Integration of Spatial Relationships in Visual Language Model for Scene Retrieval

Trong-Ton Pham, Philippe Mulhem, Loïc Maisonnasse, Eric Gaussier, Ali Aït-Bachir
Computer Science Laboratory Grenoble - LIG
385 Av. de la bibliothèque, 38041 Grenoble Cedex
firstname.lastname@imag.fr

Abstract

In this paper, we describe a method to use a graph-based language modeling approach for image retrieval and image categorization. We first mapped image regions to induced concepts and then spatial relationships between these regions to build a graph representation of images. Our method allows to deal with different scenarii, where isolated images or groups of images are used for training and testing. The results obtained on an image categorization problem comprising of 3849 images from 101 landmarks of Singapore show that (a) the procedure to automatically induce concepts from an image is effective, and (b) the use of spatial relationships, in addition to concepts, for representing an image content helps improve the classifier accuracy. This approach is the first one, to our knowledge, to present a complete extension of the language modeling approach from information retrieval to the problem of graph-based image categorization and retrieval.

1 Introduction

After almost 20 years of research in image retrieval and categorization, the domain is still considered as a great challenge for computer scientists. Problems arising with image indexing, image retrieval and image categorization are related to the *semantic gap*, and the way one can represent an image content. Besides this inherent difficulty, another dimension of interest is the fact that images are usually related to other images. For instance, all the digital cameras do now integrate the date and time of shooting and this information may be used to group them [9]. In addition, geo-localization information may be used [3] also. Therefore, groups of images may also be an interest approach for image indexing and retrieval. We show that the use of language modeling may extend smoothly to multiple image queries and multiple trained images, and that such an approach is robust with respect to the differences between these groups of images.

Several works have considered the use of spatial relationships between image regions. For instance, image descriptions expressed by 2D strings as in the Visualseek system [11] capture sequences of occurrences of objects along one or several reading directions. However, a retrieval based on 2D strings is complex, as it requires matching substrings, which is a costly operation. To overcome this problem, several heuristics to speed up the process have been proposed, as the one described in [1] which results in a process of 10 times faster than the original one.

Other works have considered relationships between image regions in a probabilistic model, e.g. through the use of 2D HMMs, as in [4]. However, these works focus on image annotation, and do not consider relations at retrieval time. Other attempts have concentrated on conceptual graphs [7] for image indexing and retrieval. However, taking into account explicit relationships may generate complex graphs representations, and the retrieval process is likely to suffer from the complexity of the graph matching process [8]. One of our concern here is to be able to represent images using graphs, without suffering from the burden of computationally expensive matching procedures. One solution is then to take benefits from existing approaches in the field of information retrieval.

The language modeling approach in information retrieval exists from the end of the 90s [10]. 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], and, more recently, with relationships between terms and graphs. Thus, Gao et al. [2] 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. Maisonnasse [6] further extend this approach with a compatible model for general graphs, as the ones obtained by a conceptual anal-

ysis of documents and queries. We rely here on this latter models, however extending it by (a) applying it to an image collections, and (b) considering that both concepts and relations can be weighted.

The remainder of the paper is organized as follows: section 2 presents the visual language model used to describe image content, as well as the matching procedure used to compute the similarity between images; section 3 then presents and discusses the results obtained with our approach in categorizing of 101 classes; we finally conclude in section 4.

2 Language Modeling for Image Correspondence

2.1 Image modeling with visual graphs

Our goal here is to automatically induce, from a given image, a graph that represents the image content. Such a graph will contain concepts directly associated with the elements present in the image, as well as relations which express how concepts are related in the image. To do so, our procedure is based on four main steps:

1. Identify regions within the image that will form the basic blocks for concept identification.
2. Index each region with a predefined set of features.
3. Cluster all the regions found in the collection in K classes, each class representing one concept. At the end of this step, each region in the image is represented as a concept, namely the class to which the region belongs. The set of concepts, $\mathcal{C}$, thus corresponds to the set of classes obtained.
4. Finally, extract relations between concepts.

