MRIM-LIG at ImageCLEF 2009: Photo Retrieval and Photo Annotation tasks

Philippe Mulhem, Jean-Pierre Chevallet, Georges Quénot, Rami Al Batal
Grenoble University, LIG
Philippe.Mulhem@imag.fr, Jean-Pierre.Chevallet@imag.fr
Georges.Quenot@imag.fr, Rami.Albatal@imag.fr

Abstract

This paper describes the different experiments that have been conducted by the MRIM group at the LIG in Grenoble for the ImageCLEF 2009 campaign. The group participated in the following tasks: Image Retrieval and Image Annotation. For the Image Retrieval task, we submitted runs with both text and image features, and a diversification process was applied. For the Image Annotation task, we used several features and classifiers in a way to generate keyword descriptions. For these two tasks, the results obtained are above the average of the participants.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software

General Terms

Image Retrieval, Image Indexing

Keywords

diversity algorithm, early fusion, late fusion, SVM

1 Introduction

We describe here the different experiments that have been conducted by the MRIM group at the LIG in Grenoble for the ImageCLEF 2009 campaign. This papers describes our participation in the following tasks : Image Retrieval and Image annotation. For Robot vision task, we used a language model to represent the image content; For the Image Retrieval track, we submitted runs with both text and image features, and a diversification process was applied; For the Image Annotation task, we used several features and classifiers in a way to generate keyword descriptions. We also participated in the Robot Vision track, describes elsewhere in the CLEF working notes. This paper is organized as follows. In section 2 we present our approach and results for the Image Retrieval track, and in section 4 we present our work for the Image Annotation track. We conclude in section 5.

2 Image Retrieval track

3 Photo Retrieval Task

3.1 Managing Diversity

This year again on the photo retrieval task, the focus was on managing results diversity [5]. CLEF2009 queries includes clusters that can be used to add and also to measure diversity of the retrieved documents. For example, the first query is only one term long: `"leterme"`, which is a proper noun.

```
<top>
<num> 1 </num>
<title> leterme </title>
<clusterTitle> yves leterme </clusterTitle>
<clusterDesc> Relevant images contain photographs of Yves Leterme. Images of
 Leterme with other people are relevant if Leterme is shown in the foreground.
 Images of Leterme in the background are irrelevant. </clusterDesc>
<image> belga28/05980958.jpg </image>
<clusterTitle> leterme albert </clusterTitle>
<clusterDesc> Relevant images contain photographs of Yves Leterme and King Albert II.
 Images with only one of them are considered to be irrelevant. </clusterDesc>
<image> belga27/05960161.jpg </image>
<clusterTitle> leterme -albert </clusterTitle>
<clusterDesc> Images which contain photographs of Leterme which are not part of
 the above categories are relevant to this cluster. </clusterDesc>
<image> belga32/06229323.jpg </image>
</top>
```

Three clusters are proposed for this query: `"yves leterme"`, `"leterme albert"` and `"leterme -albert"`. These clusters are supposed to be different possible classes of answers. The goal is then to retrieve a document's list that maximizes diversity measured using these thematic clusters. One can also note that some terms may be negated.

In the experiment we propose here, we have used clusters as "sub-query" (we called "cluster query") for the categorization of each main query answer. In this example, `"leterme"` is the main query, but we also run the 3 cluster queries. We consider only the output of the main query to be diversified according to the "cluster queries" (or sub-queries). The diversification consists in reordering the main query output list using the results of the cluster queries. In fact we consider that document lists from cluster queries are document classifications against each cluster.

In practice, each cluster query is passed through the IRS (see figure 1). Results of all these cluster queries are not fused with the main query, but they are used to categorize the document result of the main query. In fact for each document answered for the main query, we identify clusters where this document is an answer (box "Add cluster" in fig. 1). When several clusters are possible, we have decide to select the one with the lowest RSV value for this document. We could also keep all possible clusters. A special cluster (id=0) is created for document not found in any clusters. This adds to each document of the original main answer list, a cluster identification that is used for the diversification algorithm described in the next part.

