XRCE's Participation to ImageCLEF 2008

J. Ah-Pine, C. Cifarelli, S. Clinchant, G. Csurka and J.M. Renders
Xerox Research Centre Europe, 6 ch. de Maupertuis, 38240 Meylan, France
jean-michel.renders@xrce.xerox.com

Abstract

This year, our participation to ImageCLEF 2008 (Photo Retrieval sub-task) was motivated by trying to address three different problems: visual concept detection and its exploitation in a retrieval context, multimedia fusion methods for improved retrieval performance and diversity-based re-ranking methods. From a purely visual perspective, the representation based on Fisher vectors derived from a generative mixture model appeared to be efficient for both visual concept detection and content-based image retrieval. From a multimedia perspective, we used an intermediate fusion approach, based on cross-media relevance feedback that can be seen as a multigraph-based query regularization method with alternating steps. The combination allowed to improve both mono-media systems by more than 50% (relative). Finally, as one of main goals of the organizers was to promote both relevance and diversity in the retrieval outputs, we designed and assessed several re-ranking strategies that turned out to preserve standard retrieval performance (such at precision at 20 or mean average precision) while significantly decreasing the redundancy in the top documents.

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; H.3.7 Digital Libraries; H.2.3 [**Database Managment**]: Languages—*Query Languages*

General Terms

Measurement, Performance, Experimentation

Keywords

Cross-media information retrieval, Trans-media relevance feedback, diversity-based re-ranking

1 Introduction

In ImageCLEFphoto 2007, the issue we wanted to address was the semantic gap between text and image in retrieval tasks. Indeed, the challenges proposed by ImageCLEF these last years and particularly in the context of the Photo retrieval task, are well-adapted to investigate that kind of problems as we have at our disposal images that are particularly well-described from a textual point of view. In the photo retrieval task, the multi-media objects are constituted of some text and some image that correspond to two different kinds of information from a semantic point of view. Thus, the scientific challenge was to exploit in an efficient manner these two types of information in order to improve the search results.

In last year's session, we addressed the question of combining textual and visual information by developing an intermediate fusion approach which allows to go further than late or early

fusion methods [4]. The results we obtained showed that our proposal allows to outperform either monomedia-based retrieval or classical late or early fusion-based retrieval.

In ImageCLEFphoto 2008, however, the main goal is to promote diversity among the first search results. We address this issue using a two-step approach. The first step is to ignore the question of diversity. In other words, we first try to find the most relevant objects using the material introduced in [4]. Then, in a second step, we re-rank the first relevant objects by taking into account their mutual similarities in order to avoid redundancy and thus to promote diversity.

The present working note explains the different approaches that we investigated for designing and developing methods that tackle the challenges proposed in ImageCLEFphoto 2008. This was an opportunity to both enhance the system we developed last year and to implement methods which aims at incorporating diversity into the search results.

The document is organized as follows.

Section 2 is devoted to standard retrieval based on the textual part of the photos. In particular, the aim of the experiments reported in this section was to use external resources for enriching the textual description of both the objects and the queries. Two kinds of resources were used: the first one is rather linguistic and consists for example, in adding synonyms of terms; the second one is rather "cross-media oriented" as it aims at adding visual concepts obtained after processing the images. Such visual concepts are typically the categories found by an image categorizer system.

In section 3, we detail the different experiments we made in the context of image retrieval and classification. First we briefly remember the representation of images with Fisher Vectors (subsection 3.1). In subsection 3.2 we describe our image categorization method used in the ImageClef Visual Concept Detection subtask and in subsection 3.3 our image retrieval system applied to the ImageCLEFPhoto task is detailed. In subsection 3.4 we propose different strategies to incorporate diversity (i.e. to reduce redundancy) in the top retrieved photos, and in the last subsection 3.4.3, we give some further precisions concerning the submitted AUTO-IMAGE runs.

In section 4, we present the system we developed for multi-media information retrieval. First, we briefly recall in subsection 4.1 the approach we defined in [4] concerning intermediate fusion between different mono-modal sources. Second, in subsection 4.2, we explain the different methods we implemented for promoting diversity into the multi-media search results, and in subsection 4.2.3 we present the characteristics of the different AUTO-TXTIMG runs we have submitted.

The last section is, as usual, devoted to a set of conclusions we can draw from the experimental results of ImageCLEF 2008.

2 Text Retrieval

One axis of research that we wanted to investigate in this session of ImageCLEF was the use of external resources (thesauri) for enriching the textual side of the objects (image description / queries) of the task. The idea is to exploit such resource to compensate for:

- the relative sparsity of the textual representation of the photos (even if, this year, the textual part of the photos included a more detailed description of its content);
- the gap between the lexical fields of these descriptions and the queries : queries are expressed in a more abstract way than the factual description of the photos.

Another issue that we wanted to address was the use of the visual concepts provided by the organizers as extra "textual words", refining the original textual representation of the photo by higher-level visual information.

As basic text retrieval engine, we used the LEMUR Toolkit for Information Retrieval [11], adopting the Language Modeling framework [16] to retrieve and rank relevant objects with respect to the textual part of the query. For each photo, the title, location and description fields were simply concatenated to form a single document; these documents were then linguistically pre-processed (lemmatization and stop-word removal), before being indexed by the LEMUR tool. Dirichlet smoothing [20] was applied in the retrieval phase, with a parameter which is directly

related to the average document length (this parameter choice strategy was also adopted when using enriched documents). Documents are ranked by decreasing similarity with respect to the query, the similarity being defined as the cross-entropy between the query model and the document (smoothed) language model.

