CEA LIST's participation to Visual Concept Detection Task of ImageCLEF 2011

Amel Znaidia, Hervé Le Borgne, and Adrian Popescu

CEA, LIST, Laboratory of Vision and Content Engineering
18 route du Panorama, BP6, Fontenay-aux-Roses, F-92265 France

amel.znaidia@cea.fr , herve.le-borgne@cea.fr , adrian.popescu@cea.fr

Abstract. This paper describes the CEA LIST participation in the ImageCLEF 2011 Photo Annotation challenge. This year, our motivation was to investigate the annotation performance by using provided Flickr-tags as additionnal information. First, we present an overview of our local and global visual features used in this work. Second, we present a new method, that we call *"Fuzzy-tfidf"*, which takes into account the uncertainty of user tags. Our textual descriptor is based on semantic similarity between tags and visual concepts. To compute this similarity, we used two distances: the first one is based on Wordnet ontology and the second is based on social networks. We perform a late fusion to combine scores from visual and textual modalities.
Our best model, a late fusion trained on global visual features and user tags, obtains 38.3 % MAP, almost a 8 % MAP absolute improvement compared to our best visual-only system. The results show that the combination of Flickr-tags with visual features improves the results of the run using only visual features. It corroborates the importance of taking into account the uncertainty of user tags and the complementarity between visual and textual modalities.

1 Introduction

The ImageCLEF Photo Annotation Task [11] is a multi-label classification problem, with 8.000 image for training, 10.000 for testing and 99 concepts to detect. The image are extracted from the MIR Flickr dataset [6] and the Flickr user tags and/or EXIF information are available for most photos.

In our participation to the ImageCLEF Photo Annotation Task, we focus on how to use the tags associated to the images to enhance the annotation performance. We propose three different models: visual only, textual only and multimodal models. The last model takes the mean of the predicted score of the textual and visual classifiers.

This paper is organized as follows. In Section 2 we describe our local and global visual features. In Section 3 we give an overview of our *"Fuzzy-tfidf"* method which uses user tags. Then in Section 4 we present in more detail the experiments we did, the submitted runs and the obtained results. Finally, we conclude the paper in Section 5.

2 Visual features

We used two sets of descriptors, named *fklsp* and *piria5* in the following.

2.1 Local descriptors (*fklsp*)

This set was based on a non parametric estimation of Fisher vector to agregate local descriptors, as explained in detail in [1].

Fisher kernel, score and vector Let $X = \{x_t, t = 1 \dots T\}$ a set of vectors used to describe an image (*i.e* a collection of local features). It can be seen as resulting from a generative probability model with density $f(X|\theta)$. To derive a kernel function from such a generative model, being able to exhibit discriminative properties as well, Jaakola [8] proposed to use the gradient of the log-likelihood with respect to the parameters, called the *Fisher score*:

$$U_X(\underline{\theta}) = \nabla_{\underline{\theta}} \log f(X|\underline{\theta}) \tag{1}$$

This transforms the variable length of the sample X into a fixed length vector that can feed a classical learning machine. In the original work of [8] the Fisher information matrix F_λ is suggested to normalize the vector:

$$F_\lambda = E_X[\nabla_{\underline{\theta}} \log f(X|\underline{\theta}) \nabla_{\underline{\theta}} \log f(X|\underline{\theta})^T] \tag{2}$$

It then results into the Fisher vector:

$$G_X(\underline{\theta}) = F_\lambda^{-1/2} \nabla_{\underline{\theta}} \log f(X|\underline{\theta}) \tag{3}$$

Density estimation The traditional way of modeling a distribution density is to assume a classical parametric model such as normal, gamma or Weibull. For instance in [12], the vocabularies of visual words are represented with a Gaussian Mixture Models, for which the parameters (weight, mean and variance of each Gaussian) are estimated by maximum likelihood.

Alternatively, we can use a nonparametric estimate of the density, such as a histogram or a kernel-based method [17]. A histogram density estimation can be seen as modeling the unknown log-density function by a piecewise constant function and estimating the unknown coefficients by maximum likelihood. In this vein, Kooperberg [10] proposed to model the log-density function by cubic spline, resulting into the so-called logspline density estimation.

