Quaero at TRECVid 2013: Semantic Indexing

Bahjat Safadi[1], Nadia Derbas[1], Abdelkader Hamadi[1], Thi-Thu-Thuy Vuong[1], Han Dong[1], Philippe Mulhem[1], and Georges Quénot[1]

[1]UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble INP / CNRS, LIG UMR 5217, Grenoble, F-38041, France

Abstract

The Quaero group is a consortium of French and German organizations working on Multimedia Indexing and Retrieval[1]. LIG participated to the semantic indexing main task, localization task and concept pair task. LIG also participated to the organization of this task. This paper describes these participations which are quite similar to our previous year's participations.

For the semantic indexing main 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 used a number of different descriptors and a hierarchical fusion strategy. We also used conceptual feedback by adding a vector of classification score to the pool of descriptors. The best Quaero run has a Mean Inferred Average Precision of 0.2848, which ranked us 2^{nd} out of 26 participants. We also co-organized the TRECVid SIN 2013 task and collaborative annotation.

1 Participation to the organization of the semantic indexing task

For the Fourth year, UJF-LIG has co-organized the semantic indexing task at TRECVid with the support of Quaero. A list of 500 target concepts has been produced, 346 of which have been collaboratively annotated by the participants and by Quaero annotators. A subset of 60 of them was selected for participants' submissions and 38 of which have been officially evaluated.

The 500 concepts are structured according to the LSCOM hierarchy [14]. They include all the TRECVid "high level features" from 2005 to 2009, the CU-VIREO374 set plus a selection of LSCOM concepts so that we end up with a number of generic-specific relations among them. We enriched the structure with two relations, namely *implies* and *excludes*. The goal was to promote research on methods for indexing many concepts and using ontology relations between them.

TRECVid provides participants with the following material:

- a development set that contains roughly 600 hours of videos;
- a test set that contains roughly 200 hours of videos;
- shot boundaries (for both sets);
- a set of 500 concepts with a set of associated relations;
- elements of ground truth: some shots were collaboratively annotated. For each shot and each concept, four possibilities are available: the shot has been annotated as positive (it contains the concept), the shot has been annotated as negative (it does not contain the concept), the shot has been skipped (the annotator cannot decide), or the shot has not been annotated (no annotator has seen the shot).

The goal of the semantic indexing task is then to provide, for each of the 60 selected concepts, a ranked list of 2000 shots that are the most likely to contain the concept. The 2013 test collection contains 112,677 shots. More information about the organization of this task can be found in the TRECVid 2013 overview paper [17]. A *pair* version of the task in which 10 pairs of concepts (e.g. Car+Bicycle) has also been proposed this year.

1.1 Development and test sets

Data used in TRECVid are free of right for research purposes as it comes from the Internet Archive

[1]http://www.quaero.org

Table 1: Collection feature

Characteristics	TRECVid 2010
#videos	27,964
Duration (total)	~800 hours
# shots	879, 873
# shots (dev)	545, 923
# shots (test 2013)	112, 677
# shots (test 2014)	107, 806
# shots (test 2015)	113, 467

(http://www.archive.org/index.php). Table 1 provides the main characteristics of the collection set.

The whole set of videos has been split into two parts, the development set and the test set. The test set has been split in three part dedicated to the TRECVid SIN evaluations of 2013, 2014 and 2015. This has been done in order to be able to measure the performance progress over the three years. All sets were automatically split into shots using the LIG shot segmentation tool [15].

1.2 The evaluation measure

The evaluation measure used by TRECVid is the MAP (Mean Average Precision). Given the size of the corpus, the inferred MAP is used instead as it saves human efforts and has shown to provide a good estimate of the MAP [16].

1.3 Annotations on the development set

Shots in the development set have been collaboratively annotated by TRECVid 2010-2012 participants and by Quaero annotators. As concepts density is low, an active learning strategy has been set up in order to enhance the probability of providing relevant shots to annotators [2]: the active learning algorithm takes advantage of previously done annotations in order to provide shots that will more likely be relevant. Although this strategy introduces a bias, it raises the number of examples available to systems. Moreover, it exhibits some trend in the concept difficulty. As an example, the number of positive examples for the concept *Person* is larger than the number of negative examples. This means that the active learning algorithm was able to provide more positive examples than negative ones to annotators, meaning that *Person* is probably a "too easy" concept.

