

XRCE's participation to ImagEval

Stephane Clinchant, Gabriela Csurka, Florent Perronnin and Jean-Michel Renders
Xerox Research Centre Europe, 6, ch. de Maupertuis, 38240 Meylan, France
FirstName.LastName@xrce.xerox.com

ABSTRACT

This document describes XRCE's participation to Imageval, more specifically to the mixed Text-Image search. After reviewing state-of-the-art methods to exploit the correlations between texts and images in multimedia retrieval, we will examine the single-media search components and describe how we have combined them in the framework of ImagEval. It appeared that, with our current settings and the Imageval corpus, no "early fusion" approach gave significantly better results than a "late fusion" method, so that this paper is mainly dedicated to the latter approach. In this track, exploiting textual information with the Language Modelling approach alone already offered very satisfying performance, much larger than purely visual search. Still, late fusion was able to increase monomedia results by more than 10% (relative), showing the usefulness of combining both types of information, even if the purely visual retrieval component gives relatively poor results.

1. INTRODUCTION

Efficient access to multimedia information requires the ability to search and organize the information. While the technology to search and retrieve text has been available for some time - and is familiar to many people in the form of web search engines - the technology to search images and videos is much more challenging. Early systems were based mainly on visual similarity with a query image making use of lower-level features like texture, colour, and shape. The pure visual-based approach to retrieval has several drawbacks. It does not actually bridge the semantic gap but rather forces the user to work on low-level feature space. A gap remains between the user's conceptualization of a query and the query that is actually specified to the system.

The ideal CBIR (content based image retrieval) system would provide an access to an image repository involving a query either depicting specific types of object or scene using textual descriptions ("find a photo with sunset on the beach"), evoking a particular mood, or simply containing a specific texture or pattern (query by example). Potentially, images have many types of attribute which could be used for retrieval, including the presence of a particular combination of shape, texture and colour, the depiction of a partic-

ular object, scene, event, named individuals or locations, metadata or even more abstract attributes such as activities or emotions that might be associated to the image.

According to this, the user may want to access the repository using image query (to illustrate her/his needs), text to name or describe or both. One of the tasks (Task 2) in the shared evaluation tracks for image indexing and retrieval campaign ImageVAL¹ was exactly dedicated at comparing performance of combined text/image retrieval in the context of "search on the web". During this campaign, we investigated several families of text-image fusion methods but it appeared that, with our current settings and for the ImagEval corpus, the late fusion approach (doing mono-media search and then combining the resulting scores) worked better. This report will therefore mainly focus on the description of these methods and the late fusion we used in our runs. However, it is well-known that such a late fusion of information could be suboptimal as much relevant information about the correlation of the different modalities is discarded. More recent approaches have considered fusion at the data level, estimating from the training data the correspondences or joint distributions between components across the image and text modes (see section 2).

2. STATE-OF-THE-ART

One of the approaches based on early fusion is the Co-occurrence Model by Mori et al. [28]. They simply divide the image into sub-images and assign all textual keywords to each sub-image. They further vector quantize the features describing the sub-images and accumulate the frequencies of the words within clusters and calculate the likelihood for every word. For a test image, they assign each sub-image feature to the closest cluster and combine the likelihood of cluster to design the most relevant words.

Instead of looking to the co-occurrences, Vinokourov et al [38] proposed to find correlations between images and attached text using the kernel Canonical Correlation Analysis (KCCA). The method is inspired by cross-language methods in text retrieval [39] where translation invariant semantics of the text are extracted from aligned multilingual documents using KCCA.

The work of Duygulu et al [10] has similar analogy with cross-language methods for text. Their method inspired by [6] proposes to consider image annotation as machine translation between image regions and keywords. Therefore, they first segment the images into regions and described them with a variety of features related to size, position, colour and shape. The features are further clustered leading to a vocabulary of blobs. Finally they learn a mapping between region types and keywords (nouns taken from a large vocabulary) supplied with the images.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...\$5.00.

¹http://imageval.org/e_presentation.html

In [3] they extend this model to multi-modal data based on Hofmann’s hierarchical clustering combining the latent aspect model with soft clustering. Images and co-occurring text are generated by nodes arranged in a tree structure. The nodes generate both image regions using a Gaussian distribution, and words using a multinomial distribution. Each cluster is associated with a path from a leaf to the root. Taking all clusters into consideration, a document is modeled by a sum over the clusters, weighted by the probability that the document is in the cluster. In contrast to the direct translation model [10], the relationships between specific image regions and words is not modeled explicitly, but encoded to some extent through co-occurrences (“topics” collected at the nodes). To strengthen the relationship between words and image regions, they further build explicit correspondence information in the hierarchical clustering models either in an asymmetrical manner, where the word emission of a word is influenced by the region (blob) emission, or in a symmetrical manner, where the observed words and regions are emitted in pairs. Prediction for documents not in the training set is obtained either by marginalizing out the training data, as in Blei et al [5] or estimating the mixing weights using a cluster specific average computed during training.