3.2 Diversification Algorithm

We propose an algorithm to diversity a list of document associated to a cluster information (a cluster id). We make the hypothesis that each document answered to a query, is associated to a limited number of clusters. The shuffle algorithm uses only this information to propose more diversity to the result. The idea is to find a balance between the diversity of cluster among the

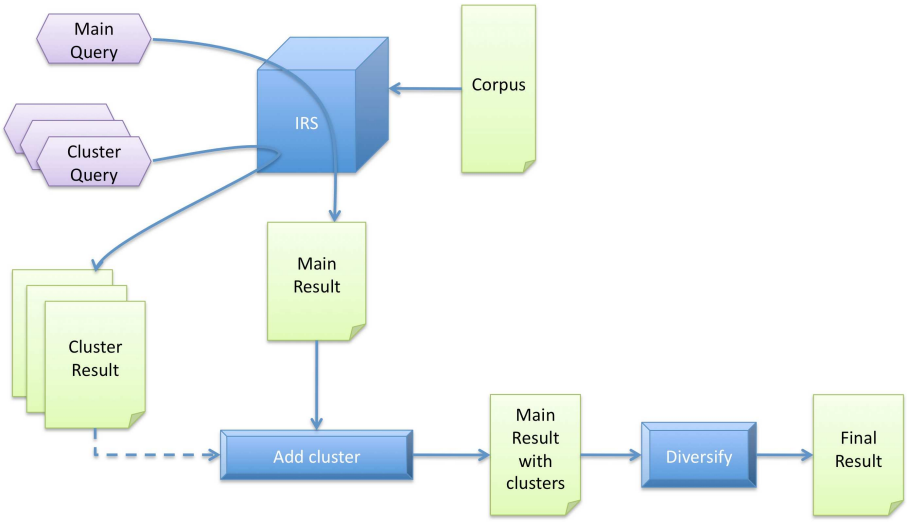

Figure 1: Diversification with cluster framework

answers, and the documents sorted by relevance status values (RSV). Hence, the shuffle algorithm should not change the original order "too much".

To implement this notion, we propose to set an absolute limit to a document move : the value r is the maximum rank a document can jump to be closer to the top. We propose also a damping parameter α which acts as a memory of clusters previously seen. Our shuffle algorithm implements the maximum rank constraint r by maintaining a queue of size r documents. This algorithm work in two stages (see fig. 2): the best document is selected from the queue, from the top, and according to the least cluster seen. This is the first document from the top of the queue that have the smaller "cluster seen" value. This document is then extracted from the queue, and added into the final order document list. Also the cluster of this selected document is added to the seen cluster structure, and the next document is push into the queue. The algorithm can be formalized as:

for all d in orginalDocumentList **do**
  ds ← topOf(queue)
  for all dq in queue **do**
    if clusterSeen(cluster(dq)) < clusterSeen(cluster(ds)) **then**
      ds ← dq
    end if
  end for
  updateSeenCluster(cluster(ds))
  push(ds,newDocumentList)
  push(d,queue)
end for

This algorithm has an extra step to process all documents that remain in the queue when the original list has all been processed. The "clusterSeen" structure associates for each cluster c a "novelty" value n_c from 0 to 1. The larger this value, the more recent a cluster has been seen. This new novelty value n'_c value is updated by the `updateSeenCluster()` function using:

$$n'_c = \begin{cases} (n_c + 1) * \alpha & \text{if pushed document belongs to cluster } c \\ n_c * \alpha & \text{otherwise} \end{cases} \quad (1)$$

The α value in $]0;1[$ acts as a damping factor that slowly reduces the cluster novelty n_c, if this cluster is not seen. A high value of α (close to 1) slows down the damping as a low value (close to