346 concepts were annotated on IACC.1.C (tv12 test) by Quaero annotators. The new annotations were once again using an active learning approach bootstrapped with a fusion of all the TRECVid SIN 2012 submissions. This also ensures that the active learning based annotation is not biased in favour of the system used for the active learning process. An improved algorithm for annotation cleaning has also been used in the annotation tool this year [11]. 8,158,517 were made directly by annotators and a total of 28,864,844 was obtained by propagating them using "implies" or "excludes" relations.

1.4 Assessments

38 concepts were selected for evaluation out of the 60 ones for which participants were asked to provide results for the main SIN task. Assessments were done partly by NIST (15 concepts) and by Quaero (23 concepts). Assessments were done by visualizing the whole shot for judging whether the target concept was visible or not at any time within the shot. Additionally, all the 10 concept pairs were selected for evaluation of which 5 were annotated by NIST and 5 were annotated by Quaero. A total of 202,707 concept $\times$ shots assessments were made by Quaero.

2 Participation to the semantic indexing main task

2.1 Introduction

The TRECVid 2013 semantic indexing task is described in the TRECVid 2013 overview paper [1, 17]. 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." 60 concepts have been selected for the TRECVid 2013 semantic indexing task. Annotations on the development part of the collections were provided in the context of the collaborative annotation and by Quaero.

As last year, our system 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 considered (section 2.2).

2. Descriptor optimization. A post-processing of the descriptors allows to simultaneaously improve their performance and to reduce their size (section 2.3).

3. Classification. Two types of classifiers are used as well as their fusion (section 2.4).

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 2.5).

5. Higher-level fusion. We fuse here descriptors of different types, e.g. color, texture, interest points, motion (section 2.6).

6. Re-ranking. We post-process here the scores using the fact that videos statistically have an homogeneous content, at least locally (section 2.7).

Additionally, our system includes a conceptual feedback in which a new descriptors is built using the prediction scores on the 346 target concepts is added to the already available set of 47 audio and visual descriptors (section 2.8). Compared to last year, our system has been improved by the inclusion of new descriptors, an improved desctiptor-classifier joint optimization and an improved scheme for hierarchical late fusion.

2.2 Descriptors

A total of 57 audio and visual descriptors have been used. Many of them have been produced by and shared with the IRIM consortium and two of them were provided by Xerox (XRCE). These include variants of a same descriptors (e.g. same methods with different histogram size or image decomposition). 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. They are described and evaluated in the IRIM consortium paper [8]. They include color histogram, Gabor transform, quaternionic wavelets, a variety of interest points descriptors (SIFT, color SIFT, SURF), local edge patterns, saliency moments, percepts, and spectral profiles for audio description. Many of them rely on a bag of words approach.

2.3 Descriptor optimization

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

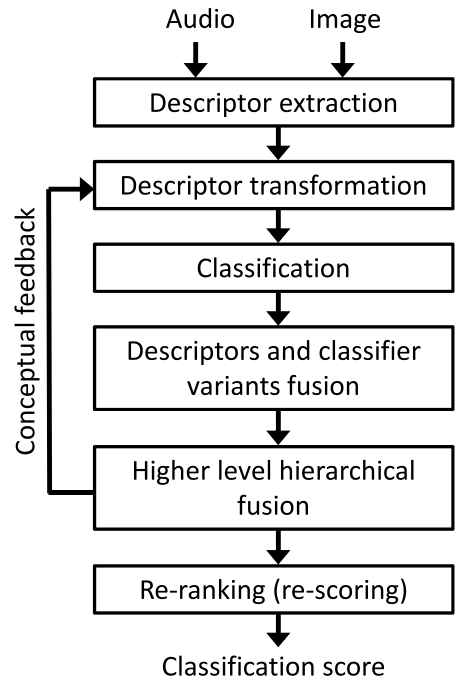


Figure 1: Semantic indexing system

2.3.1 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 $x < 0$) transformation 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.