Blei et al in [5] formalize three hierarchical probabilistic mixture models aiming to describe data with multiple types such as image and text. The Gaussian-multinomial (GM) mixture model uses a single discrete latent variable z to represent a joint clustering of an image and its caption. An image/caption is assumed to be generated by first choosing a value of z , and then repeatedly sampling N region descriptions and M caption words, conditioned on the chosen value of z . The Gaussian-multinomial LDA (GM-LDA) samples a Dirichlet random variable q which provides a probability distribution over the latent factors. Within an image, all the region descriptions and words are generated with q held fixed, while allowing the latent factor for each word and region description to (potentially) vary. Finally their correspondence latent Dirichlet allocation (Corr-LDA), combines the flexibility of GM-LDA with the association capabilities of GM-Mixture.

Jeon et al [15] proposed to estimate and exploit the joint probability distributions of blobs that could appear in image and words that could appear in the caption of the image assuming mutual independence between a word and the blobs given an image J . These joint probabilities can be used in two ways to annotate/retrieve images. From one hand, their probabilistic or fixed annotation-based cross-media relevance model (P/F ACMRM) corresponds to document-based expansion, where the blobs corresponding to each test image are used to generate words and associated probabilities from the joint distribution of blobs and words. Each test image can, therefore, be annotated with a vector of probabilities for all the words in the vocabulary. Alternatively, their direct-retrieval cross-media relevance model (DRCMRM) corresponds to query expansion. The query word(s) is used to generate a set of blob probabilities from the joint distribution of blobs and words. This vector of blob probabilities is compared with the vector of blobs for each test image using Kullback-Leibler (KL) divergence and the resulting KL distance is used to rank the images.

In [21], they showed that working directly with continuous features describing the blobs instead of quantizing them into clusters (which is the case of CMRM, co-occurrence and translation models) performs better. In their Continuous-space Relevance Model (CRM) the joint probability of a region in the test image being associated with query words is computed as an expectation over the training samples. To improve the retrieval performance the model is further normalized in [20]. In contrast to this model where the annotation words for any given image are assumed to follow a multi-

nomial distribution, Feng et al [13] proposed to model them with a multiple Bernoulli model and the image feature probabilities using a non-parametric kernel density estimate.

Monay and Gatica-Perez [27] address the problem of unsupervised image auto-annotation with probabilistic latent space models. The authors improve their PLSA-mixed system [26] which applies a standard PLSA on a concatenated representation of the textual and the visual modalities by modelling the documents two linked PLSA models sharing the same distribution over aspects. This formulation allows to treat each modality differently and give more importance to the captions in the latent space definition. A first PLSA model is completely trained on the set of image captions to learn both the probability of a keywords given an aspect $P(t|z)$ and probability of the latent aspect given a document $P(z|d)$. In the next step they train a second PLSA on the visual modality to compute the probability of a visual feature given the latent aspect $P(v|z)$, keeping $P(z|d)$ fixed.

Kosinov et al [19] goes beyond the traditional term document matrix paradigm in applying a spreading activation model context to make it more suitable for content-based digital media retrieval. The approach is initialized with the term document matrix, however initial activations are propagated via a diffusion process across terms within a document according to the high-level semantic similarities of terms (words) as well as across the documents through the low-level feature-based similarities of documents.

In contrast to previous approaches where the blobs/segments are assumed to be statistically independent, Carbonetto et al [7], allow for interactions between blobs through a Markov random field (MRF). Hence, the probability of an image blob being aligned to a particular word depends on the word assignments of its neighbouring blobs. The dependence between neighbouring objects introduces spatial context to the annotation/classification.

Similarly, the ALIP system of Li and Wang [22] models the interaction between blocks using Markov Models. Simple block-based features are extracted from each training image at several resolutions and a two-dimensional multi-resolution hidden Markov model (2D-MHMM) is used to describe statistical properties of the feature vectors and their spatial dependences. The similarity between the image and a category of images in the database is assessed by the log likelihood of this instance under the model trained from images in the category. The auto-annotation is based on a ranking of words within the description of the most likely categories/concepts according to their statistical significance. Most significant words are used to index the image.

A graph-based approach is proposed equally by Pan et al in [30]. In their GCap images and their attributes (caption words and regions) are represented as nodes of a graph and they are linked according to their known associations. For image captioning, they propose a 3-layer graph, with one layer of image nodes, one layer of captioning term nodes, and one layer for the image regions. They further define two types of links, NN (nearest-neighbour) links between the nodes of two similar regions; and IAV (Image-Attribute-Value)-links, between an image node and an attribute value (caption term or region feature vector) node. Finally random walk with restarts (“RWR”) on the built graph is used to create image captions.

