Visual Language Modeling for Mobile Localization

LIG participation at RobotVision'09

Trong-Ton Pham[1], Loïc Maisonnasse[2], Philippe Mulhem[1]
[1]Laboratoire Informatique de Grenoble (LIG)
[2]Laboratoire d'InfoRmatique en Image et Systemes d'information (LIRIS)
ttpham@imag.fr, loic.maisonnasse@insa-lyon.fr, mulhem@imag.fr

Abstract

This working note presents our novel approach for scene recognition (i.e. localization of mobile robot using visual information) in the RobotVision task [1] based on language model [2]. Language model has been successfully used for information retrieval (specifically for textual retrieval). In recent study [3], this model has also showed a good performance on modeling the visual information. For this reason, it can be used to address several problems in image understanding such as: scene recognition, image retrieval, etc. We have developed a visual language framework to participate in RobotVision'09 task this year. This framework consists of 3 principal components: a training step, a matching step and a post-processing step. Finally, we present the results of our approach on both validation set and test set released by the ImageCLEF's organizer.

Categories and Subject Descriptors

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

General Terms

Algorithms, Theory

Keywords

Information retrieval, visual language model, late fusion

1 Introduction

This year is the first year of RobotVision track [1] and of LIG participation in this track. The main task is to exploit the location information within a known environment of a mobile robot based on the visual information. The difficulty of this task is that the robot has to recognize the room in different illumination conditions and adapt as the environment changes (such as moving people or objects, new furniture added over the time, etc.). This might pose a problem for a visual recognition system as the trained data usually obtained at a fixed time. In the meanwhile, the system has to provide the location of the robot in real-time and in different time spans (from 6 months to 20 months).

Over the years, several classical approaches in computer vision have been proposed for this problem. In [4], the authors suggested an appearance-based method using Support Vector Machine (SVM) to cope with illumination and pose changes. This method achieved a satisfactory performance when considering a short time interval between training and testing phrases. Another possible approach is to detect the interest point (such as SIFT, Harris-Laplace, etc.) and do a topological matching of these points [5]. This is simple approach but quite effective for recognizing some type of non rigid object (for example: building, car, motorbike, etc.). However, this method is heavily based on the quality of the interest points detected.

To participate in this competition, we reuse our visual language approach presented in [3] with the enhancement to cope with specific conditions of this task. Our model has showed a good robustness and adaptability with different kind of image representations as well as different type of visual features. With the graph representation, we have presented another layer of image understanding that is closer to the semantic layer. Moreover, graph model is integrated well with the foundation of standard language model [2] as showed in [6]. The prominent class for each image is computed based on their likelihood value. We also employed the Kullback-Leibler divergence as proposed in the classical language model approaches. In order to enhance the classification quality, we performed some post processing of the ranked result based on their relevance values. The validating process has shown a good performance of our system on different weather conditions over the time span of 6 months.

In the next section, this paper is organized as follows: Section 2 presents our visual language approach for modeling a scene. Section 3 describes the validating process based on the training and validation set. Section 4 reports our submitted run to the ImageCLEF for evaluation. The paper then concludes with the discussion and future direction of our work.

2 Our approach

We have applied our visual language modeling framework for the competition this year. Our model has been well demonstrated to work with scene recognition as it takes the advantage of a robust platform from the standard language model in IR fields.

2.1 Image modeling

2.1.1 Image representation

We used 2 types of image representation in order to capture different visual information in image content

- **Regular patch**: images are divided into regular size patches. In order to make image representation robust with changing of the camera zoom, we have applied a multi-partition of images into 5x5 patches and 10x10 patches.

- **Interest point**: invariant keypoints are detected using Lowe's interest point detector. These keypoints are invariant with affine transformation and illumination. Local features are then extracted for each keypoint.

2.1.2 Feature extraction

From the validation set, we have learned that the color information performs quite badly in the case of changing illumination. In the same lighting condition, the color histogram could give some good results. However, in the case with a brutal changing of light condition (such as training in night condition and testing with the sunny condition) the system fails to make a satisfied judgment. So we decided to use only some features that are less sensitive with the illumination to represent the visual feature. We have extracted the following features in our experiment:

- **HSV color histogram**: we extract the color information from HSV color space. Each patch is represented by a vector of 512 dimensions.