Let consider the space $\mathcal{S}$ consisting of the twice-continuously differentiable function f_s (natural cubic splines), such that the restriction of f_s to some intervals $[t_1, t_2] \dots [t_{K-1}, t_K]$ is a cubic polynomial and linear at the extremities. Let $1, B_1, \dots, B_p$ a set of basis functions that span the space $\mathcal{S}$. Given $\underline{\theta} = (\theta_1, \dots, \theta_p) \in \mathbb{R}^p$ such that:

$$\int_L^U \exp\left(\theta_1 B_1(y) +, \dots, \theta_p B_p(y) dy\right) < \infty \tag{4}$$

We can thus consider the exponential family of distribution based on this basis function:

$$f(y, \underline{\theta}) = \exp\left(\theta_1 B_1(y) +, \dots, \theta_p B_p(y) - \mathcal{C}(\underline{\theta})\right) \tag{5}$$

Where $\mathcal{C}(\underline{\theta})$is a normalizing constant such that $f(y, \underline{\theta})$ is a density. As shown in [10], it is possible to determine the maximum likelihood estimate of $\underline{\theta}$ with a Newton-Raphson method with step-halving.

Signature derivation Each feature dimension x^i ($i \in [1 \dots D]$) of a local descriptor can be thought of as arising as a random sample from a distribution having a density h^i for a particular image and f^i for a set of images. Modelling the log-density function by a cubic spline and deriving the corresponding Fisher score lead to [1]:

$$\left.\frac{\partial \mathcal{L}(Y, \underline{\theta})}{\partial \theta_j^i}\right|_{\theta_j^i \approx \widehat{\theta_j^i}} = E_{h^i}\left[B_j^i(y)\right] - E_{f^i}\left[B_j^i(y)\right] \tag{6}$$

Where $h^i(.)$ is the density of the image descriptor and $f^i(.)$ the density class descriptor (dimension i), this last being estimated from local descriptor extracted from several learning images. The full gradient vector $U_Y(\underline{\theta})$ is a concatenation of these partial derivatives with respect to all parameters. Its number of components is $\sum_{i=1}^{D} p^i$, where p^i is the number of non-constant polynomial of the basis of $\mathcal{S}$ for dimension i.

The equation (6) simply reflects the way a specific image (with density h^i) differs from the average world (*i.e* density f^i), *through* a well chosen polynomial basis, at each dimension. The *average world* $\left(E_{f^i}\left[B_j^i(y)\right]\right)$ can be seen as a codebook. If one uses linear polynomials $\left(B_j^i(y) = \alpha_j y^i\right)$, equation (6) relates to the VLAD signature[9], with an important difference since all vectors are used (i) during learning to estimate the codeword (ii) during test to compute the signature, while (i) K-means uses the closest vectors of a codeword (cluster center) to re-estimate it at each step (ii) VLAD uses only nearest neighbours to compute the signature component (see eq. (1) in [9]).

In his seminal work, Jaakola [8] proposed to normalize the Fisher score by the Fisher information matrix. In [12], it was noted that such an operation improved the efficiency of the method in term of discrimination, by normalizing the dynamic range of the different dimensions of the gradient vector. Although some normalisation of the signature were proposed in [1], they were not used in this work.

Efficient implementation The set of basis functions $1, B_1, \dots, B_p$ that span the space $\mathcal{S}$ introduced in section 2.1, are defined according to intervals $[t_i, t_{i+1}]$ ($i \in [1 \dots K]$), where the t_i are named *knots*. We fixed a given number of knots and placed them according to statistic order of the learning data. Hence, at each dimension, the amount of information is regularly distributed between knots. For low-level features such as those presented in section 2.1, the knots are approximately placed according to a logarithmic distribution.

Several choices are possible to defined the basis B_k. In this work, we used the following basis, that is very efficient to implement:

$$\begin{aligned} &B_0(y) = 1 \text{ (not used)} \\ &B_1(y) = y \\ &B_{k>1}(y) = \begin{cases} \frac{|y-t_k|+y-t_k}{2} & \text{for } y < t_{k+1} \\ 0 & \text{for } y > t_{k+1} \end{cases} \end{aligned} \tag{7}$$

Such an implementation is equivalent to compute only $(y - t_k)$ on the interval $[t_k, t_{k+1}]$ since the polynomial is null elsewhere and $y > t_k$ on the interval. Moreover, we used a binary weighting scheme, that does not consider the value of $|y - t_k|$ in the computation

but only its existence. In other word, one can only count $+1$ each time a pixel activity y is between t_k and t_{k+1}. Such a binary weighting scheme is commonly used in the design of BOV, in particular when the codebook is large [18].