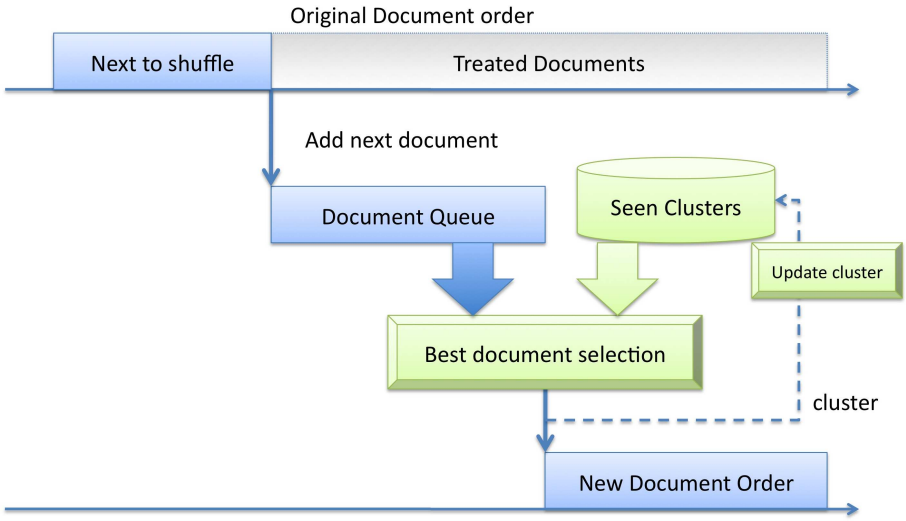

Figure 2: Diversification Algorithm

zero), produces a sharp release and tend to force this cluster to be used more quickly. Each cluster could have they own damping factor α_c. For example, value can be set according the the cluster size. We decide to set the same value for all clusters because we do not have enough information to differentiate clusters.

3.3 Adding diversity clusters

Only half part of the queries have identified clusters. The rest of CLEF queries are only composed by a simple query. We propose to automatically build cluster queries so that we can apply the same diversification technic to all queries. If we want to diversify answers using textual queries, then only terms belonging to the corpus are useful. So an efficient technic seems to be a diversification using terms that appears within the retrieved document using the original query. There exists several potential technics to do this clustering based on terms:

- Document clustering: Document clustering is a classical technic to structure a corpus. Clustering is usually perform on document vectors, using term vector space. Document clustering can be compute at indexing time, on the whole corpus, or at querying time, only on retrieved documents. Clusters can then be used for diversification.
- Term clustering: this is the reverse of document clustering. The clustering is performed on terms, using documents vector space. A similar technics is based on term collocation or cooccurrencies. Collocation refers to a short window term, where two terms appears, as cooccurency is usually performed on the whole document content and is equivalent to term clustering using document vectors.

It is difficult to predict which clustering technics is best for producing diversity. We have decided to experiment automatic generation of cluster sub queries, so to have an uniform query description, consistent with the first half part of the queries. That is the reason why we have adopted term clustering instead of document clustering.

Finally, we have not computed real term cluster, but only term neighborhood because we wanted to automatically produce automatically cluster sub queries with a similar syntax to those manually provided. Hence, k (k=100) nearest term neighbor has been computed on the document space, using just the frequency weighting. The following is an example of the neighborhood of the term “madonna” computed using the Euclidean distance.

