM officially submitted the four F_A_IRIM1_1 to F_A_IRIM4_4 runs that are described in section 1.7. Table 2 presents the result obtained by the four runs submitted as well as the best and media runs for comparison. The best IRIM run corresponds to a rank of 4 within the 16 participants to the TRECVID 2012 full SIN task.

Table 2: InfMAP result and rank on the test set for all the 46 TRECVID 2012 evaluated concepts (full task).

System/run	MAP	rank
Best run	0.3210	1
F_A_IRIM1_1	0.2379	12
F_A_IRIM3_3	0.2278	13
F_A_IRIM2_2	0.2248	16
F_A_IRIM4_4	0.2153	18
Median run	0.1944	26

2 Instance Search

Given visual examples of entities of limited number of types: person, character, object or location, Instance Search (INS) task [2] consists in finding segments of videos in the data set which contain instances of these entities. Each instance is represented by a few example images. Hence if we can consider the set of video clips as a visual database, the problem consists in retrieval of each instance in this database.

2.1 Global approach

To represent the clips we extract several keyframes of each individual video clip. For a given instance, we use each example image, from the available set, as a query image. We compute a similarity between this query example image and the keyframes of all video clips. We then produce a intermediary result where we have the similarity $S_{e,i,k,c}$ between each example image (e) of each instance (i) and each keyframe (k) of each video clip (c). We then have to fuse these intermediary results to obtain a final result that is similarity $S_{i,c}$ between each instance (i) and each clip (c) Within the IRIM consortium, several methods of four members (CEA, CNAM, LaBRI, LISTIC) were tested and their results fused.

2.2 Members methods

CEA_Markrs The Markrs methods follows the well-known framework of keypoint matching described in [34, 35]. Here, the SURF scheme [24] was used for keypoints detection and descriptors computations. Then SURF description is quantized into integer values in $[0, 255]$, leading to a compact description for each keypoint in less than 80 bytes.

Two filtering steps are added to reduce bad keypoints matches. First, matches must pass the test of relative nearest-neighbors proposed by D. Lowe in [34]. The second step selects within the previous results those that provides a similar geometrical configuration of keypoints in the query-candidate couple of images. We consider only simple similarities, and not complete homographies, that are faster to compute. The final result list is composed of images having more than p keypoints fitting the geometrical model ($p \geq 5$).

CEA_Snow The Snow method is based on a well-known color histogram for global image description. The color histogram counts the occurences of 162 shades in the HSV color space. The similarity between two images is measured as the inverse of a *dLog* distance between two histograms, defined as: [36]:

$$dLog(q, d) = \sum_{i=0}^{i<M} |f(q[i]) - f(d[i])| \tag{1}$$

$$f(x) = \begin{cases} 0, & \text{if} \quad x = 0 < \alpha \\ 1, & \text{if} \quad 0 < x \leq 1 \\ \lceil \log_2 x \rceil + 1 & \text{otherwise} \end{cases} \tag{2}$$

Where q and d are two histograms with M bins and $\lceil . \rceil$ is the ceilling function ().

These methods were also used in individual CEA LIST submission[17].

CNAM This method uses inverted lists for pairs of visual words. The frames of the videos in the database are described with local features, then visual words are obtained using a vocabulary tree and an inverted list is created for every pair of visual words co-occurring in at least one frame. The inverted lists include geometric information relating the scales of the two features. Every query is described in the same way and retrieval is performed by accessing the selected inverted lists. This solution was inspired by the method put forward in [37] that used triplets of Harris features (and associated geometric information) for fast content-based video copy detection. Since the TRECVID INS data shows scale and viewpoint variations, we employed instead SURF features. Individual SURF features include data regarding scale and orientation, which allows to use pairs of features rather than triplets, with an expected positive impact on recall, while keeping geometric information to improve prcision. Retrieved frames are ranked according to a similarity measure that takes into account the number of pairs from the query that are present in the frame, the similarity of corresponding individual SURF features (based on the path in the vocabulary tree), the ratios keypoints scale to distance between keypoints, and the relative orientations of the keypoints. The use of pairs of keypoints together with simple geometric information supports other possibilities that were not fully explored in the current implementation. Retrieval is indeed very fast: even when the database is composed of all the frames (and not only of the keyframes), less than 15 minutes are required for obtaining all the results to a query.

