MRIM-LIG at ImageCLEF 2009: Robotvision, Image annotation and retrieval tasks

Trong-Ton Pham, Loïc Maisonnasse, Philippe Mulhem, Jean-Pierre Chevallet, Georges Quénot, Rami Albatal

MRIM-LIG at ImageCLEF 2009: Robotvision, Image annotation and retrieval tasks

Trong-Ton Pham[1], Loïc Maisonnasse[2], Philippe Mulhem[1], Jean-Pierre Chevallet[1], Georges Quénot[1], Rami Al Batal[1]
Trong-Ton.Pham@imag.fr, Philippe.Mulhem@imag.fr,
Jean-Pierre.Chevallet@imag.fr
Georges.Quenot@imag.fr, Rami.Albatal@imag.fr

[1]Laboratoire Informatique de Grenoble (LIG), Grenoble University, CNRS, LIG -
[2]Laboratoire d'InfoRmatique en Image et Systemes d'information (LIRIS)

Abstract. This paper describes mainly the experiments that have been conducted by the MRIM group at the LIG in Grenoble for the the ImageCLEF 2009 campaign, focusing on the work done for the Robotvision task. The proposal for this task is to study the behaviour of a generative approach inspired by the language model of information retrieval. To fit with the specificity of the Robotvision task, we added post-processing in a way to tackle with the fact that images do belong only to several classes (rooms) and that image are not independent from each others (i.e., the robot cannot in one second be in three different rooms). The results obtained still need improvement, but the use of such language model in the case of Robotvision is showed. Some results related to the Image Retrieval task and the Image annotation task are also presented.

1 Introduction

We describe here the different experiments that have been conducted by the MRIM group at the LIG in Grenoble for the ImageCLEF 2009 campaign, and more specifically for the Robotvision task. Our goal for this task was to study the use of language models in the context where we try to guess in which room a robot is in a partially known environment. Language models for text retrieval where proposed ten years ago, and behave very well when all the data cannot be directly extracted from the corpus. We have already proposed such application for image retrieval in [10], achieving very good results. We decided to focus on this challenging task represented by the Robotvision task in CLEF 2009. We also participated to the Image retrieval and the image annotation task for CLEF 2009, and we discuss briefly, because of space constrains, some of our proposal and results.

The paper is organized as follows. First we describe the Robotvision task in section 2, our proposal based on language models and the results obtained. In this section, we focus on the features that were used to represent the images,

before describing the language model defined on such representation and the post-processing that took advantage of the specificity of the Robotvision task. Because the MRIM-LIG research group participated in two other image related tasks, we propose in section 3 to describe shortly our main proposals and finding for the image annotation and the image retrieval tasks. We conclude in section 4.

2 Robovision Track

2.1 Task description

The Robotvision task at CLEF 2009 [1], aims at determining "the topological location of a robot based on images acquired with a perspective camera mounted on a robot platform." A robot is moving on a building floor, going across several (six) rooms, and an automatic process has to indicate, for each image of a video sequence shot by the robot, in which room is the robot. In the test video, a additional room (which was not given in the training set), *unknown*, is present and has also to be tagged automatically. The full video set is the IDOL video database [6].

2.2 Image representation

We have applied a visual language modeling framework for the Robotvision task. This generative model is quite standard in the Information Retrieval field, and already lead to good results for visual scene recognition [10]. Before explaining in detail the language modeling approach, we fix some elements related to the feature extractions of images. To cover the different classes of features that could be relevant, we have extracted color, texture, and region of interest features in our proposal. These features are: **HSV color histogram**: we extract the color information from HSV color space. One image is represented by a concatenation of n×n histograms, according to non overlapping rectangular patches defined from a n×n grid applied on the image. Each histogram has 512 dimensions; **Multi-scale canny edge histogram**: we used Canny operator to detect the contour of objects as presented in [15]. An 80-dimensional vector was used to capture magnitudes and gradient of the contours for each patch. This information is extracted from a grid of m×m for each image; **Color SIFT**: SIFT features are extracted using D. Lowe's detector [5]. Region around the keypoint is described by a 128-dimensional vector for each R, G, B channel. Based on the usual bag of visual words approach, we construct for each of the features above a visual vocabulary of 500 visual words using k-means clustering algorithm. Each visual word is designated to a concept c. Each image will then be represented using theses concepts and the language model proposed is built on these concepts.

