IRIM at TRECVID 2011: Semantic Indexing and Instance Search

Bertrand Delezoide[1], David Gorisse[2], Frédéric Precioso[2,10], Philippe Gosselin[2], Miriam Redi[3], Bernard Mérialdo[3], Lionel Granjon[4], Denis Pellerin[4], Michèle Rombaut[4], Hervé Jégou[5], Rémi Vieux[6], Boris Mansencal[6], Jenny Benois-Pineau[6], Stéphane Ayache[7], Bahjat Safadi[8], Franck Thollard[8], Georges Quénot[8], Hervé Bredin[9], Matthieu Cord[10], Alexandre Benoît[11], Patrick Lambert[11], Tiberius Strat[11], Joseph Razik[12], Sébastion Paris[12], and Hervé Glotin[12,13]

[1]CEA LIST, Centre de Fontenay-aux-Roses, BP 6, 92265 Fontenay-aux-Roses, 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]GIPSA-lab UMR 5216, CNRS / Grenoble INP / UJF-Grenoble 1 / U. Stendhal-Grenoble 3 / 38402 Grenoble, France
[5]INRIA Rennes / IRISA UMR 6074 / TEXMEX project-team / 35042 Rennes Cedex
[6]LABRI UMR 5800, Université Bordeaux 1 / Université Bordeaux 2 / CNRS / ENSEIRB, Talence Cedex, France
[7]LIF UMR 6166, CNRS / Université de la Méditerranée / Université de Provence, F-13288 Marseille Cedex 9, France
[8]UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble INP / CNRS, LIG UMR 5217, Grenoble, F-38041, France
[9]Spoken Language Processing Group, LIMSI-CNRS, BP 133, 91403 Orsay Cedex, France
[10]LIP6 UMR 7606, UPMC - Sorbonne Universités / CNRS, Paris, F-75005 France
[11]LISTIC, Domaine Universitaire, BP 80439, 74944 Annecy le vieux Cedex, France
[12]Dyni Team, LSIS UMR CNRS 6168 & Université Sud Toulon-Var, BP20132-83957 La Garde CEDEX-France
[13]Institut Iniversitaire de France, 103, bd Saint-Michel, 75005 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 2011 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 0.1387, which ranked us 5th out of 19 participants. For the instance search task, we we used both object based query and frame based query. We formulated the query in standard way as comparison of visual signatures either of object with parts of DB frames or as a comparison of visual signatures of query and DB frames. To produce visual signatures we also used two apporaches: the first one is the baseline Bag-Of-Visual-Words (BOVW) model based on SURF interest point descriptor; the second approach is a Bag-Of-Regions (BOR) model that extends the traditional notion of BOVW vocabulary not only to keypoint-based descriptors but to region based descriptors.

1 Semantic Indexing

1.1 Introduction

The TRECVID 2011 semantic indexing task is described in the TRECVID 2011 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 2011 semantic indexing task. Annotations on the development part of the collections were provided in the context of a collaborative annotation

effort [14].

Twelve French groups (CEA-LIST, ETIS, EURECOM, GIPSA, INRIA, LABRI, LIF, LIG, LIMSI, LIP6, LISTIC and LSIS) collaborated for participating to the TRECVID 2011 semantic indexing task.

The IRIM 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:

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).

1.2 Descriptors

Nine IRIM participants (CEA-LIST, ETIS/LIP6, EURECOM, GIPSA, INRIA, LABRI, LIF, LIG, and LSIS) provided a total of 48 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:

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

ETIS/global_<feature>[<type>]x<size>: (concatenated) histogram features[4, 5], where:

<**feature**> is chosen among lab and qw:

lab: CIE L*a*b* colors

qw: quaternionic wavelets (3 scales, 3 orientations)

<**type**> can be

nothing: 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.

EUR/sm462: The Saliency Moments (SM) feature [6] 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]. First, the saliency information is extracted at different resolutions using a spectral, light-weight algorithm. The signals obtained are then sampled directly in the frequency domain, using a set of Gabor wavelets. Each of these samples, called "Saliency Components", is then interpreted as a probability distribution: the components are divided into subwindows and the first three moments are extracted, namely mean, standard deviation and skewness. The resulting signature vector is a 462-dimensional descriptor that we use as input for traditional support vector machines and then combine with the contributions of the other visual features.