The above approaches try to correlate or find co-occurrences between the keywords/text in the annotation and images using the training database as the sole information. Others try to integrate in their system some external information/knowledge, such as the electronic thesaurus WordNet. For example, Srikanth et al [37] use Wordnet to derive the hierarchical dependencies between annotation words and to generate improved visual lexicons for the

translation-based approaches. Kosinov et al [18] build a concept hierarchy using annotation words and their hyponyms derived from WordNet. Every concept occupies a separate node in H, and is associated with a binary classifier designed to distinguish the set of leaf concepts subsumed (directly or indirectly). The relevance of a concepts to a query is estimated as a trade-off between the goodness of fit to a given category description and its inherent uncertainty. Benitez et al [4] uses the WordNet to disambiguate the senses of the words in the annotations during their automatic image class discovery process.

Wang et al [40] propose to go further by using the whole Web as external information. To annotate images they first select a query image and an accurate keyword for it. Then they use CBIR techniques to search for semantically and visually similar images in a large database, e.g. the Web. Gathering all text information related/surrounding the retrieved images they use the Search Result Clustering (SRC) algorithm to cluster the retrieved semantically and visually similar images according to their titles, URLs and surrounding texts. The clusters are ranked according to maximum size and average member image scores and the dominant cluster concepts are used to annotate the query image.

3. GENERAL FRAMEWORK

In the previous section, we reviewed several techniques taking into account interactions between text and images. However, in the ImagEval settings, the different “early fusion” approaches we adopted did never reached a superior performance level with respect to a “late fusion” approach, where mono-media retrievals are performed first, followed by a combination operator acting on the individual scores. The figure 3 depicts the global architecture of our system: the basic components are two (monomedia) information retrieval systems and a score combiner that merges their results before presenting them to the user. First, the text side of the system will be detailed, followed by the image one.

3.1 A brief introduction to Information Retrieval

To express his information needs, the user forms a query that is then compared to the documents of the collection. Hence an IR system offers a query model (the transformation from information needs to query words), a document model and a mean to compare a document to a query according to a criterion called relevance: does this document answer the user’s information need ? Originally, queries and documents were represented as logical propositions. A document would be relevant if it implies the query. Then the vectorial models replaced them. Document and queries were represented in a vectorial space whose dimensions were the different words of the collection. Relevance was modelled by geometric similarity through the scalar product. Vectorial Models used several weighting schemas, such as the well known *tf-idf*.

Nowadays, the leading models are probabilistic: Okapi[35], Language Models[33] and Divergence from Randomness[1]. We will detail the language modelling approach to IR, since it is the one we adopted in this current work.

3.2 A language modelling approach to IR

The core idea of language models is to determine the probability $P(q|d)$ - the probability that the query would be generated from a particular document. The concept of relevance is not directly modelled but the (assumed) underlying process is the following: the user has an information need ; he guesses an ideal document. From this ideal document, he chooses some words which make its query. Thus, the most relevant documents are those which are the

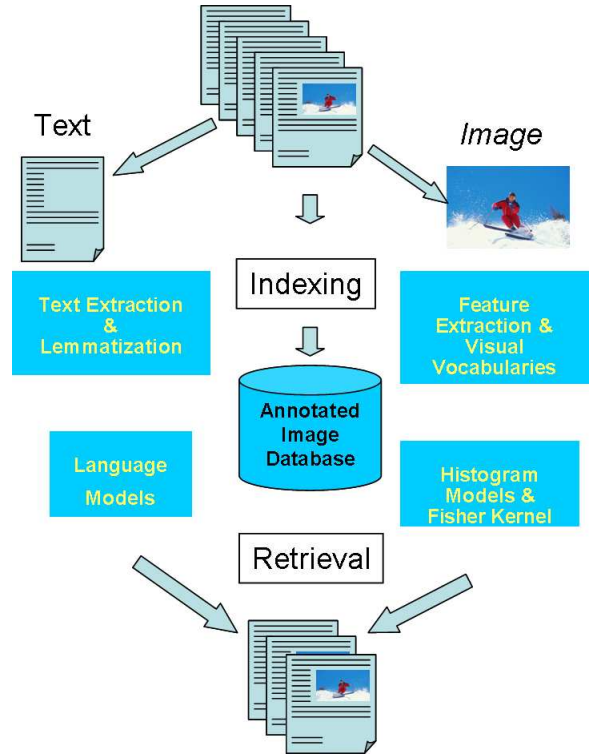


Figure 1: The schema.

most likely to generate the query.