The first variant we developed consisted in exploiting the English Open Office thesaurus[1] to enrich the textual description of the photos and/or the queries. Several strategies can be chosen. We chose the following ones:

- Document enrichment: we added all the synonyms and broader terms to the terms of the original description, when they are covered by some thesaurus entry; as we wanted to give more weight to the original terms, the latter ones are artificially replicated 15 times.
- Query enrichment: we added all the synonyms and narrower terms to the terms of the original description, when available; as we wanted to give more weight to the original terms, the latter ones are artificially replicated 5 times.

Note that we also simultaneously enriched both the queries and the documents, but this resulted in performance deterioration (too much noise introduced).

As pseudo-relevance feedback (PRF) is another way to do query expansion, we systematically ran experiments with and without pseudo-relevance feedback for each setting (baseline, document enrichment, query enrichment). The ten top terms of the ten top documents were used to expand the initial query language model by convex linear combination (coefficient =0.6 for the feedback model); query model updating was based on the mixture model method [19]. The performance (Mean Average Precision and Precision@20) is given in Table 1.

	Without PRF		With PRF	
Method	MAP	P20	MAP	P20
Baseline	0.215	0.259	0.239	0.293
Document Enrichment	0.231	0.268	0.260	0.308
Query Enrichment	0.218	0.264	0.257	0.282

Table 1: Performance (MAP and P@20) of different enrichment strategies

It clearly appears that combining document enrichment by thesaurus and query expansion by PRF (using the thesaurus-enriched documents in the first feedback phase) gives the best results. Doing query semantic enrichment followed by PRF (using the thesaurus-enriched query in the first feedback phase) gives slightly worse results. In any case, the use of this external resource is beneficial with respect to a standard PRF query expansion.

The second variant we developed aimed at assessing the benefits of introducing the Visual Concepts, as produced by two generic Visual Concept Detectors (we refer to the subsection 3.2 of this note and to the Visual Concept Detection sub-task of ImageCLEF 2008 for further information). These two detectors are designated by XRCE's and RWTH's detectors respectively. The approach we adopted was quite simple: we added to the original textual description the terms corresponding to the visual classes (e.g. indoor, outdoor, building, sky, night, animal, etc.) of the query images, if the associated confidence score was above some threshold (0.65 in our case; see subsection 3.2 for more details). We did the same for the photo annotations: both the queries and the documents of the collection were enriched by the concepts of the associated image(s). This could be considered as a very simplistic way of doing multi-media retrieval.

The performances (Mean Average Precision and Precision@20) are given in Table 2.

Clearly, the use of the visual concepts increases the retrieval performance, even if the inclusion of visual concepts is done in a rather simplistic way. But, to anticipate a little bit the results of the multi-media retrieval section, it will turn out that this advantage is lost when we use other, more complex multi-media fusion mechanisms, based on lower-level features than the visual concepts.

[1] Available on *http://wiki.services.openoffice.org/wiki/Dictionaries*

	Without PRF		With PRF	
Method	MAP	P20	MAP	P20
Baseline	0.215	0.259	0.239	0.293
XRCE Visual Concepts	0.241	0.297	0.269	0.334
RWTH Visual Concepts	0.232	0.271	0.258	0.308

Table 2: Performance (MAP and P@20) of the combination with different Visual Concept Detectors

3 Image Retrieval

In this section, we first describe briefly our image representation based on Fisher Vectors (for more details see [4, 15]). The same representation was used in the Visual Concept Detection sub-task (section 3.2) and ImageCLEFPhoto (section 3.3).

3.1 Fisher Vectors for Images

As image representation, we use the Fisher Vector as proposed in [15]. This is an extension of the bag-of-visual-words (BOV) representation; the main idea is to characterize the image with the gradient vector derived from the generative probability model (a visual vocabulary modeled by a GMM in our case). This representation can then be subsequently fed to a discriminative classifier for categorization, or used to compute similarities between images for retrieval.

The generative probability model in our case is the Gaussian Mixture Model (GMM) which approximates the distribution of the low-level features in images, where each Gaussian component can be seen as a visual word.

If we denote the set of parameters of the GMM by $\Phi = \{w_i, \mu_i, \Sigma_i,\ i = 1...N\}$ (w_i, being the mixture's weight), we can compute the gradient vector of the likelihood $\nabla_\Phi \log p(I|\Phi)$ that the image was generated by the model Φ. This gradient of the log-likelihood describes the direction in which parameters should be modified to best fit the data (image features). One of its advantages is that it transforms a variable length sample (number of local patches in the image) into a fixed length representation (which we will call Fisher Vector) whose size is only dependent on the number of parameters in the model ($|\Phi|$).

Before feeding these vectors to a classifier or computing similarities between images, each vector is first normalized using the Fisher Information matrix F_Φ (see [15] for the computational details):

$$\mathbf{f}_I = F_\Phi^{-1/2} \nabla_\Phi \log p(I|\Phi) \tag{1}$$

with

$$F_\Phi = E_X P \left[\nabla_\Phi \log p(I|\Phi) \nabla_\Phi \log p(I|\Phi)^T \right] \ .$$

and then re-normalized to have an L1-norm equal to 1.

Note that we used in our experiments two types of low-level local features extracted both on regular grids at different scales (see [4] for further details). The first feature set is based on local histograms of orientations (referred as texture) and the second one is based on local RGB statistics (referred as color). This led to two visual vocabularies, so that we have two Fisher Vectors per image.

3.2 Visual Concept Detection

The Visual Concept Detection task of ImageClef had the objective to identify visual concepts in users' photos [18]. The training images were labeled with 17 visual concepts: indoor, outdoor, person, day, night, water, road or pathway, vegetation, tree, mountains, beach, buildings, sky, sunny, partly cloudy, overcast and animal. The goal was to indicate the presence or the absence of these concepts.