GIPSA/AudioSpectro[N]-b28: Spectral profile in 28 bands on a Mel scale, N: normalized $\rightsquigarrow$ 28 dimensions.

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

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 $\rightsquigarrow$ 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), $\rightsquigarrow$ 15 × x × y dimensions.

KIT/faces KIT contributed by proposing descriptors/predictions at the face level.

LIG/h3d64: normalized RGB Histogram $4 \times 4 \times 4$ $\rightsquigarrow$ 64 dimensions.

LIG/gab40: normalized Gabor transform, 8 orientations × 5 scales, ⇝ 40 dimensions.

LIG/hg104: early fusion (concatenation) of h3d64 and gab40 ⇝ 104 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.

LISTIC/SURF_retinaMasking_<k>_cross: SURF based bag of words (BOW) with k = 1024 or 4096 dimensions using a real-time retina model [10]. We consider 40 frames around each subshot keyframe. An automatic salient blobs segmentation is applied on each frame and a dense grid is considered only within these regions. SURF descriptors are captured within each frame blobs and are cumulated along the 40 frames. This allows the BOW of the subshot keyframe to be defined globally. Descriptors are extracted from the retinal foveal vision model (Parvocellular pathway). It allows light and noise robustness and enhanced SURF description. The retinal motion channel (Magnocellular pathway) is used to perform the automatic blobs segmentation. This channel allows transient blobs to be detected during the 40 frames. Such transient blobs are related to salient detailed areas during the retina model transient state (during the 20 first frames). Its also corresponds to moving areas at the retina's stable state (during the last 20 frames). Such segmentation allows spatio-temporal low level saliency areas to be detected. For BOW training, vocabulary learning is performed with Kmeans on 1008 subshots taken from 2011a and 2011b keyframes lists using 6 622 198 points.

LSIS/mlhmslbp_spyr_<k>: Three kinds of parameters based on a Multi-Level Histogram of Multi-Scale features including spatial pyramid technique (MLHMS) [11]. In each parameters extraction method, the pictures were considered as gray-scale pictures. The two first kinds of parameters are based on local binary pattern (LBP). A two levels pyramid was used with the level being the entire picture and the second level being a half in the horizontal direction and a forth in the vertical direction respectively a third and a sixth for the second kind of parameters). Moreover, an overlapping of half of the level-direction size is used. 4 levels of scaling were also computed for the LBP parameters, from 1 to 4 pixels blocks. The resulting parameter vectors are then L2-clamp normed. For the third kind of parameters, we used second order Local Derivative Pattern (LDP). We used the same kind of level, scaling and spatial pyramid than for the two preceding parameters. The dimensions of the resulting vectors are respectively 10240 and 26624 for the MLHMS-LBP parameters, and 106496 for the MLHMS-LDP parameters. For practical reasons, we were only able to use the MLHMS-LBP descriptor with 10240 dimensions.

1.3 Descriptor optimization

The descriptor optimization consists of two steps: power transformation and principal component analysis (PCA) reduction [14].

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. It is done in practice only using the LIG_KNNB classifier (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. Also, 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.

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 is generally less good than the SVM-based one but it 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 [12], which is the typical case in the TRECVID SIN task in which the ration between the numbers of negative and positive training sample is generally higher than 100:1.

LIG_ALLC: Fusion between the two available classifiers. The fusion is simply done by averaging the classification scores produced by the two classifiers. Their output is naturally or by designed normalized in the the [0:1] range. kNN computation is done using the KNNLSB package [13]. 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 and capture different things.

1.5 Evaluation of classifier-descriptors combinations

We have evaluated a number of image descriptors for the indexing of the 346 TRECVID 2011 concepts. This was done with two-fold cross-validation within the development set. We used the annotations provided by the TRECVID 2011 collaborative annotation organized by LIG and LIF [15]. 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 for the descriptor optimization hyper-parameters.

Table 1 shows the two-fold cross-validation performance (trec_eval MAP) for all the descriptors 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.