Formally, given a query q , the language model approach to IR [33] scores documents d by estimating $P(q|d)$, probability of the query according to a language model of the document. For a query $q = \{q_1, \dots, q_\ell\}$, we get:

$$P(q|d) = \prod_{i=1}^{\ell} P(q_i|d). \quad (1)$$

For each document d , a simple language model is obtained by considering the frequency of words in d , $P_{ML}(w|d) \propto \#(w, d)$ (this is the Maximum Likelihood, or ML, estimator). The probabilities are smoothed by the corpus language model $P_{ML}(w|\mathcal{D}) \propto \sum_d \#(w, d)$. The resulting language model is:

$$P(w|d) = \lambda P_{ML}(w|d) + (1 - \lambda) P_{ML}(w|\mathcal{D}). \quad (2)$$

The reasons of smoothing are twofold: first a word can be present in a query but absent in a document. However this fact does not make it impossible and the document should give it a probability. The second reason is to play a role like IDF. Smoothing allows implicitly to renormalize the frequency of one word in a document with respect to its occurrence in the corpus. Others smoothing methods are available and can be found in [41]. Other extensions of language models take into account pseudo-feedback methods, as well as cross lingual information retrieval.

3.3 Text Pre-Processing

The data was preprocessed in the following way. Each original document, containing both textual sections and images, was first segmented and represented by a sequence such as:

$$(w_1, w_2, \dots), image1, (w_{21}, \dots) image2, \dots$$

As we wanted to associate to each image the most relevant textual part of the document, we decided to keep the left and right tex-

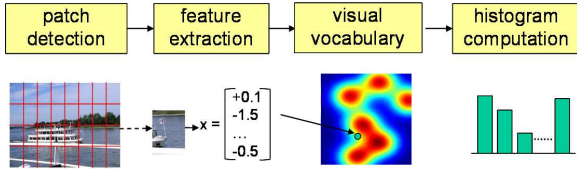


Figure 2: Image histogram building steps.

tual neighbourhoods of each image as its associated text. Note that texts associated with different (neighbor) images in general overlap. The name of the image was also added in that text. Then, for every image, its associated textual part was lemmatized with Xerox Finite State Transducers. Lemmatization refers to transforming a word into its canonical dictionary form, for example retrieving into retrieve, dogs into dog.

3.4 Visual Vocabulary

The bag of visual-words (BOV) [36, 9, 34] was inspired by the bag-of-words used in text categorization/retrieval. The main idea here is to define a visual vocabulary, and then to characterize the image with the number of occurrences of each visual word. The visual vocabulary provides a “mid-level” representation which helps to bridge the semantic gap between the low-level features extracted from an image and the high-level concepts to be categorized [2]. However, the main difference from text categorization is that there is no given vocabulary for images.

Instead we generate a visual vocabulary automatically from a set of images as described in 3.4. Therefore we need to define a set of features. Almost any set of features extracted from images can be the base of a visual vocabulary. However, those features have to be able to handle some variations irrelevant to the task of the retrieval or categorization, such as viewpoint change, lighting variations or occlusions. Therefore, local features are preferable on global features. Their further advantages are that several feature sets are extracted from a single image.

Different descriptors extracted locally on regions of interest (ROI) were used in BOV approaches mainly based on texture, colour, shape, structure or their combination. The ROI can be obtained by image segmentation [3, 7, 8, 23], by applying specific interest point detectors [9, 34], by considering a regular grid [7, 12] or simply random sampling of image patches [25, 29]. All features extracted are then mapped to the feature space and clustered to obtain the visual vocabulary.

Given the feature space, the visual vocabulary is built through the clustering of low-level feature vectors using for instance K-means [36, 9], Gaussian Mixture Models (GMM) [11, 32] or mean-shift [17].

In our experiments, image patches are extracted on regular grids at 3 different scales with a ratio of $\sqrt{2}$ between two consecutive scales after rescaling the images so that they all contain approximately 100K pixels (the aspect ratio is preserved). Since all images contain roughly the same number of pixels, they also all contain approximately the same number of patches (between 300 and 400).

Our system makes use of two types of low-level features: grey-level SIFT features [24] and colour features. To extract a SIFT feature, we first divide the patch regularly into square sub-regions and each sub-region is described with a gradient histogram (using only grey-level information). Typically, there are $4 \times 4 = 16$ sub-regions and each histogram contains 8 bins which leads to a 128 dimensional feature vector. As for the extraction of colour features, each patch is also subdivided into 16 square sub-regions and each

sub-region is described with the means and standard deviations of the 3 RGB channels, which leads to a 96 dimensional feature vector.