The first step, region identification, can be based on an arbitrary division of equal size, non-overlapping regions (such as dividing the image into 5×5 blocks) or on regions defined around interest points (such as SIFT points[1]). The second step aims at representing each region as set of vector for clustering purposes. The features we have retained in this paper are HSV color features, which can be easily and efficiently extracted. We rely on *k-means* algorithm for the third step, as it is a standard in image retrieval, well understood clustering procedure, but other methods can be used as well. Lastly, the fourth step yields a set of concepts related through certain relations. In this work, we focus on spatial relations, which are characterized by *top_of* and *left_of* (see figure 1). At the end of this procedure, one gets a set of related concepts to represent each image. Furthermore, as a concept may appear several times in an image (when different regions of the image are assigned to the same cluster, as is typical for regions describing for example a *sky* or a *sea*), each concept is associated with a weight that represents its number of occurrences in the image. Similarly, each relation is given a weight corresponding to the number of times the relation was observed between the two concepts in the image.

[1] in such a case, regions may overlap.

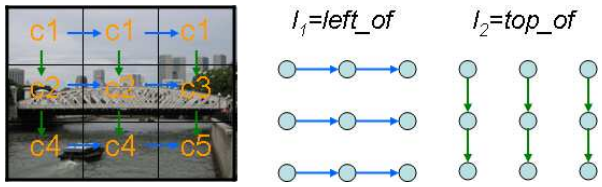

Figure 1. Example of spatial relationships extracted from one image.

In the remainder, we will denote set of weighted concepts used to represent a given image by W_C. W_C is defined on $(C \times \mathbb{N})$. Each association between any two concepts c and c' is directed and is represented by a triplet $< L(c,c'), l, n(c,c',l) >$, where $L(c,c')$ represents the fact that there exists, in the image, a spatial relation between the two concepts (in which case $L(c,c') = 1$) or not (in which case $L(c,c') = 0$), l represents the label of the association between the two concepts, and $n(c,c',l)$ represents the number of times the two concepts are connected in the image with label l. We consider here that the possible labels expressing a spatial relation between two concepts are either *top_of* or *left_of*. The converse relations are implicitly captured as the associations we consider are directed. In the case where $L(c,c') = 0$, i.e. there is no spatial relation between the concepts, then $l = \emptyset$ by definition.

Finally, the graph representing an image is defined as $G =< W_C, W_E >$, where W_C represents the set of weighted concepts defined previously, and W_E the set of triplets $< L(c,c'), l, n(c,c',l) >$ defined for each concept pair in W_C. L is an application from $C \times C$ to $\{0,1\}$, l is an element of $\{top\_of, left\_of, \emptyset\}$ and $n(c,c',l)$ is the number of times the particular association holds in the image.

2.2 A language model for graph matching

Our matching function is based on the standard language modeling approach [10], extended so as to take into account the elements defined above. To differentiate the images taken into account during the matching process, we will refer to one of the images (or potentially to one set of images) as the *document*, and to the other one as the *query*.

The probability for a query graph $G_q =< W_C^q, W_E^q >$ to be generated by the document model M_d is then defined by:

$$P(G_q|M_d) \quad = \quad P(W_C^q|M_d) \times P(W_E^q|W_C^q, M_d)$$

For the probability of generating query concepts from the document model ($P(W_C^q|M_d)$), we rely on the concept independence hypothesis, standard in information retrieval and categorization. The number of occurrences of the concepts (i.e. the weights considered previously) are naturally integrated through the use of a multinomial model, leading to:

$$P(W_C^q|M_d) \quad \propto \quad \prod_{c \in \mathcal{C}} P(c|M_d)^{n(c,q)}$$

where $n(c,q)$ denotes the number of times concept c occurs in the graph representation of the query. The quantity $P(c|M_d)$ is estimated through maximum likelihood (as is standard in the language modeling approach to IR), using Jelinek-Mercer smoothing:

$$P(c|M_d) \quad = \quad (1-\lambda_c)\frac{F_d(c)}{F_d(.)} + \lambda_c \frac{F_{\mathcal{D}}(c)}{F_{\mathcal{D}}(.)}$$

with $F_d(c)$ representing the weight of c in the graph representation of the *document*, and $F_d(.)$ being equal to $\sum_c F_d(c)$. The functions $F_{\mathcal{D}}$ are similar, but defined over the whole collection (i.e. over the union of all the graphs from all the documents of the collection). The parameter λ_c corresponds to the Jelinek-Mercer smoothing. It plays the role of an IDF[2] parameter, and helps taking into account reliable information when the information from a given document is scarce. We follow a similar process for the associations, leading to:

$$P(W_E^q|M_d) \propto \prod_{(c,c') \in \mathcal{C}^2} P(L(c,c'), l|W_C^q, M_d)^{n(c,c',l,q)}$$