LaBRI We used the baseline Bag-Of-Visual-Words (BOVW) model based on interest point descriptor[23]. It is used both for object and for frame signature construction. The descriptor used is SURF[24]. The unsupervised clustering K-means++ with a large number of clusters (16K), with the L2 distance, was used for dictionary computation. The complement of histogram intersection was used to compare signatures. These methods will be referenced as *LaBRI_Bow_Obj* and *LaBRI_Bow_Frm*.

For frame-based queries, we also used BOVW with affine deformation of object mask, described in [38]. This method uses a correlation kernel, deforming object mask according to Pan/Tilt/Zoom affine model. The correlation was done by full search in the affine parameter space. Pan and Tilt parameters were chosen in such a way that query instance mask overlaps the DB frame at least two third of its area. The Zoom factor were chosen from the set 0.25, 0.5, 1, 2. This method is obviously more computationally demanding than the traditional BOVW. Indeed, signatures can not be computed in a processing step for all the images of the DB, but have to be computed in image area overlapped by image mask. This method will be referenced as *LaBRI_Bow_Obj_Aff*.

A second approach is a Bag-Of-Regions (BOR) model, as proposed by Vieux et al. in [25], that extends the traditional notion of BOVW vocabulary to region based

Name	Formula
CombMAX	MAX(individual similarities)
CombSUM	SUM(individual similarities)
CombANZ	CombSUM / Number of non zero similarities
CombMNZ	CombSUM * Number of non zero similarities

Table 3: Definitions of different combination operators

descriptors. Regions in image plane are obtained by segmenting images by Felzenszwalb and Huttenlocher method [26]. The HSV histogram was computed as region feature. We set a uniform quantizing parameters in order to limit the feature size to approximately 100 bins (45+32+32) and to privilege the finest encoding of Hue component. We used the incremental clustering algorithm described in [28] and modified in [25], with 2K clusters and L2 distance. Region-based approach was deployed only for frame signatures construction. To compare signatures, we used the L1 distance. This method will be referenced as *LaBRI_Bor_Frm*.

For development and test sets, we computed their proper codebooks as we are not granted that the two sets have the same distribution in proposed description spaces.

LISTIC The two methods *LISTIC_SIFT_L2_k* and *LISTIC_SIFT_L2_retina_k* used in the SIN task were also used in the INS task, with k=1024 and complement of histogram intersection for signatures comparison. Codebook computed on dataset of SIN task was used.

2.3 Fusion

Each described members method was used to produce intermediary results. Thus for each method (m) , we have a similarity $S_{m,e,i,k,c}$ between each example image (e) of each instance (i) and each keyframe (k) of each video clip (c). We have to fuse these similarities to obtain a similarity for an instance (i) and a clip (c).

We used a limited number combination operators: CombMAX, CombSUM, CombANZ, CombMNZ[39], defined in table 3.

We have tested two late fusion schemes. A truly late fusion scheme considers all the similarities $S_{m,e,i,k,c}$ at once. In a two-step late fusion scheme, we first merge the results for a given method (m), and then globally. Besides, weights can be used to give an asymmetric importance to the various intermediary results. Here, all intermediary results have been previously normalized. These two fusion schemes are described by the equations 3 and 4, where α_m and β_m are weights that sum to 1.