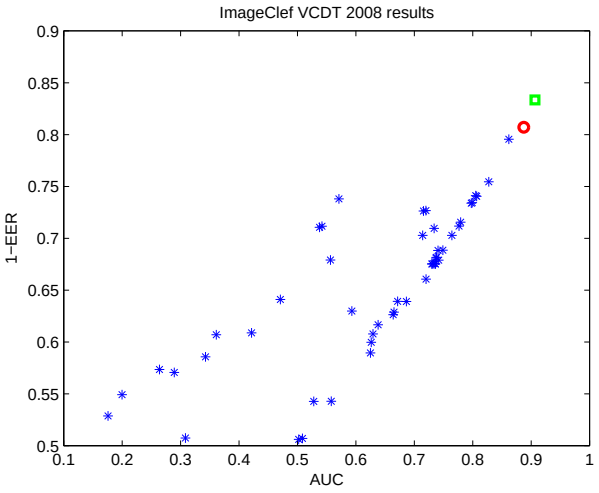

Figure 1: ImageClef Visual Concept Detection Task results. The red circle represents our linear system and the green square our non-linear system.

In spite of the fact that the concepts were presented in a hierarchy, we have not used it (neither in training nor in testing), but considered the task as a multi-class multi-label image categorization problem. Using the Fisher Vector image representation, we trained linear and non-linear (kernel based) classifiers, where each classifier was designed to detect the presence of one of the 17 concepts.

The binary one-against-all linear classifiers were trained directly on labeled normalized Fisher Vectors. As linear classifier, we used our own implementation of Sparse Logistic Regression (SLR) [10], i.e. logistic regression with a Laplacian prior.

In the case of non-linear (kernel-based) classifiers, we use the following kernel:

$$K(\mathbf{f}_I, \mathbf{f}_J) = \sum_{t=1}^{T} ||\tilde{\mathbf{f}}_I - \tilde{\mathbf{f}}_t||_1 \cdot ||\tilde{\mathbf{f}}_J - \tilde{\mathbf{f}}_t||_1 \tag{2}$$

where $||\cdot||_1$ is the L1-norm, $\tilde{\mathbf{f}}_i$ is $\mathbf{f}_i$ normalized to have an L1-norm equal to 1, and $\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_T\}$ are the Fisher Vectors of the training images. We can train a non-linear classifier (e.g. SVM) with this kernel; equivalently, we can compute first explicitly the projections of Fisher Vectors in the feature space defined by the kernel (2), i.e. the L1-distances with respect to the training examples, and then train a linear classifier in this new space (what we actually did, the linear classifier being SLR).

Note that, as we used two types of low-level local feature vectors (color and texture), we trained two classifiers for each concepts. We transformed the classifier scores s in probabilities with the sigmoid mapping $(1 + exp(s))^{-1}$. To decide about the presence or the absence of a concept, the mean of the two probabilities (color and texture) was compared to a threshold (0.65 in our case).

Both systems, the linear one and the kernel-based one, performed well compared to other systems (see figure 1), the kernel-based system slightly over-performing the linear system.

3.3 Content Based Image Retrieval

In the case of image retrieval, we use also the Fisher Vector representation of the images. To obtain the similarity between two images I and J, we first concatenate the color and the texture Fisher vectors before computing the L1-norm of the difference between the concatenated vectors:

$$sim(I, J) = norm_{max} - ||\mathbf{f}_I - \mathbf{f}_J||_1 = norm_{max} - \sum_i |f_I^i - f_J^i| \tag{3}$$

where f^i are the elements of the concatenated vector $\mathbf{f}$ and $norm_{max} = 2$.

Given a query image, we can simply rank the images of the database according to this similarity measure. Nevertheless, in the ImageClef Photo Retrieval Task, we have not one but three query images. Therefore, the question is how to combine them for a better retrieval. We investigated three different strategies:

- **I1** : We considered the mean of the three Fisher Vectors (this can be seen as the concatenation of the three image in a single one) and used this mean Fisher Vector to query the database.
- **I2** : The database images were ranked according to each of the three images independently and the three ranked list was combined using round-robin type selection (i.e. intermixing the three lists), of course eliminating the repetitions of the same images.
- **I3** : We combine the three similarity scores (with respect to each image of the query) by averaging the scores after normalisation (by studentization, a.k.a. z-scores).

Table 3 compares these three strategies. We can see that the early fusion (mean Fisher Vector) performs worse than late score level fusions. The best performance was obtained by the score averaging strategy **I3**.

Method	MAP	P20	CR20
I1 (xrce_img_ff)	0.119	0.255	0.330
I2 (xrce_img_rg)	0.130	0.301	0.351
I3 (xrce_img_of)	0.151	0.328	0.352

Table 3: Performances (MAP and P@20 and CR@20) of different strategies.

3.4 Incorporating Diversity

One of the aims of ImageCLEFPhoto 2008 is to promote both relevance and diversity into the first retrieved elements. Therefore, in order to introduce diversity at the top level (e.g top k=20 images), we try to find images that are representatives of groups of closely related images (assuming that they contain redundant information). The idea is to bring these representative images to the top of the list and to push down the images which are redundant with respect to images already selected.

To achieve the selection of such representative images, we tested two different strategies. The first one uses explicit image clustering (section 3.4.1) and selects in each cluster the most relevant image according to the original ranking described in section 3.3. The second approach is based on image density estimation (see sub-section 3.4.2), where a number of representative images are automatically detected by seeking for local maximum of the density function.