1.6 Performance improvement by fusion of descriptor variants and classifier variants

In a previous work, LIG introduced and evaluated the fusion of descriptor variants for improving the performance of concept classification. We previously tested it in the case of color histograms in which we could change the number of bins, the color space used, and the fuzziness of bin boundaries. We found that each of these parameters had an optimal value when the others are fixed and that there is also an optimal combination of them which correspond to the best classification that can be reached by a given classifier (kNN was used here) using a single descriptor of this type. We also tried late fusion of several variants of non-optimal such descriptors and found that most combinations of non-optimal descriptors have a performance which is consistently better than the individual performance of the best descriptor alone. This was the case even with a very simple fusion strategy like taking the average of the probability scores. This was also the case for hierarchical late fusion. In the considered case, this was true when fusing consecutively according to the number of bins, to the color space and to the bin fuzziness. Moreover, this was true even if some variant performed less well than others. This is particularly interesting because descriptor fusion is known to work well when descriptors capture different aspects of multimedia content (e.g. color and texture) but, here, an improvement is obtained using many variants of a single descriptor. That may be partly due to the fact that the combination of many variant reduces the noise. The gain is less than when different descriptor types are used but it is still significant.

We have then generalized the use of the fusion of descriptor variants and we evaluated it on other descriptors and on TRECVID 2010. We made the evaluation on descriptors produced by the ETIS partner of the IRIM group. ETIS has provided 3 $\times$ 4 variants of two different descriptors (see the previous section). Both these descriptors are histogram-based. They are computed with four different number of bins: 64, 128, 192 and 256; and with three image decomposition: 1x1 (full image), 1x3 (three vertical stripes) and 2x2 (2 by 2 blocks). Hierarchical fusion is done according to three levels: number of bins, "pyramidal" image decomposition and descriptor type.

We have evaluated the results obtained for fusion within a same descriptor type (fusion levels 1 and 2) and between descriptor types (fusion level 3) [16]. The fusion of the descriptor variants varies from about 5 to 10% for the first level and is of about 4% for the second level. The gain for the second level is relative to the best result for the first level so both gains are cumulated. For the third level, the gain is much higher as this could be expected because, in this case, we fuse

Table 1: Performance of the classifier and descriptor combinations

Descriptor	dim	exp	Pdim	MAP
CEALIST/tlep	576	0.350	128	0.0917
ETIS/global_lab256	256	0.350	128	0.0775
ETIS/global_labm1x3x256	768	0.350	256	0.0910
ETIS/global_labm2x2x256	1024	0.350	256	0.0872
ETIS/global_qw256	256	0.500	128	0.0718
ETIS/global_qwm1x3x256	768	0.500	256	0.0863
ETIS/global_qwm2x2x256	1024	0.500	256	0.0821
ETIS/global_lab192	192	0.350	96	0.0762
ETIS/global_labm1x3x192	576	0.350	192	0.0903
ETIS/global_labm2x2x192	768	0.350	192	0.0883
ETIS/global_qw192	192	0.450	96	0.0686
ETIS/global_qwm1x3x192	576	0.450	192	0.0841
ETIS/global_qwm2x2x192	768	0.450	192	0.0811
ETIS/global_lab128	128	0.350	96	0.0750
ETIS/global_labm1x3x128	384	0.350	192	0.0905
ETIS/global_labm2x2x128	512	0.350	192	0.0871
ETIS/global_qw128	128	0.450	96	0.0658
ETIS/global_qwm1x3x128	384	0.450	192	0.0814
ETIS/global_qwm2x2x128	512	0.450	192	0.0789
EUR/sm462	462	0.150	125	0.0798
GIPSA/AudioSpectro_b28	28	0.200	28	0.0097
GIPSA/AudioSpectroN_b28	28	0.200	28	0.0097
INRIA/dense_sift_k128	128	0.400	64	0.0903
INRIA/dense_sift_k256	256	0.400	128	0.1012
INRIA/dense_sift_k512	512	0.450	256	0.1089
INRIA/dense_sift_k1024	1024	0.450	256	0.1132
INRIA/dense_sift_k2048	2048	0.500	256	0.1170
INRIA/dense_sift_k4096	4096	0.600	362	0.1175
LABRI/faceTracks16x16	256	0.350	192	0.0135
LIF/percepts_1_1_1_15	15	0.400	15	0.0557
LIF/percepts_2_2_1_15	60	0.600	50	0.0832
LIF/percepts_5_3_1_15	225	0.700	150	0.0934
LIF/percepts_10_6_1_15	900	0.450	250	0.0927
LIF/percepts_20_13_1_15	3900	0.400	300	0.0942
LIG/h3d64	64	0.300	32	0.0665
LIG/gab40	40	0.500	30	0.0457
LIG/hg104	104	0.300	52	0.0867
LIG/opp_sift_har_1000	1000	0.450	150	0.0939
LIG/opp_sift_dense_1000	1000	0.450	200	0.1032
LIG/opp_sift_har_unc_1000	1000	0.300	200	0.0939
LIG/opp_sift_dense_unc_1000	1000	0.450	250	0.1071
LIG/stip_hof_256	256	0.450	128	0.0360
LIG/stip_hog_256	256	0.500	128	0.0550
LIG/stip_hof_1000	1000	0.400	175	0.0408
LIG/stip_hog_1000	1000	0.450	175	0.0571
LIG/concepts	346	1.750	256	0.1144
LISTIC/SURF_retinaMasking_1024_cross	1024	0.500	64	0.0468
LISTIC/SURF_retinaMasking_4096_cross	4096	0.400	64	0.0513
LSIS/mlhmslbp_spyr_10240	10240	0.750	384	0.1050