The optimization of the value of the α coefficient is optimized by two-fold cross-validation within the development set. It is done in practice only using the LIG_KNNB classifier (see section 2.4) 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 this hyper-parameter.

2.3.2 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 number of components kept in the PCA reduction is also optimized by two-fold cross-validation within the development set using the LIG_KNNB classifier. Also, the overall performance is not very sensitive to the precise values for this number.

2.3.3 Second power transformation

A second power transformation can be applied after PCA dimendionality reduction/ It has an affect which is similar to a post-PCA whitening but is has been proven to be more efficient and easy to tune. The optimal value of α_2 can be optimized by cross-validation and is often close to 0.7 [12].

2.4 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 [5], 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_BUSEB: Fusion between the two available classifiers. The fusion is simply done by a MAP weighted average of the 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 [6]. 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. The MAP values used for the weighting are obtained by a two-fold cross-validation within the development set.

2.5 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 3×6 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, 256, 512 and 1024; 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) [7]. 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 results from different information sources. The gain at level 3 is also cumulated with the gain at the lower levels.

2.6 Final fusion

Hierarchical fusion with multiple descriptor variants and multiple classifier variants was used and optimized for the semantic indexing task. We made several experiment in order to evaluate the effect of a number of factors. We optimize directly the first levels of the hierarchical fusion using uniform or average-precision weighting. The fusion was made successively on variant of the same descriptors, on variant of classifiers on results from the same descriptors, on different types of descriptors and finally on the selection of groups of descriptors.

2.7 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 [10] 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%.

2.8 Conceptual feedback

Since the TRECVid SIN 2013 task considers a quite large number (346) of descriptors and since these are also organized according to a hierarchy, one may expect that the detection scores of some concept help to imrpove the detection score of related concepts. We have made a number of attempts to use the explicit *implies* or *excludes* provided relations but these were not successful so far, maybe due to a normalization problem between the scores of the different concepts. We tried then an alternative approach using the implicit relations between concepts by creating a vector with the classification scores of all the available concepts [13]. We used for that the best hierarchical fusion result available. This vector of scores was then included as a 58^{th} one in the pool of the 57 already available descriptors and processed in the same way as the others, including the power and PCA optimization steps and the fusion of classifier outputs. The found optimal power value was quite different of the ones for the other descriptors (1.800 versus 0.150-0.700) for the other ones. This is probably linked with the way the score normalization is performed. Even though the 2013 evaluation is done on 60 concepts only, as the annotations are available for 346 concepts, we used the full set for the conceptual feedback.

2.9 Performances on the semantic indexing task

Four slightly different combinations of hierarchical fusion have been tried. The variations concerned the way the re-ranking was done: it can be locally temporal, globally temporal and or conceptual. Not all combinations could be submitted and the following were selected:

M_A_Quaero-2013-1_1: combination of M_A_Quaero-2013-3_3 with uploader information with 3:1 weights;

M_A_Quaero-2013-2_2: combination of M_A_Quaero-2013-3_3 with uploader information with 7:1 weights;

M_A_Quaero-2013-3_3: manually built hierarchical fusion of a large number (over 100) of jointly optimized descriptor-classifier combinations including two iterations of conceptual feedback combined with temporal re-ranking;

M_A_Quaero-2013-4_4: manually built hierarchical fusion of a large number (over 100) of jointly optimized descriptor-classifier combinations including a single iterations of conceptual feedback combined with temporal re-ranking.

Table 2: InfAP result and rank on the test set for all the 38 TRECVid 2013 evaluated concepts

System/run	MAP	rank
Best submission	0.3211	1
M_A_Quaero-2013-3_3	0.2848	5
M_A_Quaero-2013-2_2	0.2846	6
M_A_Quaero-2013-4_4	0.2835	9
M_A_Quaero-2013-1_1	0.2827	11
Median submission	0.1275	46
Random submission	0.0009	-

Table 2 shows the performance of the four submitted variants. Our submissions ranked between 5 and 11 in a total of 90 for type A conditions. Our best submission ranked us as the second group out of 26 for the main SIN task. The second iteration of conceptual feedback brings a quite small improvement (from 0.2835 to 0.2848). The runs including uploader information actually [9] contained a bug due to our misunderstanding of the data provided by our partner that computed it for us. Then, while we expected an improvement, we obtained a slight degradation, almost negligibe for the 7:1 weighting and still small for the 3:1 weighting.