3.4.1 Explicit Image Clustering

In order to cluster the top N ranked images using the Fisher Vector representation, we proceed as follows. First, we build the K-Nearest Neighbors (K-NN) graph: nodes i and j are connected if the image I_j is amongst the K-NN neighbour of the image I_i based on the similarity measure defined by (3). Edges are weighted by the corresponding similarity measure. As the K-NN relationship is not symmetric, the weighted adjacency matrix is asymmetric and the resulting graph is directed.

Then, Probabilistic Latent Semantic Analysis [8] (PLSA) – some form of discrete Probabilistic Principal Component Analysis – is performed on the weighted adjacency matrix. Note that PLSA is known to be an efficient clustering method to handle sparse matrices (as in our case for K much smaller than N).

3.4.2 Density Based Representative Image Detection

As an alternative to explicit clustering, we experimented a second approach that relies on direct identification of peaks in some density function estimated from the top N images. Among different possible density estimation methods, we used a modified version of the method proposed in [6]. The main idea is to define the density value at each point (image in our case) I_i as the number of elements (images) that fall in the hypersphere with center in I_i and radius r. Once again, Fisher Vector representation and the L1-norm were used to define the distance between images.

Usually, the density in one point is estimated considering only a fixed radius. We propose here a variant that allows us to obtain a smoother density estimation that better fits our scope. This can be obtained, by considering not only one possible radius of the hypersphere but a predefined set of them $\{r_1, \ldots, r_R\}$ and compute the density by:

$$d(I_i) = \sum_{k=1}^{R} n(I_i, r_k)/V(r_k) \tag{4}$$

where $n(I_i, r_k)$ is the number of images that fall in the hypersphere of radius r_k with volume $V(r_k)$.

The local maxima of the density function are considered "modes" (i.e. local agglomeration of images that generate a local maximum in the density function) and used as representative images.

3.4.3 Description of AUTO-IMAGE runs

As mentioned above, to introduce diversity, we first rank the images according to one of the methods described in section 3.3 and then re-rank the list using either clustering or density estimation on the top N elements of this list.

As the original list is obtained as an answer to a query, the hope is that selected images represent groups of relevant images. Nevertheless this is not necessarily verified. Furthermore, if N gets larger, the probability to get groups of similar but non relevant images [2] in the top list increases. Therefore the N parameter was selected relatively small (N=100) trying to reduce maximally these cases.

- N being small, we had to use a small number of clusters (we used C=10) in the first diversification strategy. In each cluster we considered as representative image the one that had the best rank in the list **I3** (see section 3.3). The selected ten images were placed at the top of the list keeping the order of the other images unchanged. The name of this run was xrce_img_cl10_of.

- Using a small C allowed to diversify only the top C=10 elements. Of course we can increase this number by increasing C and N, but as we have not submitted these runs, we have no result for them. As an alternative, we experimented another strategy which is based on the combination of l ranked lists (l is typically 2 or 3 in our particular setting). The main idea is to consecutively cluster the top N elements of each of the lists and concatenate the $C' <= l \cdot C$ representative images after eliminating images that occurred more than one time. For example we cluster the top N images in **I3** and **I2** and concatenate the 20 images selected as above in the 2x20 clusters. If the same image was selected twice, we consider it only once. We ranked them according to their fused similarity score (used to obtain list **I3**) and place them at the top. The rest of the images in the list **I3** is placed after them. This run was submitted under the name xrce_img_cl10_of_rg. Similarly, the runs xrce_img_cl10_ff_of and xrce_img_cl10_of_ff_rg were obtained by this strategy.

- In the case of density-based representative image detection, we again used only the top N=100 images which led to relatively few (4-5) local maxima in the density function. Therefore, we consecutively used the three lists **Ii** and concatenated the modes (representative

[2] Indeed, even images visually similar to query images can be non relevant according to the complete query, such as churches with one tower for query 2 or straight roads outside USA for query 6.

images) as above into a single set. These images were reordered according to their ranks in **I3** and placed at the top, while the remaining images in **I3** put just after them. Two such runs were submitted: xrce_img_knn10_mRm, with R=4 and R=5 corresponding to two different sets of radii (see section 3.4.2).

- Finally xrce_img_knn10_m5 is an alternative to xrce_img_knn10_m5m, where we simply replaced the image corresponding to the local maxima by one of its 5 nearest neighbour, if this latter had a better combined relevance score (rank in **I3**).

Table 4 shows the results obtained with these strategies. Compared to table 3 we can deduce that the CR20 was not always improved, or even where it was improved the improvement was not too big. The reason is probably that the clustering criteria are purely visual and we cannot ensure that the desired cluster diversity source (such as different cities, states, sports, etc) was taken into account.

Method	P20	CR20
xrce_img_cl10_of	0.312	0.353
xrce_img_cl10_of_rg	0.290	0.360
xrce_img_cl10_ff_of	0.270	0.349
xrce_img_cl10_of_ff_rg	0.254	0.365
xrce_img_knn10_m5m	0.276	0.334
xrce_img_knn10_m4m	0.290	0.360
xrce_img_knn10_m5	0.300	0.357

Table 4: Performances (P@20 and CR@20) of different strategies.

4 Multimedia Retrieval

4.1 Standard Retrieval, without Diversity Seeking Goal

This section deals with the standard retrieval problem, where we don't care about the object redundancy/diversity in the returned ranked list. For the standard problem, the method we adopted this year is quite similar to the one we used last year (see [4]). It will be the basis for diversity-based extensions, as explained in the next section. For completeness and ease of understanding, we describe hereafter the method we used last year.