Independent low level features According to theory, the signature derivation requires to use independent low-level features, such that the image description density could be expressed as a factorial code. Such features can be obtained with Independent Component Analysis (ICA) [4, 7] that is a class of methods that aims at revealing statistically independent latent variables of observed data. In comparison, the well-known Principal Component Analysis (PCA) would reveal uncorrelated sources, *i.e* with null moments up to the order two only. In its simplest form, ICA defines a generative model which consider multivariate data X as a linear mixtures of some unknown sources S, and the mixture A is also unknown. Under the assumption that the sources are mutually independent and at most one is Gaussian, [4] showed that it is possible to solve this ill-posed problem. For this one must compute a separating matrix w that lead to an estimate Y of the sources:

$$Y = WX = WAS \tag{8}$$

Many algorithms were proposed to achieve such an estimation, that are well reviewed in [7]. These authors proposed the fast-ICA algorithm that searches for sources that have a maximal nongaussianity. When applied to natural image patches of fixed size (*e.g* $\Delta = 16 \times 16 = 256$), ICA results into a generative model composed of localized and oriented basis functions [7]. Its inverse, the separating matrix, is composed of *independent* filters $w_1, \ldots, w_D$ (size Δ) that can be used as feature extractors, giving a new representation with mutually independent dimensions. The number of filters (D) extracted by ICA is less or equal to the input data dimension (Δ). This can be reduced using a PCA previously to the ICA. The responses of the D filters to some pixels $(p_1, \ldots, p_T)$ of an image $I(.)$ are thus independent realizations of the D-dimensional random vector Y. As a consequence, the density can be factorized as expected:

$$h_{ica}\left(I(p_t)\right) = \prod_{i=1}^{D} h_{ica}^{i}\left(I(p_y)\right) = \prod_{i=1}^{D} w_i * I(p_t) \tag{9}$$

Where $*$ is the convolution product. These independent low-level features can be further used according to the method presented into section 2.1.

2.2 Global descriptors (*piria5*)

We concatenated five descriptors to form a single global descriptor of size 1341:

- A descriptor that is itself the concatenation of a Local Edge Pattern (LEP) descriptor (derived from [3]) and a color histogram, with a global normalisation on the 576 dimensions.
- A compact histogram that count how many pixels are 4-connected according to their colors [14].
- A classic color histogram of size 64.

– A RGB color histogram of size 125.
– A HSV color histogram.

The first descriptor (LEP) gives a piece of information on the *texture* of the image and the second a weak one on the *spatial organisation* of the pixels. All other descriptors mainly give information on the *colors* present in the image.

3 Textual features

To improve visual concept annotation, image associated tags can be used. The key idea is to project the tags in the visual concept space. Each tag will be associated with one or more concepts according to their semantic similarities. In this manner, the concept voted by several tags is then considered appropriate to describe the content of the image. For example in Fig. 1, *"strawberry, sugar, spoon, frutella, fresa"* will be associated with the visual concept *"food"* which will be relevant to this image. To compute the similarity

Fig. 1. An example of image with its associated tags.

between user tags and visual concepts, we use two different distances. The first one is based on *Wordnet* ontology and the second is based on social networks.

3.1 Semantic similarity

Wordnet-based similarity First, we rely on the Wu-Palmer measure [19], which provides a similarity function for two given concepts, defined by how closely they are related in the hierarchy, *i.e.*, their structural relations as shown in Fig. 2.

The conceptual similarity between two concepts C_1 et C_2 is given by:

$$ConSim(C_1, C_2) = \frac{2 * N_3}{N_1 + N_2 + 2 * N_3} \tag{10}$$

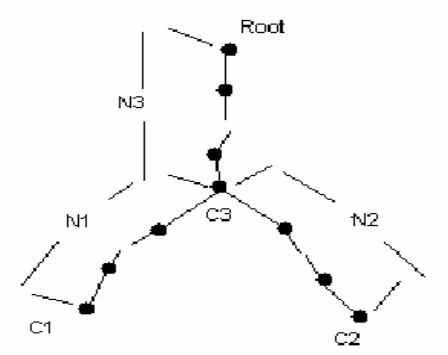

Fig. 2. The concept similarity measure.