results from different information sources. The gain at level 3 is also cumulated with the gain at the lower levels.

1.7 Final fusion

Two IRIM participants (LISTIC and LIMSI) worked on the 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. Another fusion method was tried in the context of the Quaero group using some of the same classification results; it is reported in [14].

1.7.1 LISTIC fusion

A 'selection − fusion − PCA − neighborhood' approach has been proposed. It borrows ideas from [17] and is applied for a late fusion. Each concept is treated individually. As input attributes, likelihood scores of each shot to contain a concept are considered. Such scores are calculated from each low level descriptor taken individually with a KNN classifier. The fusion consists of the following steps:

1. select only the attributes that have an individual relevance of the same order of magnitude as the most relevant attribute.

2. fuse the highly correlated pairs of attributes into a single one (with an arithmetic mean), in order to reduce redundancy.

3. apply PCA on the remaining attributes, and keep only the 5 most important dimensions.

4. use a neighborhood algorithm to classify the test shots, thus obtaining the final, fused score.

Two runs were submitted: IRIM2 utilizing the KNNB attribute set, with all of the 48 attribute variants, and IRIM3 utilizing the KNNC attribute set (a variant of the KNNB with a per concept optimization of some hyper-parameters), with 110 attribute variants (the 48 available multiplied by some additional variations on the α parameters and/or on the number of components kept after PCA reduction). Both attribute sets and classification results were provided by the IRIM consortium.

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 $k \in \{1 \dots \mathtt{K}\}$ provides scores $\mathbf{x}_k = [x_{k1}, \dots, x_{k\mathtt{N}}]$ indicating the likelihood for each shot $n \in \{1 \dots \mathtt{N}\}$ to contain the requested concept. The objective is to find a combination function $\mathbf{f}$ so that the resulting classifier $\mathbf{x} = \mathbf{f}(\mathbf{x}_1, \dots, \mathbf{x}_\mathtt{K})$ 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.* [18].

IRIM 4: Hierarchical fusion It can be divided into three consecutive steps.

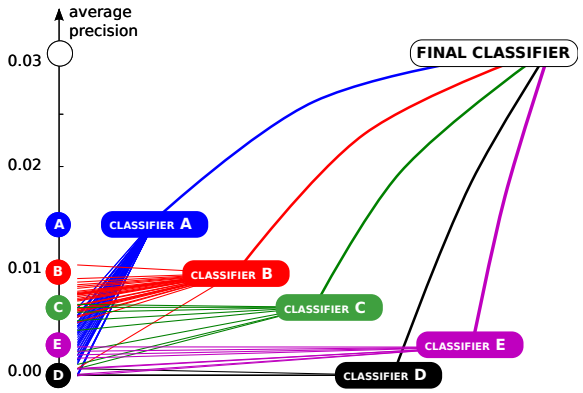

Figure 1: Community-driven hierarchical fusion

Step 1: community detection. Classifiers are automatically grouped into $\mathtt{C}$ 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.

IRIM 1 IRIM 1 is the re-ranked version of IRIM 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 pre-

dicted by classifiers. It is often possible to improve the retrieval performance by re-ranking the samples. *Safadi and Quénot* in [19] 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 following runs:

F_A_IRIM1_1: LIMSI community-driven hierarchical fusion with re-ranking;

F_A_IRIM4_4: original LIMSI community-driven hierarchical fusion;

F_A_IRIM2_2: LISTIC 'selection − fusion − PCA − neighborhood' with 48 attributes;

F_A_IRIM3_3: LISTIC 'selection − fusion − PCA − neighborhood' with 110 attributes.