This method can be seen as a graph-based query regularization method (see [5]), where we simultaneously use two graphs: the one representing visual similarities between objects, and the one representing textual similarities between objects (objects include the query and the annotated photos of the collection). As explained in the previous sections, the cross-entropy between the language models of two objects (possibly expanded by different mechanisms such as thesaurus enrichment and/or pseudo-relevance feedback) is used as textual similarity measure, while a cosine-like norm between the "Fisher Vector" representation of two images is employed as visual similarity measure.

Let us introduce some notations. Let us call $w_t(q_t)$ the similarity vector of a given textual query q_t with respect to the documents of the collection. This similarity vector is typically obtained by normalizing the retrieval scores obtained by the methods described in the section devoted to the purely textual retrieval (we used z-score normalization). Let us call W_t the textual similarity matrix between all documents of the collection, obtained in the same way. Similarly, let us define $w_i(q_i)$ and W_i, respectively the normalized similarity vector of an image query q_i with respect to the images of the collection, and the pair-wise similarity matrix between the images of the collection. They are obtained by the methods explained in the previous section. Let us denote by $\kappa(w, k)$ the thresholding function that puts to zero all values of the vector w that are lower than the top k values and keeps all other components to their initial value.

Method	MAP	P20
Best Textual	0.260	0.308
Cross-media $Sim_{IMG-TXT}$	0.356	0.500
Cross-media $Sim_{TXT-IMG}$	0.183	0.300
C1 (Late Fusion)	0.348	0.437
C1' (Late Fusion + Visual Concepts)	0.339	0.452
C2 (TXT + IMG-TXT)	0.424	0.554
C3 (IMG + TXT-IMG)	0.246	0.382

Table 5: Performance (MAP and P@20) of the different multi-media fusion strategies

The cross-media similarity measures that we developed last year could be expressed as follows:

- $Sim_{IMG-TXT}(q_i) = \kappa(w_i(q_i), k_i).W_t$
- $Sim_{TXT-IMG}(q_t) = \kappa(w_t(q_t), k_t).W_i$

where the '.' symbol designates the standard matrix product. Note that $Sim_{IMG-TXT}$ does not rely on the textual part of the query, while $Sim_{TXT-IMG}$ does not rely on the visual part of the query. Conversely, $Sim_{IMG-TXT}$ does not rely on the visual part of the documents (photos) of the collection, while $Sim_{TXT-IMG}$ does not rely on the textual part of the images of the collection.

This can be understood as query score regularization through a two-step diffusion process, the first step being performed in one mode and the second step being performed in the other one.

Note that we could design in the same way mono-media similarity measures that amount to do some pseudo-relevance feedback:

- $Sim_{TXT-TXT}(q_t) = \kappa(w_t(q_t), k_t).W_t$
- $Sim_{IMG-IMG}(q_i) = \kappa(w_i(q_i), k_i).W_i$

We could combine all these new similarity measures, for instance by linear combination, with the basic mono-media ones. We tried several combinations, namely:

- **C1** : $\lambda_1 w_t(q_t) + \sum_u w_i(u)$ (what is classically called "late fusion")
- **C2** : $\lambda_2 w_t(q_t) + \sum_u Sim_{IMG-TXT}(u)$
- **C3** : $\lambda_3 Sim_{TXT-IMG}(q_t) + \sum_u w_i(u)$

where u iterates over the three query images associated to q_t. Note that we used the simple sum to aggregate the normalized scores (z-scores) over the different (3) images associated with the query q_t.

Table 5 reports the performance (Mean Average Precision and Precision@20) of the different multimedia combinations (including the mono-media baselines). The parameters are set to: k_i=2; k_t=25; λ_1=2; λ_2=1; λ_3=1. The textual baseline (that corresponds to w_t) is the one obtained by thesaurus-based document enrichment and pseudo-relevance feedback. For completeness, we also included the simple late fusion variant where the textual part of the query is augmented by the visual concepts detected by the XRCE detector.

The order in the cross-media similarity appears to be important: identifying the k-visual-nearest neighbors of the queries and considering their associated texts as a proxy for the textual part of the query is much more efficient than the converse (identifying the k-textual-nearest neighbors of the queries and considering their associated images as a proxy for the visual part of the query). The linear combination of the purely textual relevance score with the cross-media similarity measure (IMG-TXT) allows the system to still achieve significant improvement: this is not counter-intuitive, as the IMG-TXT cross-media similarity measure does not use the textual part of the query.

As already announced, the introduction of visual concepts as extra features in the textual representation of the query and the documents (photo annotations) appears to loose its efficiency if we take into account the medium-level visual features through the channel of the purely visual similarity, at least in a late fusion context.

4.2 Incorporating Diversity

As mentioned in section 3, one of the aims of ImageCLEFPhoto 2008 compared to previous sessions, is to promote diversity in the search results so that the first retrieved elements are not redundant. We investigated two main families of methods: implicit and explicit clustering-based approaches. The first method is commonly known as "Maximal Margin Relevance" (henceforth MMR). It amounts to re-rank the search results so that the element chosen at rank j has to be dissimilar to elements that were already selected at ranks $j' < j$. We give more details on this approach in paragraph 4.2.1. The second method is based on an explicit clustering of the first k elements, followed by a strategy designed to re-rank the elements so that many different clusters are represented among the first elements of the re-ranked list. Within that framework, different clustering algorithms and strategies can be combined. For image retrieval, some combinations have already been proposed in paragraph 3.4. In paragraph 4.2.2, we explain another particular approach that we assessed in the multi-media case.