$$S_{i,c} = Comb_1(\alpha_m * S_{m,e,i,k,c}) \tag{3}$$

Method	Best operator
CEA_marks	CombSUM[S]
CEA_snow	CombMAX[R]
CNAM	CombMAX[S]
LaBRI_Bow_Obj	CombANZ[S]
LaBRI_Bow_Frm	CombANZ[S]
LaBRI_Bow_Obj_Aff	CombMAX[S]
LaBRI_Bor_Frm	CombANZ[S]
LISTIC_SIFT_L2_1024	CombMAX[S]
LISTIC_SIFT_L2_retina_1024	CombANZ[S]

Table 4: Best combination operator with similarity type used for each individual methods

$$\begin{aligned} S_{i,c} &= Comb_2(\beta_m * S_{e,i,k,c}) \\ with\ S_{e,i,k,c} &= Comb_m(\alpha_m * S_{m,e,i,k,c}) \end{aligned} \tag{4}$$

A Combination operator will be noted $Comb[S]$ if applied to score, and $Comb[R]$ if applied to rank. We have tested several combination operators with these two fusion schemes, applied both to score and to rank. We have also tested with a limited combination of weights. The best results were obtained with the two-step fusion scheme. Moreover, as performance of various methods is not homogeneous, we tried to find the best combination operator and the similarity to use for each individual method, both for 2010 and 2011 queries and datasets. Theses choices are presented in table 4.

We tested a limited number of combinations of weights. Finally, we used three decreasing functions. For N combined intermediary results, we used the formulae in (5), (6), (7) for the i-th intermediary result, ranked from left to right, within the fusion step.

$$\beta 1(i) = 1 - i/N \tag{5}$$

$$\beta 2(i) = 1/(1.4^i) \tag{6}$$

$$\beta 3(i) = 1/(3^i) \tag{7}$$

Various experiments on 2010 and 2011 datasets showed that CEA_marks indivudually was giving the best results compared to all other methods. BOVW and BOR methods gave inferior, but promising results on 2011 dataset. So these results were given a bigger weight in the second fusion step, in eq. (4).

If we consider all possibilities of fusion: two-step or one-step ((3), (4)), combination operators (3), weight functions (eq. (5), (6), (7)), choice of rank (R) or similarity (S), and order the results of individual methods according to decrease of performances, we get 1536 possibilities for one run. Obviously, it was impossible to test all these combinations. Hence, we proposed 4 runs on the basis of a two-step fusion scheme, all three weight functions and a best fusion operator for individual methods. The choice of rank (R) or similarity (S) for the final fusion step was done on the basis of tests on 2010 and 2011 sets.

The submitted runs were:

$$\begin{aligned}
&Run1 = CombMAX[R](\\
&\beta1(0) * CombSUM[S](CEA\_markrs), \\
&\beta1(1) * CombANZ[S](LaBRI\_Bow\_Obj), \\
&\beta1(2) * CombMAX[S](LaBRI\_Bow\_Obj\_Aff), \\
&\beta1(3) * CombMAX[S](LISTIC\_SIFT\_L2\_1024), \\
&\beta1(4) * CombANZ[S](LaBRI\_Bow\_Frm), \\
&\beta1(5) * CombMAX[R](CEA\_snow), \\
&\beta1(6) * CombANZ[S](LaBRI\_Bor\_Frm), \\
&\beta1(7) * CombANZ[S](LISTIC\_SIFT\_L2\_retina\_1024), \\
&\beta1(8) * CombMAX[S](CNAM))
\end{aligned} \tag{8}$$

$$\begin{aligned}
&Run2 = CombSUM[R](\\
&\beta3(0) * CombSUM[S](CEA\_markrs), \\
&\beta3(1) * CombANZ[S](LaBRI\_Bow\_Obj), \\
&\beta3(2) * CombMAX[S](LaBRI\_Bow\_Obj\_Aff), \\
&\beta3(3) * CombMAX[S](LISTIC\_SIFT\_L2\_1024), \\
&\beta3(4) * CombANZ[S](LaBRI\_Bow\_Frm), \\
&\beta3(5) * CombMAX[R](CEA\_snow), \\
&\beta3(6) * CombANZ[S](LaBRI\_Bor\_Frm), \\
&\beta3(7) * CombANZ[S](LISTIC\_SIFT\_L2\_retina\_1024), \\
&\beta3(8) * CombMAX[S](CNAM))
\end{aligned} \tag{9}$$