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 correspond ot a rank of 5 within the 19 participants to the TRECVID 2011 full SIN task. The difference between the F_A_IRIM1_1 and F_A_IRIM4_4 runs is that re-ranking has been applied to the first one. The gain obtained by the re-ranking is statistically significant but less than expected. Maybe the re-ranking parameters were not optimal for the type of fusion used. Between IRIM2 and IRIM3, the 110-attribute version performed slightly less well than the 48-attribute one. This may be due also to the fact that slightly different versions of classifier were used.

Table 2: InfMAP result and rank on the test set for all the 50 TRECVID 2011 concepts (full task).

System/run	MAP	rank
Best run	0.1731	1
F_A_IRIM1_1	0.1387	15
F_A_IRIM4_4	0.1341	17
F_A_IRIM2_2	0.1194	25
F_A_IRIM3_3	0.1142	30
Median run	0.1083	34

type	overall number of examples	number of different instances	mean number of examples per instance
PERSON	38	8	4.75
CHARACTER	24	5	4.8
OBJECT	32	8	4
LOCATION	4	1	4
total	98	22	4.45

Figure 3: Distribution of instances for devel set 2011

2 Instance Search

2.1 task presentation

Instance Search (INS) is a pilot task introduced by NIST in TRECVid 2010 Campaign and continued in 2011. Given visual examples of entities of limited number of types: person, character, object or location, it consists in finding segments of videos in the data set which contain instances of these entities. Each instance being represented by several example images.

Hence if we can see the set of video clips as a visual database, the problem consists in retrieval of each instance in this database.

For this task in 2010 and even 2011, only a few examples of each instance are available to formulate the "query". For each instance, a mask of it in video frame was also available for visual example.

As last year, this task is yet only to explore task definition and evaluation. Only a rough estimate of searched instances locations was asked. Indeed, we had to find only the movies were the instance appeared, but not the precise frame or the precise location of instance in the frame.

2.1.1 instances examples and data sets

This year, as in 2010, 4 types of instances were proposed: person, character, object, location. Instances of development and test sets are presented in tables 2 and 4. Tables 3 and 5 show the distribution of instances by type and the number of examples for each type. We can see that types of instances are quite different between the two data sets: instances for devel data set are mainly PERSON and CHARACTER, instances for test set are mainly OBJECTS. Besides, the mean number of instances examples per instance has slightly decreased in the test set compared to development set.

Devel and test data sets videos are also quite different. Devel data set is composed of Dutch TV programs, i.e., edited content. Test data set is composed of rushes, that is raw, unedited data, of BBC series or documentaries.

number	type	text	number of examples for queries
9001	PERSON	George W. Bush	5
9002	PERSON	George H. W. Bush	4
9003	PERSON	J. P. Balkenende	5
9004	PERSON	Bart Bosh	5
9005	CHARACTER	Professor Fetze Alsvanouds from the University of Harderwijk (Aart Staartjes)	5
9006	PERSON	Prince Bernhard	5
9007	CHARACTER	The Cook (Alberdinck Thijn: Gijs de Lange)	5
9008	PERSON	Jeroen Kramer	5
9009	CHARACTER	Two old ladies, Ta en To	5
9010	CHARACTER	one of two officeworkers (Kwelder of Benema en Kwelder: Harry van Rijthoven)	5
9011	PERSON	Colin Powell	3
9012	PERSON	Midas Dekkers	5
9013	OBJECT	IKEA logo on clothing	5
9014	CHARACTER	Boy Zonderman (actor in leopard tights and mesh top: Frank Groothof)	4
9015	OBJECT	black robes with white bibs worn by Dutch judges and lawyers	3
9016	OBJECT	zebra stripes on pedestrian crossing	4
9017	OBJECT	KLM Logo	2
9018	LOCATION	interior of Dutch parliament	4
9019	OBJECT	Kappa Logo	5
9020	OBJECT	Umbro Logo	5
9021	OBJECT	tank	3
9022	OBJECT	Willem Wever van	5