2.3 Visual language modeling

The language modeling approach to information retrieval exists from the end of the 90s [11]. In this framework, the relevance status value of a document for

a given query is estimated by the probability of generating the query from the document. Even though this approach was originally proposed for unigrams (i.e. isolated terms), several extensions have been proposed to deal with *n-grams* (i.e. sequences of n terms) [12, 13], and, more recently, with relationships between terms and graphs. Thus, [3] proposes (a) the use of a dependency parser to represent documents and queries, and (b) an extension of the language modeling approach to deal with such trees. [8, 9] further extend this approach with a model compatible with general graphs, as the ones obtained by a conceptual analysis of documents and queries. Other approaches (as [2, 4]) have respectively used probabilistic networks and kernels to capture spatial relationships between regions in an image. In the case of [2], the estimation of the region probabilities relies on an EM algorithm, which is sensitive to the initial probability values. In contrast, in the model we propose, the likelihood function is convex and has a global maximum. In the case of [4], the kernel used only considers the three closest regions to a given region. In [10], we have presented the image as a probabilistic graph which allows capturing the visual complexity of an image. Images are represented by a set of weighted concepts, connected through a set of directed associations. The concepts aim at characterizing the content of the image whereas the associations express the spatial relations between concepts. Our assumption is that the concepts are represented by non-overlapping regions extracted from images. In this competition, the images acquired by the robot are of poor quality, and we decided to not take into account the relationship between concepts. We thus assume that each document image d (equivalent each query image q) is represented by a set of weighted concepts W_C. The concepts correspond to a visual word used to represent the image. The weight of concepts captures the number of occurrences of this concept in image. Denoting C the set of concepts over all the whole collection, W_C can be defined as a set of pairs $(c, w(c, d))$, where c is an element of C and $w(c, d)$ is the number of times c occur in the document image i. We are then in a context similar to usual language model for text retrieval. We rely then on a language model defined over concepts, as proposed in [7], which we refer to as Conceptual Unigram Model. We assume that a query q or a document d is composed of a set W_C of weighted concepts, each concept being conditionally independent to the others. Unlike [7] that computes a query likelihood, we evaluate the relevance status value rsv of a document image d for query q by using a generalized formula, the negative Kullback-Leiber divergence, noted $\mathcal{D}$. Such divergence is computed between two probability distributions: the document model M_d computed over the document image d and the query model M_q computed over the query image q. Assuming the concept independence hypothesis, this leads to:

$$RSV_{kld}(q, d) = -\mathcal{D}\left(M_q \| M_d\right) \tag{1}$$

$$\propto \sum_{c_i \in C} \log(P(c_i|M_q) * P(c_i|M_d)) \tag{2}$$

where $P(c_i|M_d)$ and $P(c_i|M_q)$ are the probability of the concept c_i in the model estimated over the document d and query q respectively. If we assume a multinomial models for M_d and M_q, $P(c_i|M_d)$ is estimated through maximum likelihood (as is standard in the language modeling approach to IR), using Jelinek-Mercer smoothing:

$$P(c_i|M_d) = (1 - \lambda_u)\frac{F_d(c_i)}{F_d} + \lambda_u \frac{F_c(c_i)}{F_c} \tag{3}$$

where $F_d(c)$, representing the sum of the weight of c in all graphs from document image d and F_d the sum of all the document concept weights in d. The functions F_c are similar, but defined over the whole collection (i.e. over the union of all the images from all the documents of the collection). The parameter λ_u helps taking into account reliable information when the information from a given document is scarce. For this part, the quantity $P(c_i|M_q)$ is estimated through maximum likelihood without smoothing on the query. The final result $\mathrm{L}(q_i)$ for one query image i is a list of the images d_j from the learning set ranked according to the $RSV_{kld}(q_i, d_j)$ value.

2.4 Post-processing of the results

As we just mentioned, in this basic case we may associate the query image with the room id of the best ranked image. However, because we represent each image with several features and because we have several images of each room in the training set, we post-process this basic result:

- **Fusion**: an image is represented independently for each feature considered (color with a given grid, texture with a given grid, regions of interest). Each of these representations lead to different matching results using the language model. We choose to make a late fusion of the three results obtained using a linear combination:

$$RSV(Q, D) = \sum_i RSV_{kld}(q_i, d_i) \tag{4}$$

  where Q and D correspond to the image query and documents, and q_i and d_i describe the query and the document according to a feature i.
- **Grouping training images by their room**: assuming that the closest training image of a query image is not sufficient to determine the room because of their intrinsic ambiguity, we propose to group the results of the n-best images for each room. We are then able to compute a ranked list of room RL instead of an image list for each query image:

$$RL_q = [(r, RSV_r(q, r)] \quad with \quad RSV_r(q, r) = \sum_{f_{n-best}(q,r)} RSV(q, d) \tag{5}$$

where r corresponds to a room and f_{n-best} is a function that select the n images with the best RSV belonging to the room r.