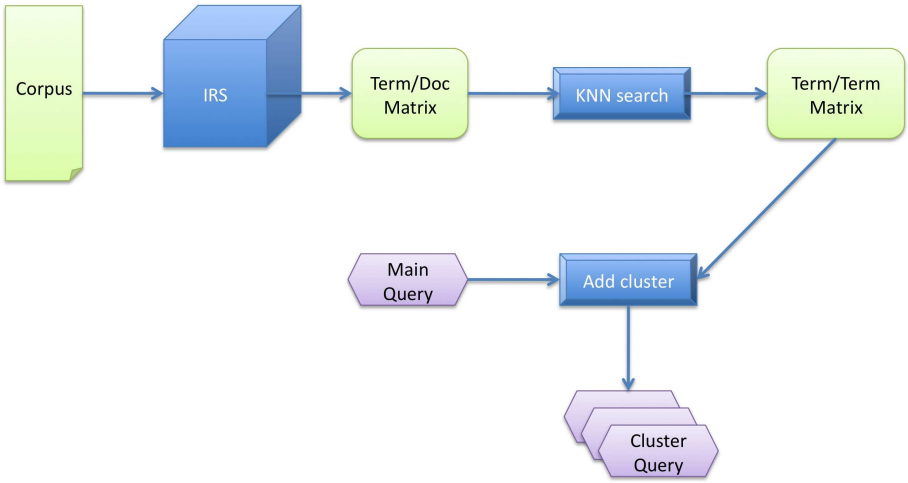

Figure 3: Building of text query clusters

```
<vector id="madonna">
<c id="campiglio" w="401"/>
<c id="evita" w="647"/>
<c id="fabbiani" w="659"/>
<c id="wrooom" w="668"/>
<c id="adig" w="673"/>
<c id="canalon" w="673"/>
<c id="miramonti" w="673"/>
<c id="yarnwind" w="674"/>
<c id="sgp" w="679"/>
<c id="girli" w="679"/>
<c id="drumlanrig" w="680"/>
<c id="u85" w="681"/>
<c id="badoer" w="682"/>
<c id="instor" w="682"/>
<c id="bbc1" w="683"/>
<c id="trentino" w="683"/>
<c id="szent" w="684"/>
<c id="gemaeldegaleri" w="684"/>
<c id="ritchi" w="684"/>
```

The neighborhood of a term, computed on its document distribution, are terms that appears in a similar set of documents. We have make the assumption that these terms are interesting to produce diversity. For example, we have automatically build 10 cluster queries by adding the 10 closest term to `euro`:

```
<top>
<num> 36 </num>
<title> euro </title>
<cluster>
<num> 36:1 </num>
<clusterTitle> euro currenc </clusterTitle>
</cluster>
<cluster>
<num> 36:2 </num>
```

```
<clusterTitle> euro banknot </clusterTitle>
</cluster>
<cluster>
<num> 36:3 </num>
<clusterTitle> euro coin </clusterTitle>
</cluster>
<cluster>
<num> 36:4 </num>
<clusterTitle> euro starter </clusterTitle>
</cluster>
<cluster>
<num> 36:5 </num>
<clusterTitle> euro ocotob </clusterTitle>
</cluster>
<cluster>
<num> 36:6 </num>
<clusterTitle> euro tender </clusterTitle>
</cluster>
<cluster>
<num> 36:7 </num>
<clusterTitle> euro switchov </clusterTitle>
</cluster>
<cluster>
<num> 36:8 </num>
<clusterTitle> euro baht </clusterTitle>
</cluster>
<cluster>
<num> 36:9 </num>
<clusterTitle> euro changeov </clusterTitle>
</cluster>
<cluster>
<num> 36:10 </num>
<clusterTitle> euro turnov </clusterTitle>
</cluster>
<cluster>
<num> 36:11 </num>
<clusterTitle> euro  -currenc -banknot -coin -starter -ocotob -tender -switchov -baht -changeov
 -turnov </clusterTitle>
</cluster>
```

3.4 Visual queries

In this CLEF track, image are only associated to sub-queries that represent clusters. For image queries, we have used standard color histograms (512 bins, RGB color space) and computed RSV using Jeffrey divergence similarly to what was described in [3]. This produces for each queries, a total order on the collection. We found that this computation is too weak semantically to produce accurate results. We have then decided to filter these results using the output of the main text queries. As a consequence, only images that are also answer to the textual queries are kept. This is a strong filter, but according to previous work like like [7], such process leads to good results.

Hence, visual is used in two different way: first as the main result of the query with a text filter, and second to build cluster information used to diversify the output of the main textual query. We explain these two approaches in the following.

3.5 Visual and textual diversity

Given a query, we have two types of cluster information: the textual sub-queries that form textual query clusters, and the visual sub-queries that form visual query clusters. We consider that these two clusters information have equivalent importance for the diversification and both cluster informations (id) are added to each answer of the main textual query.