The dimensionality of these feature vectors was subsequently reduced down to 50 using principal component analysis (PCA). This dimension reduction has three benefits. It decorrelates the dimensions of the feature vectors and thus makes the diagonal assumption more reasonable for GMMs. Discarding the last components also removes noise and thus increases the performance. Finally, it significantly reduces the cost of Gaussian computations.

The visual vocabulary estimation is performed by clustering the low-level feature vectors (after PCA projection). Assuming that the generation process of feature vectors can be modeled by a given probability density function (pdf), clustering may be performed by maximizing the likelihood of the observations given the parameters of this pdf (maximum likelihood estimation or MLE):

$$\log p(X|\lambda) = \sum_{t=1}^T \log p(x_t|\lambda). \quad (3)$$

where λ denote the set of parameters of the generative model and $X = \{x_t, t = 1 \dots T\}$ the set of training samples, *i.e.* the set of low-level features extracted from the set of image patches.

We use the GMM as a generative model proposed in [11, 32] where each Gaussian models a visual word. Then, $\lambda = \{w_i, \mu_i, \Sigma_i, i = 1 \dots N\}$ where w_i , μ_i and Σ_i denote respectively the weight, mean vector and covariance matrix of Gaussian i and where N denotes the number of Gaussians. Each Gaussian models a word of the visual vocabulary, where w_i encodes the relative frequency of a visual word, μ_i its mean and Σ_i the variation around the mean. Similarly to the approach described in [32], we assume that the covariance matrices are diagonal.

If q_t is the hidden mixture variable associated with the observation x , then the probability that x has been generated by the GMM, can be written as:

$$p(x|\lambda) = \sum_{i=1}^N w_i p(x|q_t = i, \lambda) \quad (4)$$

where the weights are subject to the constraint $\sum_{i=1}^N w_i = 1$ and:

$$p(x|q_t = i, \lambda) = \frac{\exp\left\{-\frac{1}{2}(x - \mu_i)' \Sigma_i^{-1} (x - \mu_i)\right\}}{(2\pi)^{D/2} |\Sigma_i|^{1/2}},$$

where D is the dimensionality of the feature vectors and $|\cdot|$ denotes the determinant operator.

The equation 3 is then maximized by the Expectation Maximization (EM) algorithm using the features of set of images as training data (see details in [32]). In the case of a big database, the training set can be obtained by a random sampling of the database.

To obtain the occupancy histogram of an image, we compute the occupancy probabilities of each observation x_t of the image related to each Gaussians. Using the Bayes formula, the occupancy probability $\gamma_t(i)$ (the probability for observation x_t to have been generated by the i -th Gaussian) can be written as:

$$\gamma_t(i) = p(q_t = i|x_t, \lambda) = \frac{w_i p(x_t|q_t = i, \lambda)}{\sum_{j=1}^N w_j p(x_t|q_t = j, \lambda)}. \quad (5)$$

To obtain the global occurrence histogram, it is sufficient to accumulate these occupancy probabilities over the samples. This results in a soft (continuous) histogram representation, in contrast with K-means based approaches where each image patch is assigned to a single “visual word” (cluster).

3.5 Image Signature

Using the visual vocabulary we extracted different image signatures.

Universal Histogram

The simplest signature we can construct from the visual vocabulary is the soft histogram representation, where we accumulate the occupancy probabilities over all image patches.

Universal Histogram with Language Model

The analogy between bag of visual-words (BOV) and bag of words enables to define *visual language* models, which are simply unigram models of the BOV. For each image, an histogram can be defined with the universal vocabulary. For practical reasons, as we wanted to test text retrieval algorithms on image, the histograms were discretized. This discretization can be seen as an approximation of the real distribution. Hence, all of our textual information retrieval tools can be used with the visual language model. We use the standard query likelihood between an image query and an other image $P(q_{image}|d_{image})$ as the retrieval function.

Fisher Kernel

Fisher kernels have been introduced to combine the benefits of generative and discriminative approaches [14]. Let p be a pdf whose parameters are denoted λ . Then one can characterize the samples $X = \{x_t, t = 1 \dots T\}$ with the following gradient vector:

$$\nabla_{\lambda} \log p(X|\lambda). \quad (6)$$

Intuitively, the gradient of the log-likelihood describes the direction in which parameters should be modified to best fit the data. It transforms a variable length sample X into a fixed length vector whose size is only dependent on the number of parameters in the model.