$$\begin{aligned}
&Run3 = CombMAX[R](\\
&\quad \beta2(0) * CombSUM[S](CEA\_markrs), \\
&\quad \beta2(1) * CombMAX[S](LaBRI\_Bow\_Obj\_Aff), \\
&\quad \beta2(2) * CombANZ[S](LaBRI\_Bor\_Frm), \\
&\quad \beta2(3) * CombMAX[R](CEA\_snow))
\end{aligned} \tag{10}$$

$$\begin{aligned}
&Run4 = CombMAX[R](\\
&\quad \beta2(0) * CombSUM[S](CEA\_markrs), \\
&\quad \beta2(1) * CombANZ[S](LaBRI\_Bow\_Obj), \\
&\quad \beta2(2) * CombMAX[S](LaBRI\_Bow\_Obj\_Aff), \\
&\quad \beta2(3) * CombANZ[S](LaBRI\_Bow\_Frm), \\
&\quad \beta2(4) * CombMAX[R](CEA\_snow), \\
&\quad \beta2(5) * CombANZ[S](LaBRI\_Bor\_Frm))
\end{aligned} \tag{11}$$

2.4 Discussion

MAP and rank of our runs are presented in table 5. The two-step combination method applied to Markrs method was also submitted by CEA-LIST as an individual run, as CeaList_2. It is added for reference.

Run	MAP	Rank/79
IRIM_2	0.1192	29
IRIM_4	0.1173	31
IRIM_1	0.1171	32
IRIM_3	0.1162	33
CeaList_2	0.1135	34

Table 5: MAP and rank of 4 IRIM runs and CeaList_2 runs on test set 2012, among 79 fully automatic submitted runs

The results are very much close. Nevertheless, we can see that an appropriate fusion odes bring an improvement (run IRIM_2 vs CeaList_2). In the future while improving individual methods, we plan also to explore different fusion operators , such as described in [40].

3 Data sharing

We propose to reuse and extend the organization that has been developed over four years within the members of the IRIM project of the French ISIS national Research Group (see [15] and section 1 of this paper). It is based on a limited number of simple data formats and on a (quite) simple directory organization. It also comes with a few scripts and procedures as well as with some sections for reporting intermediate results. The supporting structure is composed of a wiki (http://mrim.imag.fr/trecvid/wiki) and a data repository (http://mrim.imag.fr/trecvid/sin12). The wiki can be accessed using the TRECVid 2012 active participant username and password and the data repository can be accessed using the TRECVid 2012 IACC collection username and password.

A general rule about the sharing of elements is that:

- any group can share any element he think could be useful to others with possibly an associated citation of a paper describing how it was produced;
- any group can use any element shared by any other group provided that this other group is properly cited in any paper presenting results obtained using the considered element,

exactly as this was the case in the previous years for the shared elements like shot segmentation, ASR transcript or collaborative annotation. Groups sharing elements get “rewarded” via citations when their elements are used.

Shared elements can be for instance: shot or key frame descriptors, classification results, fusion results. For initiating the process, most IRIM participants agreed to share their descriptors. Most classification and fusion results obtained are also shared. These should also be available on the 2013 TRECVID SIN collection. Descriptors, classification scores or fusion results from other TRECVid particpants are most welcome. See the wiki for how to proceed.

4 Acknowledgments

This work has been carried out in the context of the IRIM (Indexation et Recherche d'Information Multimédia) of the GDR-ISIS research network from CNRS.

Experiments presented in this paper were carried out using the Grid'5000 experimental testbed, being developed under the INRIA ALADDIN development action with support from CNRS, RENATER and several Universities as well as other funding bodies (see https://www.grid5000.fr).