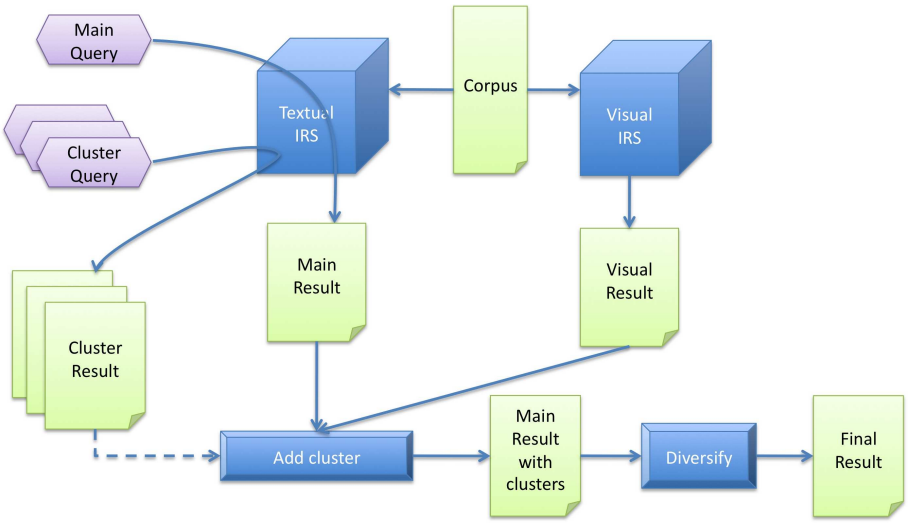

Figure 4: Textual and visual cluster to diversify main query

Then an adapted version of the diversification algorithm is used ("Add Cluster" in Figure 4) that accept each document to be associated to more that one cluster: one from text and one from image.

3.6 Submitted runs and official results

We have submited four runs:

LIG1_T_TXT : This run uses only the title of textual queries. This is a basic run using Divergence from Randomness weighting(see 3.7 for indexing details), without any diversification processing. Based on the precision at 10 documents (0.7780) and the cluster recall at 10 documents (0.5416), this run get a F-measure value of 0.6386, in rank 39 on 84 submitted runs: it means that this basic indexing produces already a good diversification for the top 10 retrieved documents. The MAP (Mean Average Precision) is quite high (0.5063), but it is not strongly correlated with the cluster recall and precision at 10 documents: among the five best results of this year one has a MAP of only 0.0747 but with a F-measure of 0.7580! This run is considered as a baseline for our submissions. We may however notice that, when considering only the title of the queries, our run is in fourth rank on 14: so no special treatment for diversification and using only the title of the query gives already decent results. This run is in position 4 over our 4 runs. It shows that clusters informations, diversification algorithm and visual aspect have a positive impact on diversification at 10 documents.

LIG2_T-CT_TXT : This run is a diversification of the previous run using the proposed textual clusters, and the one we have generated. The size of the queue in the diversification algorithm is set to 20 and the damping factor is 0.5. This run obtained similar MAP (0.5064) to the previous one. This is normal as only the order is changing with a maximum windows of 20 documents. There is no change is cluster recall at 10 (0.5416), and a non significant increase on cluster precision at 10 (0.7800). We can conclude that our diversification algorithm does

not have a significant impact in the 10 first retrieved documents. Even, our proposal to create cluster query by adding top 10 similar terms using document vector has no impact on the results. We think that further studies have to be conducted to find better parameter values for the diversification algorithm.

LIG3_T-I_TXT-IMG : This is a visual run with a filtering using textual output of the IRS (run 1). We have guessed that a visual run only using color histogram is no enough meaningful. That is the reason why we filter the visual output with the result of the textual queries. A visual query is a set of cluster images. The output is a concatenation of each sub query clusters. Because of this raw cluster result concatenation with a maximum of 1000 answers, we are forced to perform a diversification with a larger queue of 1000 with a damping factor of 0.5. We have to investigate if this larger queue value is related to the good result of the top 10 cluster diversification. This is our best run with the global rank 20 (on 84), with a F-Measure of 0.6975. There is a decrease in the MAP (0.4337), first because filtering by intersection first 1000 image answers and textual answers reduces the number of retrieved documents (from 19066 for run 1 and 2 to 18375), and second the precision at 1000 is also reduced (from 0.3813 to 0.3750). Considering the first 25 queries, our approach was ranked 39, and considering the second part of the query set, the rank is 6th, and the 4th one considering only the query title. We have no clear explanation of this difference between the first and second queries. We can conclude that the filtering of a simple image matching with text using our diversification algorithm with a very large queue is our best choice. We have to further investigate the real influence of the diversification algorithm and the mutual influence of text and images.