3 Participation to the semantic indexing localization task

3.1 Introduction

For this year, TRECVid has started an optional subtask of localization, of the Semantic Indexing task (SIN), is started to challenge systems participating in the SIN task to make their concept detection more precise in time and space. Currently systems are accurate to the level of the shot. In the localization subtask, systems are asked to determine the presence of the concept temporally within the shot, i.e., with respect to a subset of the frames comprised by the shot, and, spatially, for each such frame that contains the concept, to a bounding rectangle. The localization are restricted to ten concepts from those chosen used in the main task of SIN. These concepts are: *Airplane, Boat_Ship, Bridges, Bus, Chair, Hand, Motorcycle, Telephones, Flags and Quadruped*.

Our work for this subtask was inspired by some works on the discriminative color model. A discriminative color model can be used to classify individual pixels of images with regards to whether they may belong to the wanted object. It assigns a Boolean value to each pixel of a query image indicating whether the pixel may belong to an object of the given object class. Thus, the color model yields a binary map of positive and negative pixels.

Color models can be used for pre-filtering images or for creating regions of interest for more sophisticated classification systems. In other words, color models can be useful for large datasets for image classification systems. Also, color models can be used for quickly rejecting unambiguously negative images prior to applying more sophisticated (and thus computationally more expensive) classifiers.

Our work relies to the work done in [18], where the authors proposed a method for creating a discriminative color model for a given object class based on color occurrence statistics. However, in contrast to existing approaches, they do not exploit pixel-wise object annotations but only global negative and positive image labels. The authors use their discriminative color model for detecting different brand logos in images and for recognizing different flower classes.

In the following we report our participation at the localisation task in TRECVid.

3.2 Approach

The approach is given in figure 2, and it consists of two steps: i) for each concept, we apply the semantic indexing (SIN) system, which aims to choose the shots that contain the target concept; ii) we apply a localisation function on the frames of the top 1000 resulting shots of the SIN system.

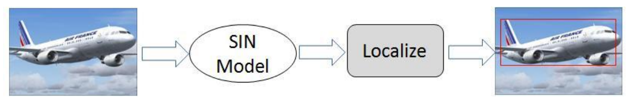


Figure 2: The framework of the localization system.

3.2.1 Semantic indexing system

The video semantic indexing system used for the localization task is exactly the same as the one used for the semantic indexing main task, see section 2.

3.2.2 Localization algorithm

Since the considered concepts occurs in an unlimited number of different colors and in very different background areas, we choose to compute a discriminative model based on the sift representation instead of color. Our model is computed from the sift occurrence statistics, which we determine from a set I_p of positive images and a set I_n of negatives images from the the development data set. The idea is to determine discriminative concept SIFTs, which appears significantly more often in positive images than in negative images.

We first compute the interest points for each image and represent these points with the SIFT Harris descriptor. Then we apply a clustering approach on all the SIFT points, of all the images, in Y different clusters. This results in assigning one specific cluster ($y \in Y$) to each SIFT point. To determine which SIFT clusters appear more often in positive images as they are likely to indicate the considered concept, we build, for each concept, a Relative Occurrence Frequencies (ROF) of the clusters within the positive and negative images.