This gradient vector can then be classified using any discriminative classifier. For those discriminative classifiers which use an inner product term it is important to normalize the input vectors. In [14], the Fisher information matrix F_{λ} is suggested for this purpose:

$$F_{\lambda} = E_X [\nabla_{\lambda} \log p(X|\lambda) \nabla_{\lambda} \log p(X|\lambda)'] . \quad (7)$$

The normalized gradient vector is thus given by:

$$F_{\lambda}^{-1/2} \nabla_{\lambda} \log p(X|\lambda). \quad (8)$$

In [31], Perronin and Dance proposed to use this framework on visual vocabularies, where the vocabularies of visual words are represented by means of a GMM (see also section 3.4). They use the normalized gradient in a discriminative approach for image categorization. We use the same normalized gradient as an image signature. In which follows we briefly describe how it is obtained in the case of visual vocabularies according to [31].

As described in section 3.4, each Gaussian represents a word of the visual vocabulary. Under an independence assumption, we have:

$$\log p(X|\lambda) = \sum_{t=1}^T \log p(x_t|\lambda). \quad (9)$$

The likelihood $p(x_t|\lambda)$ that observation x_t was generated by the GMM is given by (4).

The gradient vector $\nabla_{\lambda} \log p(X|\lambda)$ can then be computed using

straightforward derivations (see further details in [31]):

$$\frac{\partial \log p(X|\lambda)}{\partial w_i} = \sum_{t=1}^T \left[\frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1} \right] \text{ for } i \geq 2, \quad (10)$$

$$\frac{\partial \log p(X|\lambda)}{\partial \mu_i^d} = \sum_{t=1}^T \gamma_t(i) \left[\frac{x_t^d - \mu_i^d}{(\sigma_i^d)^2} \right], \quad (11)$$

$$\frac{\partial \log p(X|\lambda)}{\partial \sigma_i^d} = \sum_{t=1}^T \gamma_t(i) \left[\frac{(x_t^d - \mu_i^d)^2}{(\sigma_i^d)^3} - \frac{1}{\sigma_i^d} \right]. \quad (12)$$

where the superscript d denotes the d -th dimension of a vector and the occupancy probabilities $\gamma_t(i)$ are computed using the formula (5).

Note that (10) is defined for $i \geq 2$ as there are only $(N - 1)$ free weight parameters due to the constraint $\sum_{i=1}^N w_i = 1$, (w_1 was supposed to be given knowing the value of the other weights). The gradient vector is just a concatenation of the partial derivatives with respect to all the parameters.

Finally, the gradient vectors are normalized according to (8). See for closed form approximations of $F_{\lambda}^{-1/2}$ in [31].

3.6 Fusion

3.6.1 Fusion for queries with multiple images

As a visual query is represented by several images, we decided to first compute relevance scores for each image of the query. Then, the score of an image with respect to a multi-image query is derived from the set of scores between this image and all the query images. For this current work, the max was chosen to be the function for mixing the different scores. Others fusion techniques from cross lingual information retrieval [16], or distributed information retrieval could be a good source of inspiration too.

3.6.2 Fusion of Text and Image

Once a unique score between one image and a query is obtained, this score is then renormalized to be mixed with the text score. The renormalized score is then linearly interpolated with the text score. The coefficient of linear interpolation has been roughly estimated on the blank data set to optimize the mean average precision.

4. EXPERIMENTAL RESULTS

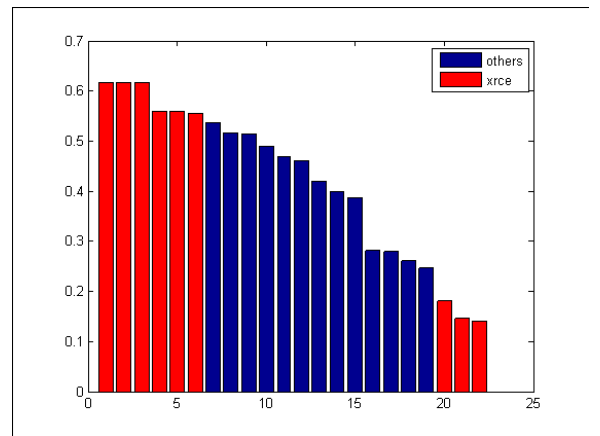


Figure 3: The overall results.

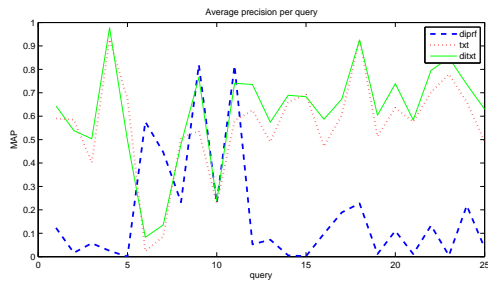


Figure 4: The overall results.

The visual vocabulary (GMM) is estimated using the whole image database. We found that a vocabulary of 1024 visual words is a good compromise between computational cost and retrieval accuracy.