LIG4_T-CT-I_TXT-IMG : Textual queries are diversified both using textual given and constructed sub-query clusters, and also by the visual sub-queries clusters. This run uses the textual run 2 with the generated textual clusters. This run considers then both text and image results in a late fusion schema. For the whole set of queries, this run is at rank 32. We may notice that, even if the mean average precision (map) is not the official measure for this competition, this run obtained the higher value for map with 0.5055, it seems that this approach tends to favor the average precision than the early results. In conclusion, the building of textual clusters using term/term distance is good solution to increase the map (from 0.4524 to 0.5221 for the second 25 queries), but it is not effective for the the cluster recall at the top values (from 0.6365 to 0.5696 at 10 documents), even if at 1000 document it is slightly better than the previous run (from 0.9477 to 0.9677).

3.7 Technical details and problems

The textual search engine used during this task is our system XIOTA [2]. The weighing scheme is the Divergence from Randomness [1]. Standard stop-word removal and Porter stemming were applied. When using the textual search engine, we have always limited the size of the answer to 1000 documents.

The term expansion is computed on stemmed terms, so there is an inconsistency, because stemmed terms are added with non stemmed terms of the original query. Hence expended terms and stemmed twice. That may cause a recall problem because these terms are not consistent with the rest of the collection.

Term negation is treated using a negative weight into the query vector.

Because of the new ordering, original RSV values are no longer meaningful and are replaced by a fake values.

In the run 3, we have diversified the raw output of the visual IRS. The problem is that visual output are sorted by cluster first and by IRS second. This imply that the diversification algorithm should have a queue greater than the largest cluster output (1000 documents in our case). A better solution should be to first sort each visual query answer, only by IRS only, before doing the diversification on clusters. We had also a last minute problem: keeping the initial raw output

of the visual system, implies possible duplication of answer in each cluster answer. This also can be solved by sorting the visual result by IRS, and be keeping only the lowest IRS value when the same image appears in several clusters answers.

4 Image Annotation track

The Image Annotation track focuses on the evaluation of state of art automatic image annotation approaches. The whole image set is composed of 5,000 images as the learning set with manually annotated images, and 13,000 images as the test set. The number of annotation concepts is 53, and these concepts are organized in a (simple) hierarchy, representing for instance the four seasons (winter, summer, spring, autumn) in an exhaustive way. The evaluation uses Equal Error Rate measure as well as a measure that considers the hierarchy of concepts to compute the quality of the systems [4]. We describe now our approach, the submitted runs and the results obtained.

4.1 Our approach

Our idea was to test early fusion of features and late fusion of concept detectors. We studied the use of color features (RGB and HSV histograms), and texture features (SIFT points). For the concepts detectors, we used the well known support vector machines approach. We used a "one against all" (OAA) framework (all positive versus all negative samples). Because of the small size of positive (resp. negative) samples compared to negative (resp. positive) ones, we used one more complex approach: Consider one concept C having p_c positive samples, and n_c negative samples (ni = 5000 - pc >> pc). We define N_c SVMs with all the positive samples and 2*p_c negative samples, so that union the negative samples of all the SVMs cover all the p_c negatives samples of C. Each of these SVM learns the probability of belonging to each class concept/non concept. For one concept, we sum-up then the results for all the N_C SVMs. We call this framework MOAA (Multiple One Against All). The MOAA approach is strongly inspired from the work of Tahir, Kittler, Mikolajczyk and Yan called Inverse Random Under Sampling [6]. We chose also to fuse all the results obtained (late fusion) by selecting the best detector for each concept.

As a post processing, we applied on all our different runs a linear scaling in a way to fit the learning set a priori probabilities. We took afterward into account the hierarchy of concept in the following way: a) when conflicts occur (for instance the tag Day and the tag Night are associated to one image of the test set), we keep unchanged the larger value tag, and we decrease (linearly) the value all the other conflicting tags, b) we propagated the concepts values in a bottom-up way if the values of the generic concept is increased, otherwise we do not update the pre-existing values.