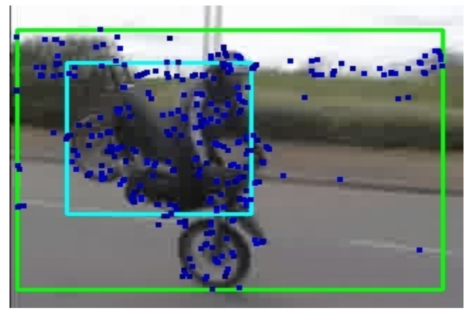
For each concept ($c \in C$), let $p = |I_p|$ and $n = |I_n|$ be the numbers of positive and negatives images, respectively. Then the respective relative occurrence frequencies are given by: $ROF_p(y) = \frac{1}{p}p'_y$, and $ROF_n(y) =$

$\frac{1}{n}n'_y$, where p'_y and n'_y are the absolute numbers of positive and negative images, respectively, in which at least one point belonging to the cluster y is present in the image. However, the set of negative images (i.e. n) is significantly bigger than the set of positive images (i.e. p). We decide to consider only the ROF_p for each cluster y.
To localize a given concept in an image, we need to determine two points of a rectangle around the concept. Ideally, the good localization is obtained when having SIFT points only on the concept in the image, then it is easy to determine the rectangle which covers these SIFTs. However, in practice the images contain many concepts and the SIFTs appear almost on the whole image. Thus, it is better to filter the SIFT points of each image in a way to have SIFTs only on the considered concept. We propose a method based on this filtering. The method first assigns to each SIFT the ROF of its cluster. Then, we calculate a histogram of the ROF of the image's SIFTs and normalize their locations into s bins, on both X and Y-axis. We define a threshold β to filter the ROF histogram. The idea is to find the two points that cover the SIFTs located on the concept. However, this β is tuned for each concept separately. Since, we don't have yet any evaluation metric for the localization, we have tuned β manually on a small collection.
Figure 3(b) illustrates an example of the bounding box and the filtering algorithm we propose. As the plot shows, there are many SIFT points (i.e. the blue points), the rectangular in green presents the localization result of concept *Motorcycle* by taking all the SIFTs in the image (frame). The blue rectangular defines the localization after the filtering that we propose. As we can see, the filtering can help to precise better the localization of concepts, especially, when the concept appears as the main object in the image.
In our work, we have applied the SIFT clustering on $Y = 4096$ different cluster. We have fixed the number of bins for the histogram of ROF to $s = 32$.

3.3 Experiments

The experiments on the concept localization in videos, were conducted on the TRECVid 2013 collection. This data collection consists of two large sets: the development and the test set. The development set contains 545,923 annotated video shots, while the test set contains 112,677 shots. The evaluations were conducted on the ten semantic concepts, which were provided by NIST and Quaero for the localization subtask and that are included in the list of the concepts for the SIN task. The samples of the development set are annotated as containing the concept or not, however there are no information about the localization of the concept within the shots. This makes the task very difficult since the learned samples have no information about where the learned concepts are in the shots.

(a) Initial image



(b) Concept localization

Figure 3: Example of the localization of the concept *Motorcycle* using ROF. a) is the initial image; b) image with SIFT points, and two bounding rectangles around the concept, the rectangle in green is the base-line and the blue one is the proposed method with $\beta = 0.25$.

The tuning parameter β of the localization method was tuned and optimized on the development set, in which we have manually examined the localization in the top 500 retrieval images for each concept. We have choose to fix the β to the value that we think it gives more accurate bounding rectangles around the concepts. Note: since the localization task in TRECVid is a new task, we do not have yet any information about the evaluation metric, which will be used for the evaluation. Thus all the examinations and analysis of the experiments will be based on viewing some of the frames manually by ourselves.

3.4 Effectiveness and some results

We have applied the localization algorithm on each of frame of the top 1000 retrieved shots by the SIN system. The SIN system has retrieved coherent number of positive samples for each concept, and the concept may not appear in all the frames of a positive shot. However, the localization algorithm was applied on all the frames and resulted in drawing a rectangle around the region where it expected to contain the concept. Figure 4 shows some examples of the bounding boxes for the learned concepts. By viewing We have observed from the results that the indexing system was quite good for most of the concepts. The localization was quite good for concepts as Airplane, Motorcycle, Hands and Quadruped, which are considered to cover most the shot pixels in the training set. For the other concepts, the localization was not good, since we believe that these concepts do not appear as the main concept in the training set. For example, concept Flags appears always with president, concept Chair appears with people (people sitting on chairs), concept Telephone is normally very small and appears usually with people and other concepts. Some of the concepts are not well detected (the indexing is not very good) but the localization algorithm could find the main concept within the frames. For instance, concept Bus and Boat_ship.