Where C_3 is the least common superconcept of C_1 and C_2. N_1, N_2 and N_3 represent respectively the number of nodes on the path from C_1 to C_3, from C_2 to C_3 and from C_3 to $Root$. This measure is based on *WordNet* structure [5]. This can be seen as a semantic network where each node represents a concept of the real world. Each node consists of a set of synonyms that represent the same concept, this set is called *synset*. These *synsets* are connected by arcs that describe relations between concepts. This measure is defined between two synsets s_1 and s_2 by:

$$sim_{wup}(s_1, s_2) = \frac{2 * depth(lcs(s_1, s_2))}{depth(s_1) + depth(s_2)} \tag{11}$$

where $lcs(s_1, s_2)$ denotes the least common subsumer (most specific ancestor node) of the two synsets s_1 and s_2 in a *WordNet* taxonomy, and $depth(s)$ is the length of the path from s to the taxonomy $Root$. Since a word can belong to more than one synset in *WordNet* that is, it can have more than one conceptual meaning. We opt to determine the similarity between tags and concepts as the maximum similarity between all their *synsets*. Let $syns(t)$ denotes the set of synsets that contain the tag t, we define the similarity between a tag t_k and a concept C_i as:

$$sim_{Wordnet}(t_k, C_i) = max\{sim_{wup}(s_k, s_i) | (s_k, s_i) \in syns(t_k) X syns(C_i)\} \tag{12}$$

Flickr-based similarity Second, we rely on the work of Popescu et *al.* [13] to define a semantic measure between tags and visual concepts according to their social relatedness. Given two terms T and Q, their social relatedness is defined as follows:

$$SocRel(T, Q) = users(Q, T) * \frac{1}{log(pre(T))} \tag{13}$$

where $users(Q, T)$ is the number of distinct users which associate tag T to a query Q; and $pre(T)$ is the number of distinct users from a prefetched subset of Flickr users

that have tagged photos with tag T. The model obtained from Flickr for a tag t_k can be expressed by:

$$M_{Flickr}(t_k) = \cup_{x=1}^{N}(weight(T_x), T_x) \tag{14}$$

where N is the number of retained Flickr socially related tags and $weight(T_x)$ is the social normalized social weight of T_x using relation (13). In this context, we define a semantic similarity between a tag t_k and a visual concept C_i as:

$$sim_{Flickr}(t_k, C_i) = \frac{dot(t_k, C_i)}{norm(t_k) * norm(C_i)} \tag{15}$$

where $dot(.,.)$ represents the scalar product and $norm(.)$ the vector norm.

3.2 Textual descriptors

Term weighting is a key method in the context of text classification. As in the vector space model introduced by Salton et *al.* [16] to represent text document, we represent the visual concepts as a vector of weights $(w_{i,1}, ..., w_{i,j}, ..., w_{i,|C|})$. In our case, the weight $w_{i,j}$ for a considered concept C_i in a document d_j is obtained by the product of $tf_{i,j}$ and idf_i. The term frequency characterizes the frequency of a concept in the given image and it is calculated as follows:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{16}$$

where $n_{i,j}$ is the number of occurrences of the considered concept C_i in document d_j and the denominator is the sum of number of occurrences of all visual concepts in the document. The inverse document frequency is a measure of the general importance of the visual concept and it is given by:

$$idf_i = log(\frac{|D|}{|j : C_i \in d_j|}) \tag{17}$$

where $|D|$ is the total number of images in the corpus and $|j : C_i \in d_j|$ is the number of images where the concept C_i appears. In this manner, we should perform a hard assignment to determine the presence or the absence of a concept (1 or 0). Or the semantic similarity between tags and a visual concept is not equal. Moreover, users usually do not use the same visual concepts to tag their photos. Then, it is more appropriate to proceed on a soft assignment in which a tag is matched to a visual concept with some confidence value. This confidence value represents the uncertainty of the presence of a visual concept. Ideally, if the user use the same visual concept to tag his photo, this value is equal to 1. Else, it is a value between 1 and 0 depending on how similar they are.