4.2 Submitted runs

In the LIG part of the AVEIR submitted runs, we used three sets of features:

- The first one, namely RGB, is only based on colors vectors represented in RGB color space. We split each image in three horizontal stripes (same height of 1/3 of the image height, whole width of the image). For each stripe we extract a simple RGB histogram (uniform sub-sampling) of 512 bins ($= 8 \times 8 \times 8$), and we normalize it. We concatenate then these three histograms into a 1536 bins histogram.

- The second set of features we used are SIFT (Scale Invariant Feature Transform). These features were extracted using a software provided by K. van de Sande [8]. SIFT features were extracted from regions selected according to Harris-Laplace feature points detection. Each feature is a 128-dimensional vector. A visual vocabulary containing 4000 dimensions was then generated using the SIFT features of the learning set, leading to a 4000 dimensions vector for each image.

- The third set of features, called RGBGAB, comes from an early fusion of color and texture features. We used a 64 dimensions RGB color histogram concatenated with a 40 dimensions

vector describing Gabor filters energy (8 orientations× 5 scales). The result is then a 104-dimensional vector.

The learning of the concepts was based on SVMs (Support Vector Machines), which proved their strength in the domains of learning and image annotation [ref]. For the RGB, SIFT, and the RGBGAB features we used the "one against all" approach using Radial Basis Function (RBF) as SVM kernel. The SVM learns the probability for one sample of belonging to each concept. For the SIFT and HSBGAB features, we used additionally MOAA SVM learning process as described previously. Using the rescaling and the hierarchy propagation, the results (average equal error rate) achieved on the test set are presented in Table 1. These results show that, on the learning set, the SIFT MOAA (Id=3) and the RGBGAB OAA (Id=4) give the best results. The fusion selecting the best detector for each concept leads to a EER of 0.26% .

Id	Features	Learning framework	Average EER
1	RGB	OAA	0.480
2	SIFT	OAA	0.372
3	SIFT	MOAA	0.340
4	RGBGAB	OAA	0.339
5	RGBGAB	MOAA	0.371
6	all	selection best	0.256

Table 1: EER results for different features and learning frameworks

4.3 Official results obtained

Table2 presents the official results that we obtained. This table also presents the average results over all the runs submitted by the participants. The four runs officially submitted correspond to the SIFT OAA (run id. 50_2_1245573948330.txt), SIFT MOAA (run id. 50_2_1245574356224.txt), RGBGAB OAA (run id. 50_2_1245574874836.txt) and Fusion (cf. id 6 in Table 1, run id. 0_2_1245578233616.txt). As expected, the EER values for the submitted runs on the test set are greater than on the test set, and if we rank these runs they do not behave as on the test set regarding the EER values:

- the late fusion of the runs still provides our best results for EER (11th site on 18, EER of 0.384 when the average for the participants is 0.390), but it is very close to the SIFT MOAA run.
- the RGBGAB OAA run achieves almost similar EER than the SIFT OAA run.

If we focus now on the evaluation of the recognition based on concept hierarchy (not available on the test set), then the RGBGAB OAA achieves our best result (recognition rate of 0.74, position 27 on 74 runs submitted, 10th site on 18), before the SIFT MOAA, the SIFT OAA and eventually the Fusion run. It seems so that our fusion run gives good results, but when an annotation concept is false is not close (in the hierarchy) to a relevant concept. Conversely, the HSVGAB OAA provides less accurate concepts, but when the concept is wrong then is it more often close in the hierarchy to a relevant concept compared to the fusion run.

The results obtained for annotation are the given in Table 2.

5 Conclusion

In this section, we conclude for the two tracks in which runs where submitted by the MRIM research group from the LIG.

For the Image Retrieval task, we studied the use of a new diversification algorithm and the automatic building of sub queries using term distance, for diversification. Results obtained with

official id	Features	Learning framework	Average EER (/74) (avg. 0.390)	Hierarchical recognition (/74) (avg. 0.619)
50_2_...8330.txt	SIFT	OAA	0.443 (42)	0.680 (36)
50_2_...6224.txt	SIFT	MOAA	0.392 (36)	0.714 (34)
50_2_...4836.txt	RGBGAB	OAA	0.440 (40)	0.741 (27)
50_2_...3616.txt	all	selection best	0.384 (34)	0.591 (45)