Figure 2: Instances for devel set 2011

number	type	text	number of examples for queries
9023	OBJECT	setting sun	3
9024	LOCATION	upstairs, inside the windmill	2
9025	OBJECT	fork	5
9026	OBJECT	trailer	2
9027	OBJECT	SUV	4
9028	OBJECT	plane flying	5
9029	LOCATION	downstairs, inside the windmill	3
9030	OBJECT	yellow dome with clock	3
9031	OBJECT	the Parthenon	5
9032	OBJECT	spiral staircase	2
9033	OBJECT	newsprint balloon	4
9034	OBJECT	tall, cylindrical building	3
9035	OBJECT	tortoise	5
9036	OBJECT	all yellow balloon	3
9037	OBJECT	windmill seen from outside	3
9038	PERSON	female presenter X	5
9039	PERSON	Carol Smilie	3
9040	PERSON	Linda Robson	5
9041	OBJECT	monkey	5
9042	PERSON	male presenter Y	5
9043	PERSON	Tony Clark's wife	6
9044	OBJECT	American flag	4
9045	OBJECT	lantern	3

Figure 4: Instances for test set 2011

2.2 Search methods

An instance as defined in the task is an object in an image. Hence it is natural to search for an object in frames of video clips. Such type of query would be adapted to the situation when an object in video clips

type	overall number of examples	number of different instances	mean number of examples per instance
PERSON	24	5	4.75
CHARACTER	0	0	0
OBJECT	59	16	3.68
LOCATION	5	2	2.5
total	88	23	3.82

Figure 5: Distribution of instances for test set 2011

evolves in a different context than in a query example frame. When the content of a database of clips is such that the query object evolves in the same context as in a query exmaples, the use of context would enhance the result. Hence in our apporach we used both: object based query and frame based query.

2.2.1 Object based and frame based queries

We formulated the query in standard way as comparison of visual signatures either of object with parts of DB frames or as a comparison of visual signatures of query and DB frames. To produce visual signatures we also used two apporaches . The first one is the baseline Bag-Of-Visual-Words (BOVW) model based on interest point descriptor, as proposed by Sivic and Zisserman[20]. The descriptor used is SURF (Speeded Up Robust Features)[21]. The second approach is a Bag-Of-Regions (BOR) model, as proposed by Vieux et al. in [22], that extends the traditional notion of BOVW vocabulary not only to keypoint-based descriptors but to region based descriptors. In this second approach regions in image plane are obtained by segmenting images by Felzenszwalb and Huttenlocher method [23].

The BOVW approach was used both for object signatures construction and for frame signature construction. As for region-based approach, it was deployed only for the whole frame. Figure 6 illustrates these approaches. In Figure 6 first line, we present the information available for query for example 1 of instance 9026: the original frame and object mask. Figure 6 second line depicts the global and local signature computation for BOVW: the features points are extracted for the whole image or only on the object mask. In figure 6 third line, we show the segmented regions for frame signature computation for BORW.

2.2.2 Features for signatures

As mentioned above, for BOVW we computed standard SURF features.

For BOR approach, the global feature such as HSV histogram was computed, expressing color distribution. For this histogram, we set a uniform quantizing parameters in order to limit the feature size to approximately 100 bins and to privilege the finest encoding of Hue component. This led us to 45+32+32 bins in the feature representing concatenated normalized marginal distributions. We note that HSV histograms of frames proved to be an efficient feature for video similarity search [24]. As our problem is similar, the choice of this feature is straightforward.

2.2.3 Computation of visual dictionaries

Both BOVW and BOR suppose the availability of a dictionary or codebook. For the BOVW dictionary computation, we used the unsupervised clustering K-means++ with a large number of clusters (16384), with the L2 distance. For the BOR, we used the incremental clustering algorithm described in [25] and modified in [22], with 2000 clusters and L2 distance, thus yielding a Bag of Region Words (BORW) model.

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.

2.2.4 Search of instances

As stated in task presentation, the search of video clip containing an instance can be expressed as a problem of query-by-example in an image database. Here the example image Q is the keyframe containing the concept. The database DB is a set of keyframes of all video clips of the test set. Both the Q and DB are characterized by BOW build on chosen feature space. Hence the problem to address is the computation of similarity measure S between BOW(Q) and BOW(I)/IєDB.

In order to compare BOW(Q) and BOW(R) we used the L1 distance for BORW method and the complement of histogram intersection for BOVW method.

Let us consider now the object based query. We have to compare the signature with potential objects in DB frame. The problem here is the locus of the object in DB frame is unknown. Hence, we used a correlation kernel, deforming object mask according to Pan/Tilt/Zoom affine model. The correlation was done by full search in the offline parameter space. Pan 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, 4. 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.