In this context, we propose a new version of $tfidf$, that we call *"Fuzzy-tfidf"*. In this method, instead of hard assignment of a concept to a given tag, we add a confidence score. This score is the semantic similarity between a tag t_k and a concept C_i using formulas (12) or ccc. In this way, we take into account the ammount of similarity between tags and visual concepts. Let $s_{k,i}$ denotes the conceptual similarity between a tag t_k

and visual concept C_i. T represents the set of tags in document d_j. C and D represent respectively the set of visual concepts and documents in the dataset. The fuzzy term frequency is obtained by:

$$fuzzy - tf_{i,j} = \frac{\sum_{k \in T} s_{k,i}}{\sum_{i \in C} \sum_{k \in T} s_{k,i}} \tag{18}$$

The *Fuzzy inverse document frequency* is computed as follows:

$$fuzzy - idf_i = log(\frac{|D|}{\sum_{j \in D} \frac{\sum_{k \in T} s_{k,i}}{n_{i,j}}}) \tag{19}$$

where $n_{i,j}$ is the number of occurrences of the considered concept C_i in document d_j. The $fuzzy - tfidf$ is obtained by the product of the obove two frequencies. In case $s_{k,i}$ is equal to 1, we found the same formula as the classic $tfidf$. In this method, we consider only concepts that are similar to the considered tag in a neighborhood. This neighborhood is deterimed by cross-validation.

4 Experiments

4.1 Dataset

We evaluated our annotation methods on the MIR Flickr dataset [6] containing 8.000 images for training and 10.000 for testing belonging to 99 concept classes. This year a special focus is laid to the detection of sentiment concepts (funny, scary, unpleasant, active, happy ...). Fig. 3 shows samples of images taken from the ImageCLEF 2011 Photo Annotation Task Dataset with their annotated concepts.

4.2 Submitted runs

We submitted five runs to the campaign, allowing relevant comparison between the methods:

CEALIST_text_Fsb use only the textual feature computed according to the method presented in section 3.2. In this run, we use the Wordnet-based semantic similarity and we use the *Fuzzy-tfidf* to compute the textual descriptor. Two terms are considered semantically similar if their $sim_{Wordnet}$ is upper than a threshold α obtained by cross validation. In our experiments, this threshold is fixed to 0.8. The Fast Shared Boosting algorithm [2] is applied for classification.

CEALIST_piria5_FsbRdsa use only global visual descriptors presented in section 2.2. It is the concatenation of five descriptors of color and texture.

CEALIST_piria5_FsbRdsa_text is a multimodal run where a late fusion process is performed. Scores of the late fusion are obtained by averaging the scores of the two previous runs. Both visual and textual scores are normalized before fusion.

CEALIST_piria5_FsbRdsa_text2 is our best run and it is also a multimodal one. It is a late fusion of the visual scores of the second run and the textual scores of the same method as the first run but this time using Flickr-based similarity.

CEALIST_fklsp_FsbRdsa_text2 is a multimodal run where a late fusion process is performed between a visual classifier based on local visual descriptors presented in section 2.1 and the same textual classifier used in the previous run.