- **Filtering the** *unknown* **room**: in the test set of the Robotvision task, we know that one additional room is added. To tackle this point, we assume that if one room r is recognized, then the matching value for r is significantly larger than the matching value for the other rooms, especially compared to the room with the lower matching value. So, if this difference is large ($> \beta$), we consider that there is a significant difference and then we keep the tag r for the image. Otherwise we consider the image room tag as *unknown*. In our experiment, we fixed the threshold β to 0.003 after experiments.
- **Smoothing window**: we exploit the visual continuity in a sequence of images by smoothing the result across the temporal axis. To do that, we use a *flat* (i.e., all the images in the window have the same weight) smoothing window centered on the current image. In the experiments, we choose the width of window $w = 40$ (i.e. 20 images before and after the classified image).

2.5 Validating process

The validation aims at evaluating robustness of the algorithms to visual variations that occur over time due to the changing conditions and human activity. We trained our system with the *night*3 condition set and tested against all the other conditions from validation set. Our objective was to understand the behavior of our system with the changing conditions and with different types of features. We first study the models one by one. We built 3 different language models corresponding with 3 types of visual features. The training set used is *night3* set. The model Mc and Me correspond to the color histogram and the edge histograms generated from a 5×5 grid. The model Ms corresponds to the SIFT color feature extracted from interest points. The recognition rates according to several validation sets are presented in Table 1.

Table 1. Results obtained with 3 visual language models (Mc, Me, Ms)

Train	Validation	HSV(Mc)	Edge(Me)	SIFT color(Ms)
night3	night2	**84.24%**	59.45%	79.20%
night3	cloudy2	39.33%	58.62%	**60.60%**
night3	sunny2	29.04%	52.37%	**54.78%**

We noticed that, in the same condition (e.g. night-night), the HSV color histogram Mc outperforms the two other models. However, in different conditions, the result of this model dropped significantly (from 84% to 29%). On the other hand, the edge model (Me) and the SIFT color model (Ms) are more robust to the change of conditions. In the worst condition (night-sunny), it still obtains a recognition rate of 52% for Me and 55% for Ms. As the result, we choose to consider only the edge histogram and SIFT feature for the official runs. Then,

we studied the impact of the post-processing on the ranked list of the models Me and Ms on the recognition rate in Table 2.

Table 2. Result of the post-processing step based on 2 models Me and Ms

Train	Validation	Fusion	Regrouping	Filtering	Smoothing
night3	sunny2	62%	67% (n=15)	72% (β=0.003)	92%(k=20)

The fusion of the 2 models leads to an overall 8% of improvement. The regrouping step helped to pop-up some prominent rooms from the score list by averaging room's n-best scores. The filtering, using the threshold β=0.003, eliminated some of the uncertain decisions. Eventually, the smoothing step with a window size of 40 helped to increase the performance of a sequence of images significantly, by more than 20% compared to the initial result.

2.6 Submitted runs and results

For the official test, we have constructed 3 models based on the validating process. We eliminated the HSV histogram model because of its poor performance on different lighting conditions and there was a little chance to have the same condition. We used the same visual vocabulary of 500 visual concepts generated for night3 set. Each model provided a ranked result corresponding with the test sequence released. The post-processing steps were performed similarly to the validating process employing the same parameters. The visual language models built for the competition are: **Me1**: visual language model based on edge histogram extracted from 10x10 patches division; **Me2**: visual language model based on edge histogram extracted from 5x5 patches division, and **Ms**: visual language model based on color SIFT local features. Our test has been performed on a quad core 2.00GHz computer with 8Gb of memory. The training took about 3 hours on a whole night3 set. Classification of the test sequence was executed in real time. Based on the 3 visual models constructed, we have submitted 4 valid runs to the ImageCLEF evaluation (our runs with smoothing windows were not valid).

- **01-LIG-Me1Me2Ms**: linear fusion of the results coming from 3 models (score = 328). We consider this run as our baseline;
- **02-LIG-Me1Me2Ms-Rk15**: re-ranking the result of 01-LIG-Me1Me2Ms with the regrouping of top 15 scores for each room (score = 415);
- **03-LIG-Me1Me2Ms-Rk15-Fil003**: if the result of the 1st and the 4th in the ranked list is too small (i.e. $\beta = 0.003$), we remove that image from the result list (score = 456.5);
- **05-LIG-Me1Ms-Rk15**: same as 02-LIG-Me1Me2Ms-Rk15 but with the fusion of 2 types of image representation. (score = 25);

These result show that the grouping increases results by 27% compared to the baseline. Adding a filtering after the grouping increases again the results,

gaining more that 39% compared to the baseline. The use of SIFT features is also validated: the result obtained by the run **05-LIG-Me1Ms-Rk15** in not good, even after grouping the results by room. Our best run 03-LIG-Me1Me2Ms-Rk15-Fil003 for the obligatory track is ranked at 12^{th} place among 21 runs submitted in overall. We conclude from these results that the use of post-processing is a must in the context of Robotvision room recognition.