For BORW, according to preliminary tests, we have chosen to use only frame-based query. In this case, the

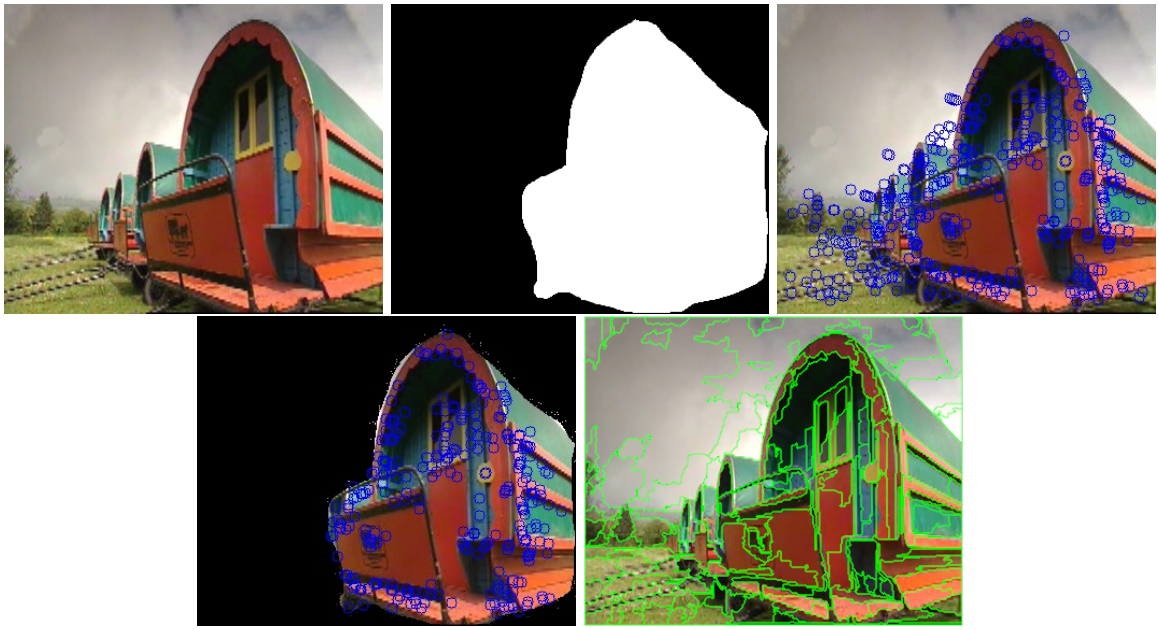

Figure 6: Example of instance from test set: original image, mask, interest points on whole image, interests points on mask, segmented regions on whole image.

visual signatures are precomputed for the whole set of DB images.

All available information for search instances was used. Indeed, we made query comparing signatures of all examples frames available for a given instance with all the DB frames. The fusion of results was done by mean operator with a further re-ranking.

2.3 Runs

Test data set is composed of rushes, i.e., raw, unedited data. This kind of data often contains several takes of the same scene, maybe be with a different camera angle. We expect that images between theses takes could be quite similar. Hence use of context such as global BORW signatures for examples frames and DB is justified. Furthermore, if object based query is considered, the mask of query could be small. This would entail too few points inside the mask. Thus, in our runs, we wanted to limit the use of the mask for the query when we had enough points. After studying query images and available masks, we have decided to use mask for BOVW only if we had at least 8 interest points detected.

We have computed four results: BOVW for the whole frame, BOVW for object based query supposing the object in DB frames is of approximately the same and at the same position as in query example, BOVW for object based query with affine deformation , BORW for the whole image. These results are computed for all keyframes (RKF and NRKF). Finally, We have submitted four fully automatic runs:

- run1: we merge BORW and BOVW both for the whole frame.
- run2: if we have enough points of interest in query, we merge BORW for the whole frame and BOVW results for object based query with affine deformation. Otherwise, we keep only BORW for the whole frame results.
- run3: if we have enough points of interest in query, we merge BORW for the whole frame and BOVW results for object based query without affine deformation. Otherwise, we keep only BORW for the whole frame results.
- run4: pure BORW for the whole frame results.

2.4 Results

There were 37 fully automatic runs submitted this year. Table 7 presents our results for the different runs, for the various instances and in average.