- **Multi-scale canny edge histogram**: we used canny operator to detect the contour of objects as presented in [7]. An 80-dimensional vector was used to capture magnitudes and gradient of the contours for each patch. We have captured this information in 2 different scales of image (10x10 patches and 5x5 patches).
- **Color SIFT**: SIFT features are extracted using D. Lowe's detector [8]. Region around the keypoint is described by a 128-dimensional vector for each R, G, B channel.

2.1.3 Visual vocabulary construction

Based on the analogy of image and text (i.e. visual word - word), for each feature, we construct a visual vocabulary of 500 visual words using k-means clustering algorithm. Each visual word will be designated to a concept c. Each image will then be represented using theses concepts and we used them to build our language model in the next step.

2.2 Visual language modeling

In [3], 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 we used a reduced version of this model. We do 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 define 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.

2.2.1 Language model

We rely on a language model defined over concepts, as proposed in [6], 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.

Contrary to [6] that compute a query likelihood, we compute the relevance status value rsv of a document image d for query q by using Kullback-Leiber divergence between the document model M_d computed over the document image d and the query model M_q computed over the query image q. By relying on the concept independence hypothesis, this leads to:

$$\begin{aligned} RSV_{kld}(q,d) &= -\mathcal{D}\left(M_q \| M_d\right) && (1)\\ &= \sum_{c_i \in C} P(c_i|M_q) \log\left(\frac{P(c_i|M_q)}{P(c_i|M_d)}\right) && (2)\\ &= \sum_{c_i \in C} \log(P(c_i|M_q) * P(c_i|M_d)) - \sum_{c_i \in C} \log(P(c_i|M_q) * P(c_i|M_q)) && (3) \end{aligned}$$

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. Since the last element of the decomposition correspond to query entropy and does not affect documents ranking, we only compute the following decomposition:

$$RSV_{kld}(q,d) \propto \sum_{c_i \in C} \log(P(c_i|M_q) * P(c_i|M_d)) \qquad (4)$$

where the quantity $P(c_i|M_d)$ is estimated through maximum likelihood (as is standard in the language modeling approach to IR), using Jelinek-Mercer smoothing:

$$P(c_i|M_d) = (1 - \lambda_u)\frac{F_d(c_i)}{F_d} + \lambda_u \frac{F_c(c_i)}{F_c} \tag{5}$$

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 corresponds to the Jelinek-Mercer smoothing. It plays the role of an IDF parameter, and 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:

$$P(c_i|M_q) = \frac{F_q(c_i)}{F_q} \tag{6}$$

where $F_q(c)$, representing the sum of the weight of c in all graphs from query image q and F_q the sum of all the query concept weights in q. The final result of each query image is a ranked list of documents associated with their rsv value.

2.2.2 Querying

Using this model, we query the training set with each test image using one type of concepts (i.e. concepts obtains with one feature). Thus for each test image we obtain a list, standard in IR, that contains all the training set images ranked according to the rsv defined in the previous part. This list can be represented as:

$$IL_q = [(d, rsv(q, d))] \tag{7}$$

Where IL_q is a ranked list of image for query q, d is one image of the training set and $rsv(q, d)$ is the rsv computed for this query and document images.

Assuming a function that, for each training image given its room id, we can obtain the room id of any image from the ranked list. Then, in our basic approach, we associate the query image with the room id of the best ranked image. As we can represent one image with different features and as we have more than one images of each room in the training, we will present in the following a post-processing steps to take advantage of these considerations.

2.3 Post-processing of the results

We perform some fine-tuning steps of this results in order to enhance the accuracy of our system as presented in Figure 1.