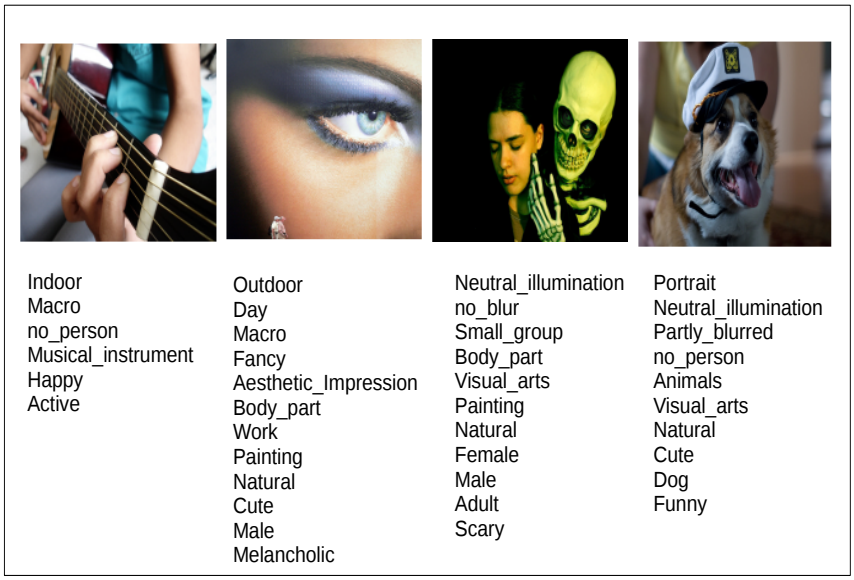

Fig. 3. Samples of images taken from the ImageCLEF 2011 Photo Annotation Task Dataset with their annotated concepts.

4.3 Analysis of the Results

Performance evaluation To determine the quality of the annotations five measures were used, three for the evaluation per concept and two for the evaluation per photo. For the concept based evaluation the mean Average Precision (MAP), the equal-error-rate (EER), and the area-under the curve (AUC) are used, using the confidence scores. For the example based evaluation, F-measure (F-ex) [15] and Semantic R-Precision (SR-Precision) are used. The SR-Precision is a novel performance measure derived from the example-based R-Precision measure. In contrast to R-Precision, it considers the Flickr Tag Similarity measure to determine the semantic relatedness of misclassified concepts.
Overview of Results In Table 1, we list the performance of our submitted runs.

Run	Modality	MAP	EER	AUC	F-ex	SR-Pr
1: CEALIST_text_Fsb	T	0.292	0.356	0.684	0.478	0.675
2: CEALIST_piria5_FsbRdsa	V	0.300	0.290	0.774	0.503	0.700
3: CEALIST_piria5_FsbRdsa_text	V& T	0.372	0.259	0.808	0.497	0.704
4: CEALIST_piria5_FsbRdsa_text2	V& T	**0.383**	**0.250**	0.**819**	**0.508**	**0.710**
5: CEALIST_fklsp_FsbRdsa_text2	V& T	0.347	0.283	0.784	0.484	0.693

Table 1. Overview of the different submissions.

We can notice that the use of user tags improves significantly the results ($\approx$ 8 % MAP). The textual run (run 1) based on *WordNet* gave the same MAP as our best visual only run (run 2). We tested also run 1 with the Flickr-based similarity and it gaves 0.31 MAP. Fig. 4 and 5 show the Average Precision (AP) of our best ImageCLEF run through different classes.

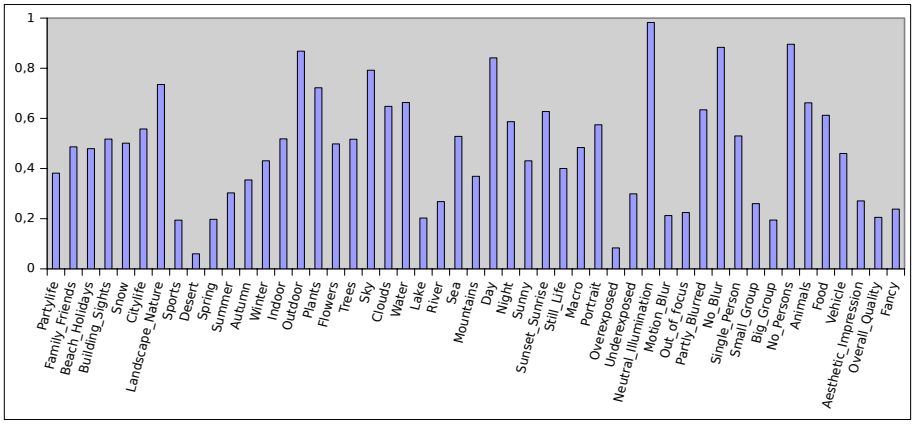

Fig. 4. The Average Precision per concept.