Table 2: EER and hierarchical results for different features and learning frameworks

our diversification algorithm are not enough conclusive, and we still have to study the behavior of this algorithm and it effective influence on diversification. Also the automatic building of sub queries using distance among terms has no clear positive effect. Again, we have to further experiment with and without this extension. Our best run (rank 20) integrates textual and visual matching: we process the image queries and we keep the images that are relevant according to the text queries. We uses Divergence form Randomness weighting for text. We obtain overall very good results according the MAP, but average results using the top 10 cluster ranking. We underline here that this evaluation measure proposed is based only on the first 10 documents, and is largely uncorrelated to the MAP (the usual evaluation measure for information retrieval systems) with a correlation value of 0.45 [COMMENT TU CALCULES CELA ?]. What is happening in fact is: a) a low map value does not implies a low F-measure for diversity, and b) a medium or high value of map (for instance higher than 0.3) has good chances to have a medium to high value for the F-measure. For instance, the top 5 run have a F-measure of 0.76 and a map of 0.07, which means that this system is not a good information retrieval system according to an information retrieval standard evaluation measure, but has a top evaluation for this task. Considering the task from a news context, we consider that using only the 10 first documents result for such evaluation is probably too constrained to reflect the real behavior of a system.

For the image annotation track, we proposed to study different learning approaches. Early fusion used RGB color space and gabor features. We also extracted SIFT features and used simple one against all and multiple one against all learning techniques. A late fusion was defined as a selection of the best learning configuration for each concept to be detected.

The results that we obtained show that fusing several approaches (early and late fusion, simple or multiple one against all learning technique) leads to higher results for the EER (our best rank is 34 on 74 runs). We used for all our runs the concept hierarchy to avoid conflicts. However, our best result according to the hierarchical recognition rate was obtained using an early fusion and simple one against all learning technique (rank 27 on 74). The concept hierarchy was quite simple, but allowed to study in a real case how to manage such data. In the future, we will study further the behaviour of the different learning approaches in a way to increase the quality of our results, and we will also explore different uses of the hierarchy during the learning and recognition steps.

Acknowledgment

This work was partly supported by: a) the French National Agency of Research (ANR-06-MDCA-002), b) the Quaero Programme, funded by OSEO, French State agency for innovation and c) the Région Rhones Alpes (projet LIMA).

References

[1] Gianni Amati and Cornelis Joost van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transaction on Information Systems*, 20(4):357–389, October 2002.

[2] JeanPierre Chevallet. X-iota: An open xml framework for ir experimentation. *Lecture Notes in Computer Science (LNCS), AIRS'04 Conference Beijing*, 3211:263–280, 2004.

[3] Loic Maisonnasse, Philippe Mulhem, Eric Gaussier, and Jean-Pierre Chevallet. Lig at imageclef 2008. In *Evaluating Systems for Multilingual and Multimodal Information Access – 9th Workshop of the Cross-Language Evaluation Forum*, 2008.

[4] Stefanie Nowak and Peter Dunker. Overview of the clef 2009 large scale - visual concept detection and annotation task. In *CLEF working notes 2009*, Corfu, Greece, September 2009.

[5] Monica Paramita, Mark Sanderson, and Paul Clough. Diversity in photo retrieval: overview of the imageclefphoto task 2009. In *CLEF working notes 2009*, Corfu, Greece, September 2009.

[6] Muhammad Atif Tahir, Josef Kittler, Krystian Mikolajczyk, and Fei Yan. A multiple expert approach to the class imbalance problem using inverse random under sampling. In *Multiple Classifier Systems*, pages 82–91, Reykjavik, Iceland, June 2009.

[7] Sabrina Tollari, Philippe Mulhem, Marin Ferecatu, Hervé Glotin, Marcin Detyniecki, Patrick Gallinari, Hichem Sahbi, and Zhong-Qiu Zhao. A comparative study of diversity methods for hybrid text and image retrieval approaches. In *Evaluating Systems for Multilingual and Multimodal Information Access – 9th Workshop of the Cross-Language Evaluation Forum*, 2008.

[8] Koen E. A. van de Sande, Theo Gevers, and Cees G. M. Snoek. Evaluation of color descriptors for object and scene recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, Alaska, USA, June 2008.