3 Image retrieval and Image annotation tasks results

This paper focuses on the robovision task, but the MRIM-LIG group also submitted results for the image annotation and the image retrieval tasks. For the image annotation task, we tested a simple late fusion (selection of the best) based on three different sets of features: RGB colors, SIFT features, and an early fusion of hsv color space and Gabor filters energy. We tested two learning frameworks using SVM classifiers: a simple one against all, and a multiple one against all inspired from the work of Tahir, Kittler, Mikolajczyk and Yan called Inverse Random Under Sampling [14]. As a post processing, we applied on all our different runs a linear scaling in a way to fit the learning set a priori probabilities. We took afterward into account the hierarchy of concept in the following way: a) when conflicts occur (for instance the tag Day and the tag Night are associated to one image of the test set), we keep unchanged the larger value tag, and we decrease (linearly) the value all the other conflicting tags, b) we propagated the concepts values in a bottom-up way if the values of the generic concept is increased, otherwise we do not update the pre-existing values. The best result that we obtained was 0.384 for equal error rate (rank 34 on 74 runs) and 0.591 for recognition rate (rank 45 on 74). These results need to be studied further. For the image retrieval task, we focused on a way to generate subqueries, corresponding to potential clusters for the diversity process. We extracted the ten most cooccurring words with the query words, and used these words in conjunction with the initial query to generate sub-queries. One interesting result obtained comes from the fact that, for a text+image run, the result we obtained for the 25 last queries (the one for which we had to generate sub queries) was ranked 6th. This result encourages us to further study the behavior of our proposal.

4 Conclusion

To summarize our work on the Robotvision task, we have presented a novel approach for localization of a mobile robot using visual language modeling. Theoretically, this model fits within the standard language modeling approach which is well developed for IR. On the other hand, this model helps to capture in the same time the generality of the visual concepts associated with the regions from a single image or sequence of images. The validation process has proved a good recognition rate of our system against different illumination conditions. We believe that a good extension of this model is possible in the real scenario of scene

recognition (more precisely for robot self-localization). With the addition of more visual features and the increase of system robustness, this could be a suitable approach for the future recognition systems. For the two other tasks in which we participated, we achieved average results. For the image retrieval we will study in the future more specifically the diversity algorithm.

Acknowledgment

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

References

1. B. Caputo, A. Pronobis, and P. Jensfelt. Overview of the clef 2009 robot vision track. In *In CLEF working notes*, 2009.
2. R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *Conference on Computer Vision and Pattern Recognition*, 2005.
3. J. Gao, J.-Y. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *In ACM SIGIR '04*, pages 170–177, 2004.
4. P. Gosselin, M. Cord, and S. Philipp-Foliguet. Kernels on bags of fuzzy regions for fast object retrieval. In *International conference on Image Processing*, 2007.
5. David G. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, pages 91–110, 2004.
6. J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt. Incremental learning for place recognition in dynamic environments. In *Proc IROs 2007*, 2007.
7. L. Maisonnasse, E. Gaussier, and J.P. Chevalet. Model fusion in conceptual language modeling. In *In ECIR09*, pages 240–251, 2009.
8. L. Maisonnasse, E. Gaussier, and J. Chevallet. Revisiting the dependence language model for information retrieval. In *poster SIGIR 2007*, 2007.
9. L. Maisonnasse, E. Gaussier, and J. Chevallet. Multiplying concept sources for graph modeling. In *Lecture Notes in Computer Science, to be published*, 2008.
10. T. T. Pham, L. Maisonnasse, P. Mulhem, and E. Gaussier. Visual language model for scene recognition. In *In Proceedings of SinFra'2009, Singapore*, 2009.
11. J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *In ACM SIGIR '98*, pages 275–281, 1998.
12. F. Song and W. B. Croft. general language model for information retrieval. In *CIKM'99*, pages 316–321, 1999.
13. M. Srikanth and R. Srikanth. Biterm language models for document retrieval. In *Research and Development in Information Retrieval*, pages 425–426, 2002.
14. Muhammad Atif Tahir, Josef Kittler, Krystian Mikolajczyk, and Fei Yan. A multiple expert approach to the class imbalance problem using inverse random under sampling. In *Multiple Classifier Systems*, pages 82–91, Reykjavik, Iceland, 2009.
15. Chee Sun Won, Dong Kwon Park, and Soo-Jun Park. Efficient use of mpeg-7 edge histogram descriptor. In *ETRI Journal*, pages vol.24, no.1, 2002.