- **Linear fusion**: we take the advantage of the different features extracted from the images. We represent an image by a set of concept sets C_i, each C_i corresponding to a visual feature. Assuming that all the concepts sets are independent one to another, we fuse the Kullback-Leiber divergence of individual sets of concepts using a sum:

$$RSV(Q, D) = \sum_i RSV_{kld}(q_i, d_i) \tag{8}$$

  where $Q = q_i$ and $D = d_i$ are the set of concept sets corresponding to the query image and to the document image respectively.

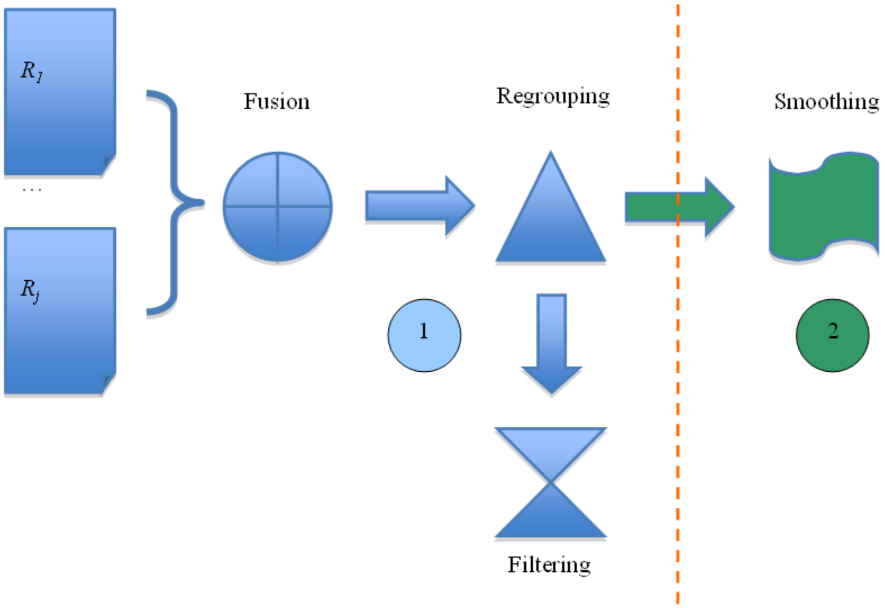

Figure 1: Post-processing step of results. (1) is the scheme for the obligatory track and (2) is the scheme for the optional track

- **Regrouping training image by their room**: On the basis that using only the closest training image to determine the room of a query image is not enough, we proposed to group the results of the n-best images for each room. We compute a ranked list of room RL instead of an image list:

$$RL_q = [(r, RSV_r(q,r)] \tag{9}$$

with

$$RSV_r(q,r) = \sum_{f_{n-best}(q,r)} RSV(q,d) \tag{10}$$

where r correspond 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**: we measured a difference from the score of the 4th room to the 1st room in the room list RL. If the difference is big enough (> threshold β) we keep this image. Otherwise we remove it from the list (or consider as an *unknown* room). In our experiment, we fixed the value $\beta = 0.003$.
- **Smoothing window**: we exploited the continuity in a sequence of images by smoothing the result in the temporal direction. To do that, we use a smoothing window sliding on the classified image sequences. Here, we choose the width of window $w = 40$ (i.e. 20 images before and after the classified image). As the result, the score of the smoothed image is the mean value of their neighborhood images.

$$RSV_{window}(Q_i, R) = \frac{\sum_{j\in[j-w/2;j+w/2]} RSV(Q_j, R)}{w} \tag{11}$$

where w is the width of the smoothing window. In the real case, we could only use a semi-window which considers only the images before the current classified image. This leads to:

$$RSV_{semi-window}(Q_i, R) = \frac{\sum_{j \in [j-w;j]} RSV(Q_j, R)}{w} \tag{12}$$

where w is the width of the semi-window.

3 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. Moreover, the validation process can help us to fine-tune the model parameters that the latter will be used for the official test.

We built 3 different language models corresponding with 3 types of visual features. The training set used is night3 set. Model Mc and Me correspond with the color histogram and the edge histogram extracted from image with the division of 5x5 patches. Model Ms corresponds with the SIFT color feature extracted from interest points. We measure the precision of system using the accuracy rate. Summary of the results is reported in Table 1.