We can see that:

- Our runs sorted from best to worst are : run1, run3, run4 and run2.
- All four runs are better than median.

topic	run1		run2		run3		run4	
	map	rank	map	rank	map	rank	map	rank
9023	0.1080	9	0.0839	15	0.0839	15	0.0846	14
9024	0.3814	15	0.3819	14	0.3813	15	0.3637	17
9025	0.0994	4	0.0244	22	0.0956	6	0.1075	3
9026	0.3127	3	0.3267	2	0.2742	4	0.2052	8
9027	0.2491	13	0.0167	27	0.2466	15	0.2488	13
9028	0.4177	12	0.1031	22	0.4005	13	0.3654	15
9029	0.3764	13	0.3752	15	0.3764	13	0.3771	12
9030	0.2000	15	0.1972	16	0.2070	13	0.2011	14
9031	0.2771	12	0.1178	17	0.3366	8	0.2965	11
9032	0.3726	12	0.2524	17	0.2826	16	0.2438	18
9033	0.3996	10	0.0306	14	0.3411	13	0.3591	12
9034	0.2345	7	0.1509	13	0.2087	8	0.2056	9
9035	0.3874	7	0.3901	6	0.3962	5	0.4026	4
9036	0.3436	7	0.3077	8	0.3077	8	0.3062	11
9037	0.1367	14	0.1165	16	0.1227	15	0.1000	17
9038	0.3169	5	0.3072	9	0.3170	6	0.3143	8
9039	0.0462	18	0.0444	20	0.0469	17	0.0367	21
9040	0.1533	13	0.1523	14	0.1461	15	0.1539	12
9041	0.2286	11	0.0034	30	0.2092	13	0.2124	12
9042	0.1181	12	0.0487	18	0.0982	13	0.0897	14
9043	0.4994	1	0.2769	8	0.4910	3	0.4971	2
9044	0.1594	13	0.0738	19	0.1409	15	0.1424	14
9045	0.2898	15	0.2117	19	0.2819	16	0.2773	17
9046	0.4206	12	0.1329	18	0.3943	13	0.3929	14
9047	0.3099	6	0.0290	21	0.2829	8	0.2927	7
mean	0.2735	10.36	0.1662	16	0.2588	11.44	0.2511	11.96

Figure 7: Results for 4 runs on test set 2011

- run1, run3 and run4 are in the first third of the sorted results.
- The fact that run3, object based query without affine deformation, outperforms run2, object based query with affine deformation is surprising. This has to be further investigated.

2.5 Discussion

In view of these results, one thing can be stated that merging results for visual signatures for local features (BOVW approach) and region features (BORW approach) gives better performances than BORW alone. The latter has proven to outperform the classical BOVW approach on some data sets [22].

In our opinion, the choice of optimal approach: BOVW for object based query, whole frame based query, BORW, or their combination is very much dependent on data. Indeed for SIVAL dataset [26], we obtained better results for the object based query with affine deformation than for object based query without affine deformation.

Furthermore, the optimal combination on development set can remain optimal on test set, only if we have the same characteristics of BOW in terms of global structure of visual scene: presence of objects in the same context or different contexts.

Now, as the ground truth on instances on the test set is available, we have to investigate other fusion methods and do more balanced tuning of our algoirthms.

As a conclusion, we stress that our approach was totally generic. We do not use the knowledge that some instances represented persons for example. All the queries were considered containing generic objects.

3 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 2011 − An Overview of the Goals, Tasks, Data, Evaluation Mechanisms, and Metrics In *Proceedings of the TRECVID 2011 workshop*, Gaithersburg, USA, 5-7 Dec. 2011.

[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] D. Gorisse, M. Cord, F. Precioso, SALSAS: Sub-linear active learning strategy with approximate k-NN search, Pattern Recognition, In Press, Corrected Proof, Available online 21 December 2010.

[6] 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.

[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. 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

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

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

[14] Safadi et al. Quaero at TRECVID 2011: Semantic Indexing and Multimedia Event Detection, In *Proceedings of the TRECVID 2011 workshop*, Gaithersburg, USA, 5-7 Dec. 2011.

[15] 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.

[16] 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.

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

[18] 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.

[19] B. Safadi, G. Qunot. 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.

[20] 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.

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

[22] 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.

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

[24] Emilie Dumont and Bernard Merialdo. Rushes video summarization and evaluation. *Multimedia Tools and Applications, Springer, Vol.48, N1, May 2010*, 2010.

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

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