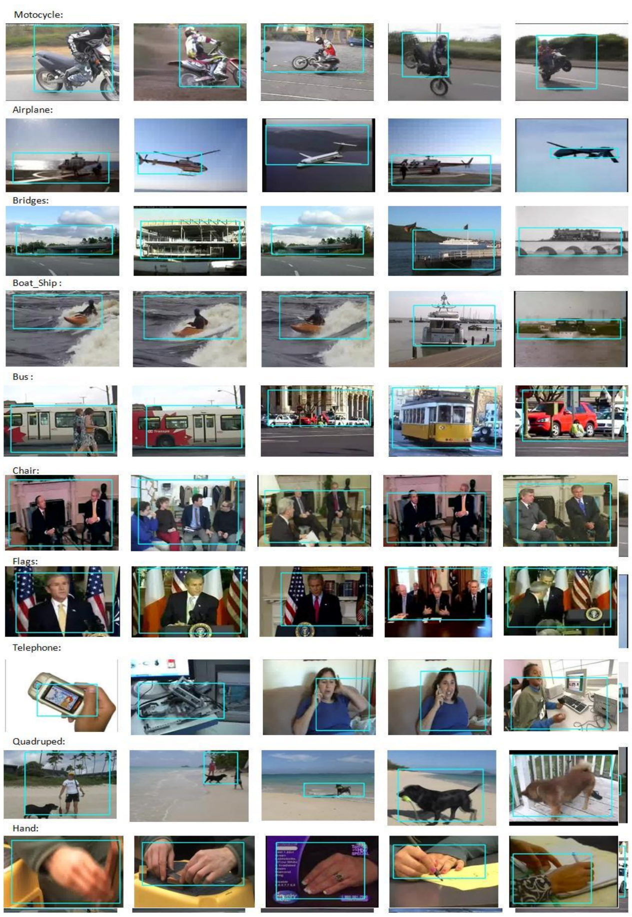


Figure 4: Some results of the localization approach.

We submitted only one run based on an unsupervised system. We applied the localization algorithm on each of frame of the top 1000 retrieved shots by the SIN system. The SIN system has retrieved coherent number of positive samples for each concept, and the concept may not appear in all the frames of a positive shot. However, the localization algorithm was applied on all the frames and resulted in drawing a rectangle around the region where it expected to contain the concept.

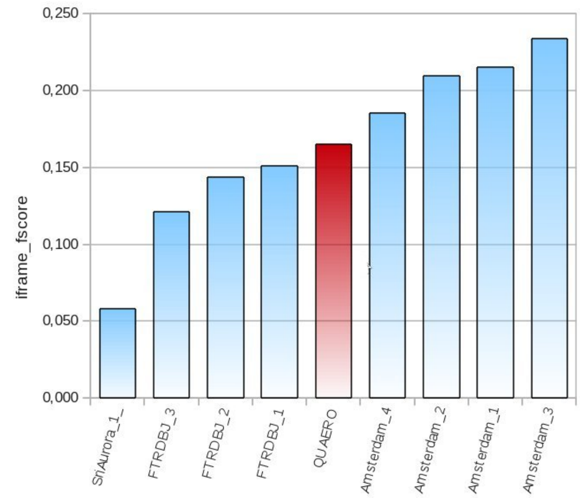


Figure 5: The mean iframe fscore per run

Four teams participated in TRECVid 2013 in the subtask of localization for the Semantic Indexing task (SIN) with nine different runs. The used metric is the iframe fscore, the figure 5 shows the mean iframe fscore obtained over the 10 concepts for the nine runs. Our approach had the best results for the completly unsupervised system with a mean iframe fscore of about 16, 51%. Differently from our unsupervised approach, the first ranked team got the best results for the supervised system with a mean iframe fsore of about 23.36%. They used an approach based on the annotations at the object level. They obtained those annotations by manually adding a bounding box to a small number of image-level labels from the development collection.