4.2.1 Implicit clustering based approach

MMR is a re-ranking algorithm which aims at avoiding redundancy among the first elements to be re-ranked [3]. It has been successfully applied in different fields such as active learning in information retrieval [17], [9] or in document summarizing [12], [2].

We suppose that we are given a relevance score vector s_1 (for a given query q) as well as an inter-object similarity matrix S_2 (for each pair of annotated photos of the collection). The MMR framework supposes that the elements i should be ranked according to s_1 and S_2. It is a greedy algorithm: at each step (rank) j we choose the element i that maximizes the following re-ranking criterion:

$$MMR(i) = \beta(j)s_1(i) - (1 - \beta(j)) \max_{i' \in P_j} S_2(i, i') \tag{5}$$

where $\beta(j)$ is a mixture parameter (between 0 and 1) depending on the rank and P_j is the set of objects already selected (rank lower than j).

Traditionally, β is kept constant, but we propose here a more efficient variant, where $\beta(j)$ linearly increases between $\beta(1) = \alpha$ (< 1) and $\beta(k)$=1 for some k (typically k=100), before saturating at value $\beta = 1$.

Regarding the choice of s_1, we adopted the (best) combination of mono-media and cross-media similarity measures (based on the **C2** measure introduced in section 4.1). For S_2, we tested different possibilities. We will give some details on this part in paragraph 4.2.3.

4.2.2 Explicit clustering based approach

In this method, for a topic q, we start by selecting the first k elements according to s_1. Let us denote by Q_k the subset constituted of these selected elements. Next, we cluster objects within Q_k in order to find different themes. The similarity measures used in the clustering process are provided by a similarity matrix between multi-media objects, S_2. From this global matrix, we extract the sub-matrix corresponding to objects in Q_k, and we row-normalize it. This row-normalized sub-matrix is denoted S'_2. Then, we use the Relational Analysis (RA henceforth) approach for the clustering stage [14], [13].

From a theoretical point of view the RA method allows us to model the clustering problem through an integer linear program. The extended form of the partitioning criterion used in this approach is the following one:

$$C(S'_2, X) = \sum_{i,i'}^{k} \underbrace{(S'_2(i,i') - m)}_{\text{contribution}} X(i,i') \tag{6}$$

m representing a threshold to which the similarity between two objects is compared. X is called a relational matrix. It is a binary square matrix of dimension k, which represents the partition we are looking for.

In our case, we avoid fixing arbitrarily this parameter by taking a central tendency measure of the similarities [1]. For our experiments we took the arithmetic means of the strictly positive similarities. We first define $\mathbb{S}'^{+}_2 = \{(i,i') \in Q_k \times Q_k : S'_2(i,i') > 0\}$. Then we have:

$$m = \frac{1}{|\mathbb{S}'^{+}_2|} \sum_{(i,i') \in \mathbb{S}'^{+}_2} S'_2(i,i') \tag{7}$$

The more the similarity between two objects exceeds m, the higher the chance for them to be in the same cluster.

The general term $X(i,i')$ equals 1 if element i and i' are in the same cluster; 0 otherwise. Using this formalism, we can express the relational properties that a partition must respect by linear equations [14]:

- binarity: $X(i,i') \in \{0,1\},\ \forall i,i' = 1,\ldots,k$
- reflexivity: $X(i,i) = 1,\ \forall i = 1,\ldots,k$
- symmetry: $X(i,i') - X(i',i) = 0,\ \forall i = 1,\ldots,k$
- transitivity: $X(i,i') + X(i',i'') - X(i,i'') \leq 1,\ \forall i,i',i'' = 1,\ldots,k$

As the partitioning criterion and the constraints are linear with respect to X, we can use integer linear programming in order to solve the clustering problem. Furthermore, we can see that this model doesn't require to fix the number of clusters. However, the integer linear program is NP-hard and in practice, we need to apply an heuristic for dealing with large data-sets.

The heuristic we used in our experiments is quite similar to the leader algorithm introduced in [7]. This approach is based upon transfer operations [13], [1]: at each step, it reassigns an element to the cluster with which its aggregated contribution is maximal. If this last contribution score is negative then it creates a new cluster based on the current element.

We give a sketch of this heuristic in algorithm 1. Using this algorithm, we don't need to fix the number of clusters which can evolve at anytime. It is a property that is important as it allows to obtain pure clusters (as long as the parameter m is strict enough and this is the case when taking the value given by equation (7)). However, we need to provide a number of iterations (or a stopping criterion) for having an approximated solution in a reasonable amount of time; the algorithm typically converges in less than 5 iterations for a number of elements (k) smaller than 500.

This kind of algorithm is highly dependent on the objects' scanning order. In our context, this order can be given by the relevance score provided by s_1. In other words, the scanning order of objects Q_k used in the RA heuristic is the decreasing order based on s_1: we first take the most relevant objects then the less relevant ones.