The performance as measured by the Mean Average Precision (MAP) for all queries is illustrated on Figure 3. Referring to the run numbering in Figure 3 (by decreasing order of MAP), the different settings of XRCE’s runs are:

- Run 1** Late fusion of textual and image relevance scores; textual relevance scores are obtained using the LM-based approach, with Jelinek-Mercer smoothing; visual relevance scores are computed using the same approach (LM-based with Jelinek-Mercer smoothing) on a bag-of-visual keyword representation using the concatenation of the (discretized) colour and texture histogram, but with an extra pseudo-relevance feedback step.
- Run 2** Idem as *Run 1*, but without the pseudo-relevance feedback step for image retrieval.
- Run 3** Idem as *Run 1*, but with a “hierarchical” fusion approach: we keep the most relevant textual documents (TOP 300), but re-rank them using the visual relevance scores; this fusion method, that we designed to gain more robustness, turned out not to give better results than a standard, linear combination method.
- Run 4** Idem as *Run 2*, but with a Fisher-kernel representation of the image, concatenating both colour and texture components as a single vector and using cosine similarity measure between these vectors; in this run, the number of components in the GMM was chosen to be 64.
- Run 5** This is a purely textual run, using the LM-based approach, with Jelinek-Mercer smoothing.
- Run 6** Idem as *Run 4*, with the number of components in the GMM chosen to be 128.
- Run 20** Purely visual run: relevance scores are computed using the LM-based retrieval approach with Jelinek-Mercer smoothing on the bag-of-visual keyword representation using the concatenation of the (discretized) colour and texture histogram with an extra pseudo-relevance feedback step (same as *Run 1*, without any textual information).
- Run 21** Same as *Run 7*, but without the pseudo-relevance feedback step.
- Run 22** Purely visual run, using a Fisher-kernel representation of the image, concatenating both colour and texture components as a single vector and using cosine similarity measure

between these vectors (number of components in the GMM = 64).

Note that, experimentally, pseudo-relevance feedback (PRF) on the textual part did not give better results; that’s why it does not appear in these results. On the contrary, PRF helps significantly for purely visual retrieval, using the bag of visual word histogram representation. Approach of *Run 1* turned out to be the best one, for the ImageEval corpus and queries, but its difference in performance with respect to the second and third ones is actually not significant.

Figure 4 gives the detail of performance by query for 3 Runs. The lower curve (dashed - bold) corresponds to Run 20 (pure visual), the medium one (dotted) to Run 5 (pure textual) and the upper one to Run 1. It can be seen that, even when purely visual performance is low, the hybrid text-image approach enhances the results of the purely textual search.

Our results show that our text system is better than any other run proposed by the other participants. We believe that this is mainly due to the fact of our lemmatizer as well as to the use of language models for text modelling. Our leading run shows that our fusion of text and image, although not perfect, gave an improvement of 8% over the text. Learning what is a relevant image is hard task, given the diversity of images queries and we believe that text could provide more data to adapt an interesting image similarity through the use of relevance feedback techniques.

5. CONCLUSION

Exploiting interesting correlations between text and image in order to enhance multimedia retrieval performance actually raises numerous issues: firstly, the relationship between text and image is not necessarily a mutually descriptive / illustrative relationship; only a part (if any) of the image or of the associated text could have some correspondence. Secondly, even with an intermediate image representation such as “bag of visual kwords”, there is still a significant difference in the semantic expressiveness of the adopted representation, the visual one being in any case at a lower level than the textual one, so that mixing both representation at an early stage is not necessarily useful, neither meaningful.

It appeared that, with our current settings and the Imageeval corpus, no “early fusion” approach gave significantly better results than a “late fusion” method. In this track, exploiting textual information with the Language Modelling approach alone, already offered very satisfying performance, much larger than purely visual search. Still, late fusion was able to increase monomedia results by more than 10% (relative), showing the usefulness of combining both types of information, even if the purely visual retrieval component gives relatively poor results.

Future works will be focused on two axes. The first one is to better understand and classify the nature of the relationship between the image and its associated text, so that some “filtering” of the textual / visual content could be done to extract more significant links — or cross-modal translations — between the two. The second one is to explore hybrid pseudo-relevance feedback methods, where both worlds (image and text) will share their initial retrieval results to mutually enrich the initial, mono-modal representation of the query.