4 Participation to the semantic indexing concept pair task

For the concept pair detection task, we compared a baseline approach in which we simply average the score obtained by the individual concept classifiers with two alternative methods and a combination of all of them. The first alternative method is a direct learning of the concept pair as if it was a single concept. The second one is a two-step ranking.

Baseline: We considered as a baseline system a simple combination of the two single concept detectors. To detect a pair of concept (c1,c2), we built a single concept detector for $c1$ and $c2$, as described above, by considering the annotations by $c1$ and $c2$. The new detection score of (c1,c2) is given by : $NewScores_i^{(c1,c2)} = \frac{1}{2}(scores_i^{c1} + scores_i^{c2})$, where $scores_i^{c1}$ and $scores_i^{c2}$ are the detection scores of $c1$ and $c2$, respectively.

Learning based approach: After generating the annotation per concept pairs, we built concept pairs detectors using MSVM learners for each pair of concepts and each feature. A late fusion of the scores obtained in the first stage is performed in order to improve performance. This is achieved by averaging, for each sample, a number of scores obtained using different descriptors as detailed in 2. We call the results of this fusion in the following $late\_fusion$.

Two-step ranking We propose a scoring approach which is based on single concepts detectors. The idea consists of detecting a concept pair ($c1$, $c2$) based on the results of the detection of the other. So, firstly, we calcule the detection scores of $c1$ and $c2$ in the whole dataset to obtain the detection scores $scores_i^{c1}$, $scores_i^{c2}$, respectively. Secondly, we rank the samples based on $scores_i^{c1}$. Finally, we rank the list already ranked in the first stage, by using $scores_i^{c2}$. We do the same things by interchanging $c1$ and $c2$, and we merge the two results obtained. We propose to apply ranking stages by bins of size $binSize$.

Approaches combination We considered the fusion of the "baseline" and "two-steps ranking" as scoring method and the fusion of "baseline", "two-step ranking" and "the learning method" as an hybrid one.

We submitted three runs: P_A_Quaero-2013-P7_7: baseline P_A_Quaero-2013-P7_6: baseline + two-step ranking P_A_Quaero-2013-P5_5: baseline + two-step ranking + learning

Table 3: InfAP result and rank on the test set for all the 10 TRECVid 2013 evaluated concepts pairs

System/run	MAP	rank
Best submission	0.1616	1
P_A_Quaero-2013-P5_5	0.1266	6
p_A_Quaero-2013-P6_6	0.1205	7
P_A_Quaero-2013-P7_7	0.1205	8
Median submission	0.1126	10

Table 3 shows the performance of the three submitted variants. Our submissions ranked between 6 and 8 in a total of 20 for type A conditions. Our best submission ranked us as the fourth group out of 10 for the concept pair SIN task.

Our submissions were not completely finalized and the results, though quite good, are not fully representative. The single concept scores used used for the baseline and for the two-step ranking came from a hierarchichal fusion procedure which is similar to the one used in our best run in the main task but with not all the finally available descriptors and with less feedback iterations. They were not the finally best available ones. Similarly, the direct bi-concept learning was done in a different and less optimal way so direct comparison is not really meaningful.

It should be noted however that most of the value of the MAP come only from the AP of two concepts, "Chair+George_Bush" with an AP of about 0.7 and "Government_Leader+Flag" with an AP of about 0.45 while other pair have AP below or well below 0.05 as this can be shown in figure 6.

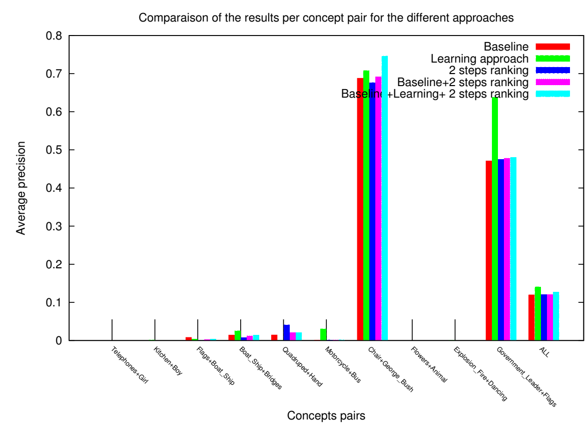


Figure 6: Comparaison of the results per concept pair for the different approaches, on TRECVID 2013 test corpus.