After having clustered Q_k, the k most relevant multi-media elements of a topic, we have to define a re-ranking strategy which takes into account the diversity provided by the clustering results. The main idea of our approach is to represent, among the first re-ranked results, elements

Algorithm 1 RA heuristic

Require: $nbitr$ = number of iterations; S'_2 the similarity matrix
  $i = 1$
  Take the first element i as the first element of the first cluster clu_1
  $\kappa = 1$ where κ is the current number of cluster
  for $l = 1$ to $nbitr$ **do**
    for $i = 1$ to k **do**
      for $j = 1$ to κ **do**
        Compute the contribution of i with cluster clu_j: $cont_j = \sum_{i' \in clu_j}(S'_2(i,i') - m)$
      end for
      j^* is the cluster id which has the highest contribution with i and $cont_{j^*}$ is the corresponding contribution value
      if $cont_{j^*} < (S'_2(i,i) - m)$ **then**
        Create a new cluster where i is the first element
        $\kappa \leftarrow \kappa + 1$
      else
        Assign i to cluster clu_{j^*}
        if the cluster where was taken i before its new assignment, is empty **then**
          $\kappa \leftarrow \kappa - 1$
        end if
      end if
    end for
  end for
  Return the partition found

which belong to different clusters until a stopping criterion is fulfilled. The strategy employed is described in algorithm 2. In this approach the scanning order of objects in Q_k is the same one as previously used, namely we take the most relevant objects first. By doing this, each cluster is represented by its most relevant object according to s_1.

Algorithm 2 Re-ranking strategy for a topic

Require: A topic q; s'_1 the relevance score vector between objects in Q_k and q; R the clustering results of objects in Q_k
  $L1$, $L2$ and CL are empty lists
  $i = 1$
  Take the first element i as the first element of the re-ranked list $L1$
  Add $R(i)$, the cluster id of element i in the cluster list CL
  $i = 2$
  while $i \leq k$ and Stopping criterion is not fulfilled **do**
    if $R(i) \in CL$ **then**
      Append i to $L2$
    else
      Append i to $L1$ and add $R(i)$ in CL
    end if
    $i = i + 1$
  end while
  $L3$ is the complementary list of objects from i to k
  Extend $L1$ with $L2$ then with $L3$
  Return $L1$

We tested different stopping criteria in algorithm 2:

1. the number of different clusters represented among the first results should not exceed a predefined number $nbdiv \in 1, \ldots, \kappa$ where κ is the number of clusters found

2. we append different clusters' representatives as long as their relevance scores are not lower than a predefined threshold $qpert \in s'_1 = \{s_1(i,q) : i \in Q_k\}$

Concerning the first stopping criterion, let us assume that $nbdiv = 10$. Then, this implies that the first 10 elements of the re-ranked list have to belong to 10 different clusters (assuming that $\kappa \geq 10$). Once 10 different clusters are appended, the complementary list (from the 11^{th} rank to the k^{th} rank), is constituted of the remaining multi-media objects sorted in the decreasing order with respect to s_1 and without taking into account the cluster membership information.

For illustrating the second stopping criterion, let us take for example $k = 100$. Q_k is sorted according to the decreasing order with respect to s_1 (such as for the scanning order). Then if we take furthermore $qpert = s'_1(70)$, this means that the threshold is the 30^{th} percentile of the distribution given by s'_1. Accordingly, in the re-ranked list, we append the most relevant element of a cluster that it is not already represented as long as the relevance measure of this chosen element is greater or equal to the 30^{th} percentile of s'_1. This is an alternative which, unlike the first stopping criterion, avoid fixing a number of clusters.

4.2.3 Description of AUTO-TXTIMG runs

In this paragraph, we give more details about the runs we submitted in the EN-EN-AUTO-TXTIMG category. Our different approaches concerning this sub-task have been described in the previous paragraph. For s_1, we obviously chose the best multimedia combination as explained in section 4.1. Let us come back to the choice of the pairwise similarity matrix S_2 between the annotated photos of the collection. The kind of similarity coefficient computed in S_2 differs in general from s_1. The idea is to investigate whether some weighting schemes are more adequate, or whether the use of a single mode or the use of specific fields give better results in diversity-based re-ranking. Let us be more precise in the three alternatives we adopted for S_2:

1. the cross-entropy measure between texts given by the concatenation of the title and the location fields associated to each multi-media object. This similarity matrix is referred as "tilo" in the title of our runs.

2. the $tfidf$[3] measure between texts given by the concatenation of the title, the location and the narrative fields associated to each multi-media object. This similarity matrix is referred as "tfidf" in the title of our runs.

3. the (best) cross-media similarity measure which combines textual and visual similarity measures such as described in paragraph 4.1. In other words, it is the extension of s_1 to pairs of elements in the dataset. This similarity matrix is referred as "cm" in the title of our runs.

Clustering – or re-ranking – is limited to the k most relevant elements with respect to a topic q. In all our runs, we took $k = 100$.

The title of our runs is constituted as follows: "xrce_*(S2 matrix)*_*(Stopping criterion)*_*(Parameter value)*". We give below some examples of the meaning of the titles of our runs:

- xrce_tilo_nbdiv_15: explicit clustering-based approach where S'_2 is constituted of the cross-entropy measure using textual data (title and location fields). The number of different clusters is $nbdiv = 15$.

- xrce_cm_mmr_07: implicit clustering based approach where S'_2 is constituted of the global similarity measures using textual and visual data. The basic mixture parameter in the MMR re-ranking function (for rank 1) is set to $\alpha = 0.7$ (β is linearly increasing from 0.7 to 1, value achieved for j=k=100).

[3] the standard $tfidf$ implemented in the LEMUR toolbox

- xrce_cm_qpert_70: explicit clustering based approach where S'_2 is constituted of the global similarity measures on textual an visual data. The relevance threshold to stop appending new clusters is given by the 30^{th} percentile of S'_1.

Our baseline is the basic result we obtained without any diversity goal seeking (such as in paragraph 4.1):

- xrce_cm_best_basic: it is the basic search result based on the global similarity measures **C2** using textual and visual data, which doesn't involve any methods for promoting diversity.

Finally, we give in table 6 the results obtained for the best runs we submitted in the EN-EN-AUTO-TXTIMG category. The runs are sorted with respect to the mean between the ranks obtained according to P20 and CR20 among XRCE runs.