6. ACKNOWLEDGMENTS

This work was partly funded by the French Government under the *Infomagic* project, part of Pole CAP DIGITAL (IMVN) de Paris, Ile-de-France

7. REFERENCES

- [1] G. Amati and C. J. V. Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357–389, 2002.
- [2] A. Amir, J. Argillander, M. Berg, S.-F. Chang, M. Franz, W. Hsu, G. Iyengar, J. Kender, L. Kennedy, C.-Y. Lin, M. Naphade, A. Natsev, J. Smith, J. Tesic, G. Wu, R. Yang, and D. Zhang. IBM research TRECVID-2004 video retrieval system. In *Proc. of TREC Video Retrieval Evaluation*, 2004.
- [3] K. Barnard, P. Duygulu, D. B. D. Forsyth, N. de Freitas, and M. Jordan. Matching words and pictures. *J. of Machine Learning Research*, 3, 2003.
- [4] A. Benitez and S.-F. Chang. Image classification using multimedia knowledge networks. In *ICIP*, 2003.
- [5] D. Blei, Michael, and M. I. Jordan. Modeling annotated data. In *ACM SIGIR*, 2003.
- [6] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer. The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, 19(10), 1993.
- [7] P. Carbonetto, N. de Freitas, and K. Barnard. A statistical model for general contextual object recognition. In *ECCV*, 2004.
- [8] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *JMLR*, 5, 2004.
- [9] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning for Computer Vision*, 2004.
- [10] P. Duygulu, K. Barnard, J. de Freitas, and D. Forsyth. Object recognition as machine translation :learning a lexicon for a fixed image vocabulary. In *ECCV*, 2002.
- [11] J. Farquhar, S. Szedmak, H. Meng, and J. Shawe-Taylor. Improving “bag-of-keypoints” image categorisation. Technical report, University of Southampton, 2005.
- [12] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *CVPR*, volume 2, 2005.
- [13] S. Feng, V. Lavrenko, and R. Manmatha. Multiple bernoulli relevance models for image and video annotation. In *CVPR*, 2004.
- [14] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Advances in Neural Information Processing Systems 11*, 1999.
- [15] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *ACM SIGIR*, 2003.
- [16] J. Savoy. Report on clef-2003 multilingual tracks. In *Report on CLEF 2003- Working Notes for the Workshop*, 2003.
- [17] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *IEEE Int. Conf. on Computer Vision*, 2005.
- [18] S. Kosinov and S. Marchand-Maillet. Hierarchical ensemble learning for multimedia categorization and autoannotation. In *IEEE Signal Processing Society Workshop MLSP*, 2004.
- [19] S. Kosinov, S. Marchand-Maillet, and I. Kozintsev. Dual diffusion model of spreading activation for content-based image retrieval. In *ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2006.
- [20] V. Lavrenko, S. Feng, and R. Manmatha. Models for automatic video annotation and retrieval. In *ICASSP*, 2004.
- [21] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *NIPS*, 2003.
- [22] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *PAMI*, 25:9, 2003.
- [23] Y. Li, J. A. Bilmes, and L. G. Shapiro. Object class recognition using images of abstract regions. In *ICPR*, 2004.
- [24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2), 2004.
- [25] R. Marée, P. Geurts, J. Piater, and L. Wehenkel. Random subwindows for robust image classification. In *CVPR*, volume 1, 2005.
- [26] F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *ACM MM*, 2003.
- [27] F. Monay and D. Gatica-Perez. Plsa-based image auto-annotation: Constraining the latent space. In *ACM MM*, 2004.
- [28] Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *In MISRM'99 First International Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.
- [29] E. Novak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*, 2006.
- [30] J. Pan, H. Yang, C. Faloutsos, and P. Duygulu. Gcap: Graph-based automatic image captioning. In *CVPR Workshop on Multimedia Data and Document Engineering*, 2004.
- [31] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007.
- [32] F. Perronnin, C. Dance, G. Csurka, and M. Bressan. Adapted vocabularies for generic visual categorization. In *ECCV*, 2006.
- [33] J. Ponte and W. 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.
- [34] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. V. Gool. Modeling scenes with local descriptors and latent aspects. In *ICCV*, 2005.
- [35] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241. Springer-Verlag New York, Inc., 1994.
- [36] J. S. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, volume 2, 2003.
- [37] M. Srikanth, J. Varner, M. Bowden, and D. Moldovan. Exploiting ontologies for automatic image annotation. In *ACM SIGIR*, 2005.
- [38] A. Vinokourov, D. R. Hardoon, and J. Shawe-Taylor. Learning the semantics of multimedia content with application to web image retrieval and classification. In *Fourth International Symposium on Independent Component Analysis and Blind Source Separation*, 2003.
- [39] A. Vinokourov, J. Shawe-Taylor, and N. Cristianini. Inferring a semantic representation of text via cross-language correlation analysis. In *Advances of Neural Information Processing Systems*, volume 15, 2002.
- [40] X. Wang, L. Zhang, and W.-Y. M. F. Jing. Annosearch: Image auto-annotation by search. In *CVPR*, 2006.
- [41] C. Zhai and J. Lafferty. A study of smoothing methods for

language models applied to ad hoc to information etrieval. In *Proceedings of SIGIR'01*, pages 334–342. ACM, 2001.