Table 1: Results obtained on different conditions with 3 visual language models (Mc, Me, Ms)

Train	Test	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 outperformed all the other models. However, in different conditions, the result of this model dropped significantly (from 84% to 29%). It showed that the color information is very sensitive with the changing of illumination condition. On the other hand, the edge model (Me) and the SIFT color model (Ms) are practically robust with the changing of condition. In the worst condition (night-sunny), we still obtained a quite good recognition rate of 52% for Me and 55% for Ms. As the result, edge histogram and SIFT feature are shosen as the appropriate features for our recognition system.

Follow is the results of the post-processing step based on the ranked list of Me and Ms (Table 2).

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

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

The fusion of these 2 models gives overall 8% of improvement. The regrouping step (as expected) helped to pop-up some prominent rooms from the score list by averaging room's n-best scores. The filtering takes part in eliminating some of the uncertain decisions base on the difference of their score after the regrouping step. Finally, the smoothing step (which is an optional step) helps to increase the performance of a sequence of images significantly by 20% more.

4 Description of submitted runs

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 similar to the validating process employing the same parameters. Follows are the visual language models built for the competition:

- **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
- **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 executed in real time.

Based on the 3 visual models constructed, we have submitted 6 runs to the ImageCLEF evaluation.

- **01-LIG-Me1Me2Ms**: linear fusion of the results coming from 3 models (Score = 328)
- **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 image that from the list. We refrain the decision from some cases other than to mark them as *unknown* room (Score = 456.5)
- **04-LIG-Me1Me2Ms-Rk2-Diff20**: re-ranking the result of 01-LIG-Me1Me2Ms with the regrouping of top 2 scores for each room and using smoothing window (±20 images/frame) to update the room-id from image sequences (Score = 706)
- **05-LIG-Me1Ms-Rk15**: same as 02-LIG-Me1Me2Ms-Rk15 but with the fusion of 2 types of image representation. (Score = 25)
- **06-LIG-Me1Ms-Rk2-Diff20**: same as 04-LIG-Me1Me2Ms-Rk2-Diff20 but with the fusion of 2 model Me1 and Ms (Score = 697)

Note: run 04-LIG-Me1Me2Ms-Rk2-Diff20 and run 06-LIG-Me1Ms-Rk2-Diff20 are invalid as we used the images after the classified image for the smoothing window.

Our best run 03-LIG-Me1Me2Ms-Rk15-Fil003 for the obligatory track is ranked at 12^{th} place among 21 runs submitted in overall. Although, run 04-LIG-Me1Me2Ms-Rk2-Diff20 had not met the criteria of the optional task which only used the sequence before the classified image. Nervertheless, it has increased by roughly 250 points from the best obligatory run. It means that we still have room to improve the performance of our systems with the valid smoothing window.

5 Conclusion

In this paper, we have presented a novel approach for localization of a mobile robot using visual language modeling. Theorically, 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, enhancement of system robustness and choosing the right parameter, this could be the solution to the future recognition system.

Acknowledgment

This work was partly supported by the French National Agency of Research (ANR-06-MDCA-002) and by the Rgion Rhones Alpes (projet LIMA).

References

[1] B. Caputo, A. Pronobis, and P. Jensfelt. Overview of the clef 2009 robot vision track. In *CLEF working notes 2009, Corfu, Greece*, 2009.

[2] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Research and Development in Information Retrieval*, 1998.

[3] T. T. Pham, L. Maisonnasse, P. Mulhem, and E. Gaussier. Visual language model for scene recognition. In *In Proceedings of SinFra'2009, Singapore*, 2009.

[4] A. Pronobis, O. Martnez Mozos, and B. Caputo. Svm-based discriminative accumulation scheme for place recognition. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA08)*, 2008.

[5] David G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, 1999.

[6] L. Maisonnasse, E. Gaussier, and J.P. Chevalet. Model fusion in conceptual language modeling. In *In 31st European Conference on Information Retrieval (ECIR09)*, pages 240–251, 2009.

[7] 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.

[8] David G. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, pages 91–110, 2004.