The quantity $P(L(c,c') = x, l|W_C^q, M_d)$ can be decomposed as the probability of generating a particular type of association ($x = 0$ or 1) and then as the probability of using a particular label l to annotate the association. This amounts to:

$$P(L(c,c') = x, l|W_C^q, M_d) =$$
$$P(L(c,c') = x|W_C^q, M_d) \times P(l|L(c,c') = x, M_d)$$

The two quantities appearing in the right-hand side of the above equation are then directly estimated through maximum likelihood. For the first quantity, we have:

$$P(L(c,c') = x|W_C^q, M_d) =$$
$$(1-\lambda_r)\frac{xF_d(c,c',R) + (1-x)F_d(c,c',\neg R)}{F_d(c,c',R) + F_d(c,c',\neg R)} \quad + \quad$$
$$\lambda_r \frac{xF_{\mathcal{D}}(c,c',R) + (1-x)F_{\mathcal{D}}(c,c',\neg R)}{F_{\mathcal{D}}(c,c',R) + F_{\mathcal{D}}(c,c',\neg R)}$$

[2]Inverse Document Frequency

where $F_d(c,c',R)$ represents the number of times c and c' are related through a spatial relation in the *document*, whereas $F_d(c,c',\neg R)$ represents the number of times they are not related through a spatial relation. Again, a smoothing is used based on the whole collection (and the associated functions $F_{\mathcal{D}}$). The probability for a particular label is estimated in the same way:

$$P(l|L(c,c') = 1, M_d) =$$
$$(1-\lambda_l)\frac{F_d(c,c',l,R)}{F_d(c,c',.,R)} + \lambda_l \frac{F_{\mathcal{D}}(c,c',l,R)}{F_{\mathcal{D}}(c,c',.,R)}$$

For $x = 0$, the only possible label is $\emptyset$, so that the probability of this label given $x = 0$ is 1. The definitions of the functions F_d and $F_{\mathcal{D}}$ in the above equations are similar to the ones seen previously, but concern labels.

The model we have just presented is inspired by the model defined in [6]. It differs however from it in (a) we propose in this paper a complete methodology for automatically indexing images at a conceptual level, and (b) it takes into account weights on each concept and association. As the weights are integers, we relied on multinomial distributions for the underlying generative process. The consideration of real-valued weights would lead to consider continuous distributions instead of the multinomial one. We are now going to illustrate the behavior of our model in the context of image categorization.

3 Experiments

We want to illustrate here the validity of our approach within an image classification task. In particular, we want to assess (a) the well-foundedness of the conceptual indexing method we have retained, as well as (b) the usefulness of spatial relationships for a better characterization of image content. We also show that our overall methodology is robust with respect to the changes in the usage scenarii.

3.1 The STOIC-101 collection

The Singapore Tourist Object Identification Collection is a collection of 3849 images containing 101 popular tourist landmarks (mainly outdoor). These images were taken, mainly from a consumer digital cameras in a manner typical of a casual tourist, from 3 distances and 4 angles in natural light, with a mix of occlusions and cluttered background to ensure a minimum of 16 images per scene. Images in the collection are affected by different weather patterns and different image capturing styles. For experimental purposes, the STOIC-101 collection has been divided into a training set containing 3189 images (82.8% of the collection) and a test set containing 660 images (17.15% of the colelction). The average number of images per class for training is 31.7,

Figure 2. Interface of searching engine with an image query of Merlion statue.

and 6.53 for testing. In the test set, the minimum number of images per class is 1, and the maximum 21.

The main application of STOIC collection is a web-based image search engine. An user can upload an image and post it as query to the system. On the server side, the images from the 101 scenes of the STOIC collection are matched against the user query. The search engine architecture is two folds: a) the query process server takes a query image as input and generates a query graph file b) the language model server receives the query graph and computes the matching function based on trained graphs. The results are sent back to the query process server and an user can visualize these result images. The mean execution time for one query image is about 2 seconds. However, the engine can be optimized and be removed some intermediate steps to accelerate the processing time.

As an user can take one or several images of the same scene and query the system accordingly, we have considered several usage scenarii. Table 1 summarizes these different scenarii (a scene (S) corresponds to a group of images and a single image (I)). Note that some images in the collection have been rotated into the correct orientation (for both portrait and landscape layouts).