References

[1] A. Smeaton, P. Over and W. Kraaij, Evaluation campaigns and TRECVid, In *MIR'06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pp321-330, 2006.

[2] P. Over, G. Awad, J. , B. Antonishek, M.2Michel, A. Smeaton, W. Kraaij, and G. Quénot, TRECVID 2012 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms, and Metrics In *Proceedings of the TRECVID 2012 workshop*, Gaithersburg, USA, 26-28 Nov. 2012.

[3] Y.-C. Cheng and S.-Y. Chen. Image classification using color, texture and regions. In *Image and Vision Computing,* 21:759-776, 2003.

[4] P.H. Gosselin, M. Cord, Sylvie Philipp-Foliguet. Combining visual dictionary, kernel-based similarity and learning strategy for image category retrieval. In *Computer Vision and Image Understanding,* Special Issue on Similarity Matching in Computer Vision and Multimedia. Volume 110, Issue 3, Pages 403-41, 2008.

[5] M. Redi and B. Merialdo. Saliency moments for image categorization, In *ICMR 2011, 1st ACM International Conference on Multimedia Retrieval,* April 17-20, 2011, Trento, Italy.

[6] R. Negrel, D. Picard and P.H. Gosselin. Compact Tensor Based Image Representation for Similarity Search. In *IEEE International Conference on Image Processing*, Orlando, Florida, U.S.A, September 2012.

[7] A. Oliva and A. Torralba, Modeling the shape of the scene: A holistic representation of the spatial envelope, In *International Journal of Computer Vision,* vol 42, number 3, pages 145-175, 2001.

[8] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. A comparison of color features for visual concept classification. In *ACM International Conference on Image and Video Retrieval*, pages 141–150, 2008.

[9] Ivan Laptev, On space-time interest points, *Int. J. Comput. Vision*, 64:107–123, September 2005.

[10] A. Benoit, A. Caplier, B. Durette, and J. Herault. Using Human Visual System Modeling for Bio-inspired Low Level Image Processing, In *Computer Vision and Image Understanding,* vol. 114, no. 7, pp. 758-773, 2010.

[11] S. T. Strat, A. Benoit, P. Lambert and A. Caplier, Retina Enhanced SURF Descriptors for Spatio-Temporal Concept Detection, In *Multimedia Tools ans Applications,* to appear, 2012.

[12] S. Paris, H .Glotin, Pyramidal Multi-level Features for the Robot Vision@ICPR 2010 Challenge, In *20th International Conference on Pattern Recognition,* pp.2949-2952, 2010

[13] B. Safadi, G. Quénot. Evaluations of multi-learners approaches for concepts indexing in video documents. In *RIAO,* Paris, France, April 2010.

[14] Georges Quénot. *KNNLSB: K Nearest Neighbors Linear Scan Baseline*, 2008. Software available at `http://mrim.imag.fr/georges.quenot/freesoft/knnlsb/index.html`.

[15] Delezoide et al. IRIM at TRECVID 2011: Semantic Indexing and Multimedia Instance Search, In *Proceedings of the TRECVID 2011 workshop*, Gaithersburg, USA, 5-7 Dec. 2011.

[16] Safadi et al. Quaero at TRECVID 2011: Semantic Indexing and Collaborative Annotation, In *Proceedings of the TRECVID 2012 workshop*, Gaithersburg, USA, 26-28 Nov. 2011.

[17] Nicolas Ballas, Benjamin Labbé, Hervé Le Borgne, Aymen Shabou CEA LIST at TRECVID 2012: Semantic Indexing and Instance Search, In *Proceedings of the TRECVID 2012 workshop*, Gaithersburg, USA, 26-28 Nov. 2012.

[18] Stéphane Ayache and Georges Quénot, Video Corpus Annotation using Active Learning, In 30th European Conference on Information Retrieval (ECIR'08), Glasgow, Scotland, 30th March - 3rd April, 2008.