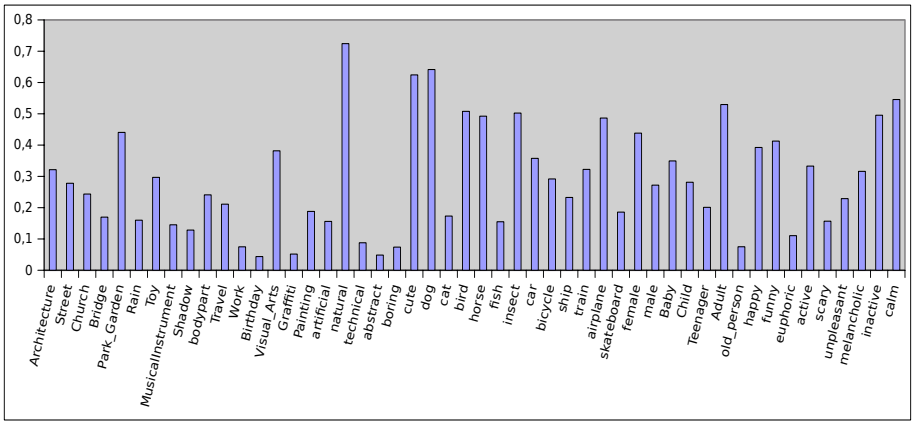

Fig. 5. The Average Precision per concept.

5 Conclusion

Our goal for the ImageCLEF 2011 Photo Annotation challenge was to take advantage of the available user tags as additional information. Results have shown that all our methods combining visual and textual modalities outperform our visual only classiers. Our best scoring classier obtains 38.3 % in MAP, ≈ 8 % higher than our best visual-only system.

6 Acknowledgment

This work was supported by grants from DIGITEO and Rgion Ile-de-France.

References

1. Le Borgne, H., Muñoz-Fuentes, P.: Nonparametric estimation of fisher vectors to aggregate image descriptors. In: Proc ACIVS. Lecture Notes in Computer Science 6915, Ghent, Belgium (2011)
2. Borgne, H.L., Honnorat, N.: Fast shared boosting: Application to large-scale visual concept detection. In: Quénot, G. (ed.) International Workshop on Content Based Multimedia Indexing, CBMI. pp. 13–18. Grenoble, France (2010)
3. Cheng, Y.C., Chen, S.Y.: Image classification using color, texture and regions. Image Vision Computing (2003)
4. Comon, P.: Independent component analysis, a new concept? Signal Processing 36(3), 287–314 (1994)
5. Fellbaum, C. (ed.): WordNet: An Electronic Lexical Database (Language, Speech, and Communication). The MIT Press (May 1998)
6. Huiskes, M.J., Lew, M.S.: The mir flickr retrieval evaluation. In: MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval. ACM, New York, NY, USA (2008)
7. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. Wiley-Interscience (May 2001)
8. Jaakola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: NIPS. pp. 1–8 (1999)
9. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: CVPR. San Francisco, USA (june 2010)
10. Kooperberg, C., Stone, C.J.: Logspline density estimation for censored data. Journal of Computational and Graphical Statistics 1, 301–328 (1997)
11. Nowak, S., Nagel, K., Liebetrau, J.: The clef 2011 photo annotation and concept-based retrieval tasks. In: CLEF 2011 working notes (2011)
12. Perronnin, F., Dance, C.R.: Fisher kernels on visual vocabularies for image categorization. In: CVPR. pp. 1–8 (2007)
13. Popescu, A., Grefenstette, G.: Social media driven image retrieval. In: Proceedings of the 1st ACM International Conference on Multimedia Retrieval. pp. 33:1–33:8. ICMR '11, ACM, New York, NY, USA (2011)
14. R. O. Stehling, M. A. Nascimento, A.X.F.: A compact and efficient image retrieval approach based on border/interior pixel classification. In: Proceedings of the eleventh international conference on Information and knowledge management. pp. 102–109. McLean, Virginia, USA (2002)

15. van Rijsbergen, C.J.: Information Retrieval. Butterworths, London, 2 edn. (1979)
16. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM 18(11), 613–620 (November 1975)
17. Silverman, B.W.: Density estimation for statistics and data analysis. Chapman and Hall (1986)
18. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: ICCV. pp. 1470–1477 vol.2 (April 2003)
19. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: 32nd. Annual Meeting of the Association for Computational Linguistics. pp. 133 –138. New Mexico State University, Las Cruces, New Mexico (1994), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.14.1869