Run	P20	CR20
xrce_tilo_nbdiv_10	0.5282	0.4146
xrce_cm_nbdiv_10	0.5269	0.4111
xrce_tilo_nbdiv_15	0.5115	0.4262
xrce_cm_mmr_07	0.5282	0.4015
xrce_cm_qpert_70	0.5256	0.3965
xrce_tfidf_nbdiv_10	0.5115	0.4081
xrce_tilo_mmr_07	0.5167	0.4006
xrce_tilo_mmr_05	0.3936	0.4175
xrce_tilo_qpert_70	0.5244	0.3953
xrce_cm_best_basic	0.5577	0.3727

Table 6: Precision at 20 (P20) and Cluster Recall at 20 (CR20) measures for the best runs in EN-EN-AUTO-TXTIMG category

Our best runs allow us to improve the CR20 measure compared to the baseline xrce_cm_best_basic. Therefore, both the implicit and the explicit clustering based methods allow us to incorporate diversity into the first re-ranked elements. However, we are less precise in terms of P20. Concerning the different kinds of measures we chose to use for representing thematic similarity between annotated photos, we can see that either the pure textual data ("tilo ", constituted of title and location fileds) using cross-entropy measures or the combination of textual and visual data ("cm") using cross-media similarities perform better.

5 Conclusion

This year, the main lessons we learnt from our participation to ImageCLEF-Photo were:

- in the case of pure text-based retrieval, both document enrichment by thesaurus and query enrichment improve the results, and combining the former with query expansion using PRF improves further the results (Table 1);
- Fisher Vectors are rich image signatures and have state-of-the-art performance both in visual concept detection (see Figure 1) and content based image retrieval (Table 3);
- the use of the visual concepts increases the retrieval performance when combined with pure text, but this advantage is lost when we use other, more complex multi-media fusion mechanisms, based on lower-level features than the visual concepts (Table 2);
- combining the two mono-media information sources (image and text) using cross-media similarities based on transmedia pseudo-relevance feedback improves significantly (by more than 50% relative) the retrieval results (xrce_cm_best_basic with MAP=0.39 and P@20=0.56).

- concerning the diversity, most strategies we proposed succeeded in reducing the redundancy in the top documents. As none of the techniques used explicitly the provided clustering criterion (e.g. diversifying according to cities or states or sports, etc.), the CR20 score was not always significantly increased (or in a few cases it was even decreased). This is not surprising, as we were seeking and improving the diversity in a blind (unsupervised) way, and therefore the "new" documents introduced at top level in many cases were not necessarily different following the criterion the organizers decided. For example, a pure visual content-based system will surely not group pictures of "persons on a beach" into different clusters each corresponding to a different "country" (topic 34). There is more chances that this happens for text or multi-modal system. Obviously, one possible improvement is the explicit integration of the provided "clustering criterion" into the diversification strategy when possible.

References

[1] J. Ah-Pine. *Sur des aspects algébriques et combinatoires de l'Analyse Relationnelle.* PhD thesis, University of Paris 6, 2007.

[2] F. Boudin, M. El-Bèze, and J.M. Torres-Moreno. A scalable MMR approach to sentence scoring for multi-document update summarization. In *COLING*, 2008.

[3] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.

[4] Stephane Clinchant, Jean-Michel Renders, and Gabriela Csurka. Xrce's participation to imageclef 2007. In *Working Notes of the 2007 CLEF Workshop*, Budapest, Hungary, September 2007.

[5] F. Diaz. Regularizing query-based retrieval scores. *Information Retrieval*, 10, 2007.

[6] M. Ester, H. P. Kriegel, and X. Xu. A density-based algorithm for discovering clusters in large databases with noise. In *In Proceedings of ACM SIGKDD*, 1996.

[7] J.A. Hartigan. *Clustering Algorithms.* John Wiley and Sons, 1975.

[8] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *J. of Machine Learning*, 42, 2001.

[9] T.S. Huang, C.K. Dagli, S. Rajaram, E.Y. Chang, M.I. Mandel, G.E. Poliner, and D.P.W. Ellis. Active learning for interactive multimedia retrieval. In *IEEE*, 2008.

[10] B. Krishnapuram, L. Carin, M. Figueiredo, and A. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *PAMI*, 2005.

[11] Lemur. http://www.lemurproject.org/.

[12] Z. Lin, T.S. Chua, M.Y. Kan, W.S. Lee, L. Qiu, and S. Ye. NUS at DUC 2007: Using evolutionary models of text. In *Document Understanding Conference*, 2005.

[13] J.F. Marcotorchino and P. Michaud. Heuristic approach of the similarity aggregation problem. *Methods of operation research*, 43:395–404, 1981.

[14] P. Michaud and J.F. Marcotorchino. Optimisation en analyse de données relationnelles. In *Data Analysis and informatics.* North Holland Amsterdam, 1980.

[15] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007.

[16] J. Ponte and W.B. Croft. A language modelling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281. ACM, 1998.

[17] X. Shen and C.X. Zhai. Active feedback in ad hoc information retrieval. In *SIGIR*, 2005.

[18] ImageClef Visual Concept Detection Task. http://www.imageclef.org/2008/vcdt.

[19] C. Zhai and J lafferty. Model-based feedback in the kl-divergence retrieval model. In *Proceedings of the 10th International Conference on Information and Knowledge Management (CIKM)*, pages 403–410, 2001.

[20] C. Zhai and J lafferty. A study of smoothing methods for language models applied to ad hoc to information retrieval. In *Proceedings of SIGIR'01*, pages 334–342. ACM, 2001.