We made post-submission experiments with the final best available individual concept scores (M_A_Quaero-2013-3_3). We tried four single concept baselines as well as two simple score combination methods: score sum and score product. Results are given in table 4. The sum and product methods has a performance comparable to the performance of the best official submission. At the time of submission of the notebook paper, the computation for learning approach has not been completed and the two-step ranking gave results comparable to the sum combination. From experiments using only a subset of the available descriptors, the learning approach has a lower performance than the simple sum combination.

5 Acknowledgments

This work was partly realized as part of the Quaero Programme funded by OSEO, French State agency for innovation.

Results from the IRIM network were also used in these experiments [8].

Table 4: InfAP result and rank on the test set for all the 10 TRECVid 2013 evaluated concepts pairs

System/run	MAP
Single most frequent	0.1096
Single least frequent	0.1130
Single with best AP	0.1222
Single with worst AP	0.1004
Score sum	0.1613
Score product	0.1761

The authors also wish to thank Florent Perronnin from XRCE for providing descriptors based on classification scores from classifiers trained on ILSVRC/ImageNet data.

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] 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.

[3] S. Ayache, G. Quénot, J. Gensel, and S. Satoh. Using topic concepts for semantic video shots classification. In Springer, editor, *CIVR – International Conference on Image and Video Retrieval*, 2006.

[4] 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.

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

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

[7] 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.

[8] Ballas et al. IRIM at TRECVid 2013: Semantic Indexing and Multimedia Instance Search, In *Proceedings of the TRECVid 2013 workshop*, Gaithersburg, USA, 20-22 Nov. 2013.

[9] U. Niaz, M. Redi, C. Tanase, B. Merialdo, EURECOM at TrecVid 2012: The Light Semantic Indexing Task, In *Proceedings of the TRECVid 2012 workshop*, Gaithersburg, USA, 25-28 Nov. 2012.

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

[11] B. Safadi, S. Ayache, G. Quénot. Active Cleaning for Video Corpus Annotation. *International MultiMedia Modeling Conference,* 7131:518-528, Klagenfurt, Austria, jan 2012. Glasgow, Scotland, oct 2011.

[12] Bahjat Safadi, Georges Quénot. Descriptor Optimization for Multimedia Indexing and Retrieval. *CBMI 2013, 11th International Workshop on Content-Based Multimedia Indexing,* Veszprem, HUNGARY, jun 2013.

[13] Abdelkader Hamadi, Georges Quénot, Philippe Mulhem. Conceptual Feedback for Semantic Multimedia Indexing, *CBMI 2013, 11th International Workshop on Content-Based Multimedia Indexing,* Veszprem, HUNGARY, jun 2013.

[14] Milind Naphade, John R. Smith, Jelena Tesic, Shih-Fu Chang, Winston Hsu, Lyndon Kennedy, Alexander Hauptmann, and Jon Curtis. Large-scale concept ontology for multimedia. *IEEE Multimedia*, 13:86–91, 2006.

[15] Georges Quénot, Daniel Moraru, and Laurent Besacier. CLIPS at TRECvid: Shot boundary detection and feature detection. In *TRECVid'2003 Workshop*, Gaithersburg, MD, USA, 2003.

[16] Emine Yilmaz, Evangelos Kanoulas, and Javed A. Aslam. A simple and efficient sampling method for estimating AP and NDCG. In *SIGIR*. ACM 978-1-60558-164-4/08/07, July 2008.

[17] P. Over, G. Awad, J. , B. Antonishek, M.2Michel, A. Smeaton, W. Kraaij, and G. Quénot, TRECVid 2013 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms, and Metrics In *Proceedings of the TRECVid 2013 workshop*, Gaithersburg, USA, 20-22 Nov. 2013.

[18] Christian X. Ries and Rainer Lienhart. Deriving a discriminative color model for a given object class from weakly labeled training data. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, ICMR '12, pages 44:1–44:8, New York, NY, USA, 2012. ACM.