[19] D. Gorisse et al., IRIM at TRECVID 2010: High Level Feature Extraction and Instance Search. In *TREC Video Retrieval Evaluation workshop*, Gaithersburg, MD USA, November 2010.

[20] Alice Porebski, Color texture feature selection for image classification. Application to flaw identification on decorated glasses printing by a silk-screen process. *Phd thesis,* Université Lille 1, Sciences et Technologies, Nov. 2009

[21] V. D. Blondel and J. Guillaume and R. Lambiotte and E. Lefebvre, Fast Unfolding of Community Hierarchies in Large Networks, In *Computing Research Repository,* abs/0803.0, 2008.

[22] B. Safadi, G. Quénot. Re-ranking by Local Re-scoring for Video Indexing and Retrieval, *CIKM 2011: 20th ACM Conference on Information and Knowledge Management,* Glasgow, Scotland, oct 2011.

[23] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *ICCV'03*, volume 2, pages 1470–1477, 2003.

[24] H. Bay, Herbert, T.Tuytelaars,and L. Van Gool. SURF: Speeded Up Robust Features, In *ECCV 2006*, pp 404-417, 2006.

[25] R. Vieux, J. Benois-Pineau, and J.-Ph. Domenger. Content based image retrieval using bag of region. In *MMM 2012 - The 18th International Conference on Multimedia Modeling*, 2012.

[26] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59:167–181, 2004.

[27] Emilie Dumont and Bernard Merialdo. Rushes video summarization and evaluation. *Multimedia Tools and Applications, Springer, Vol.48, No1, May 2010*, 2010.

[28] Edwin Lughofer. Extensions of vector quantization for incremental clustering. *Pattern Recognition*, 41:995–1011, 2008.

[29] http://accio.cse.wustl.edu/sg-accio/SIVAL.html.

[30] S. T. Strat, A. Benoit, H. Bredin, G. Quénot and P. Lambert. Hierarchical Late Fusion for Concept Detection in Videos. In *ECCV workshop on Information Fusion in Computer Vision for Concept Recognition,* Firenze, Italy, 13 Oct. 2012.

[31] A. Shabou and H. Le Borgne. Locality-constrained and spatially regularized coding for scene categorization, In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3618–3625, 2012.

[32] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and Kaleem. Siddiqi. Turbopixels: Fast superpixels using geometric flows. In *IEEE Trans. Pattern Anal. Mach. Intell.,* 31(12):2290-2297, December 2009.

[33] C. Zhu, C.-E. Bichot, L. Chen. Color orthogonal local binary patterns combination for image region description. In *Technical Report, LIRIS UMR5205 CNRS*, Ecole Centrale de Lyon.

[34] D.G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004

[35] R. Hartley, and A. Zisserman. Multiple view geometry in computer vision. Cambridge Univ Press, 2000

[36] R. 0. Stehling, M. A. Nascimento, and A.X. Falcão. A compact and efficient image retrieval approach based on border/interior pixel classification In *11th International Conference on Information and Knowledge Management* 2002

[37] S. Poullot, M. Crucianu, and S. Satoh Indexing local configurations of features for scalable content-based video copy detection In *1st ACM workshop on Large-Scale Multimedia Retrieval and Mining (LS-MMRM'09)* New York, NY, USA, 2009

[38] B. Mansencal, J. Benois-Pineau, R. Vieux and J.-Ph. Domenger Search of objects of interest in videos In *10th Workshop on Content-Based Multimedia Indexing* Annecy, France, 2012

[39] E. Fox and J. Shaw Combination of Multiple searches In *Proceedings of the 2nd Text Retrieval Conference* Gaithersburg, USA, 1994

[40] G. Csurka and S. Clinchant An empirical study of fusion operators for multimodal image retrieval In *10th Workshop on Content-Based Multimedia Indexing* Annecy, France, 2012