	Training by (I)	Training by (S)
Query by (I)	$\surd$	$\surd$
Query by (S)	$\surd$	$\surd$

Table 1. Summary of experiments on STOIC-101 collection

3.2 Indexing images with concepts and spatial relationships

Several studies on the STOIC collection have shown that color plays a dominant role, and should be preferred over other visual features as edge features or texture [5]. Furthermore, color features can be easily and efficiently extracted. For these reasons, we rely, for our methods, on HSV color features only.

In order to assess the validity of our methodology, we followed different ways to (a) divide each image into regions and (b) assign each region with a concept. For the division of images into regions, we retained:

1. A fine-grained division where a region corresponds to one pixel. We refer to this division as *fg*.
2. A division of medium grain, where blocks of 10x10 pixels are used, the center pixel being considered as a representative for the region. We refer to this division as *mg*.
3. A gross division where the image is divided into 5x5 regions of equal size. We refer to this division as *gg*.

For the *fg* and *mg* divisions, we first respectively quantized each RGB (red, green, blue) and HSV (hue, saturation, value) channel values into 8 bins of equal size (from 0 to 64). This yielded a 512 (8x8x8) dimensional binary vector for each region. Each dimension corresponds to a concept (defined according to the bins) whereas the coordinate on each dimension corresponds to the presence (1) or absence (0) of the concept in the region. The global image is then indexed by the sum of all region vectors. We will refer to the indexing thus obtained as *fg-PreCon* and *mg-PreCon*, resp. for "division *fg* with predefined concepts" and "division *mg* with predefined concepts". The rationale for doing so is to assess the validity of the clustering method we proposed in section 2 for identifying concepts in the collection. In the above setting, concepts are arbitrarily defined through bins, whereas in the following ones, they are identified through unsupervised clustering, as described in section 2.

For *mg* (again) and *gg* divisions, we clustered the HSV feature vectors of all regions into $k = 500$ classes with *k-means*. This results in a hard assignment of each region to one class/concept. The set of weighted concepts, W_C, is then obtained by counting how many times a given concept occurs in the image. The choice of $k = 500$ is motivated by the fact that we want a certain granularity in the number of concepts used to represent an image. With too few concepts, one is likely to miss important differences between images, whereas too many concepts will tend to make similar images look different. We will refer to the indexing obtained in this way as *mg-AutCon* and *gg-AutCon*, resp. for "division *mg* with automatically induced concepts" and "division *gg* with automatically induced concepts".

Training	Query	*fg-PredCon*	*mg-PredCon*	*mg-AutoCon*	*mg-AutoCon-Rel*	*gg-AutoCon*	*gg-AutoCon-Rel*
I	I	0.687	0.670	0.789	**0.809**(+2.5%)	0.484	0.551(+13.8%)
I	S	0.653	0.65	0.822	**0.851**(+3.6%)	0.465	0.762(+63.8%)
S	I	0.409	0.402	0.529	0.594(+12.3%)	0.478	**0.603**(+26.1%)
S	S	0.940	0.940	**1.00**	**1.00**	0.891	0.920(+3.2%)

Table 2. Impact of spatial relations on the performance (best results are in bold, relative improvement over the method without relations is in parentheses)

In addition, for the methods *mg-AutoCon* and *gg-AutoCon*, we extracted the spatial relations between concepts mentioned previously: *left_of* and *top_of*, and counted how many times two given concepts are related through a particular relation in order to get weights for our relations. This last step provides a complete graph representation for images. We will refer to these two complete methods as *mg-AutoCon-Rel* and *gg-AutoCon-Rel*.

Last but not least, to classify query images in the 101 scenes, we used, for all indexing methods, the language model for visual graphs presented in section 2. This similar to using a 1-NN classifier, with the "similarity" defined by equation 1 (and its development). When there is no relation, the term $P(w_E^q|M_d)$ equals 1 (see equation 2.2) so that only concepts are taken into account to compare images.

3.3 Experimental results

The performance of the different methods was evaluated using the accuracy, per image or per scene. They both are defined as the ratio of correctly classified images or scenes. More precisely:

$$\text{Image accuracy} = \frac{TP_i}{N_i}, \ \text{Scene accuracy} = \frac{TP_s}{N_s}$$

where TP_i, respectively TP_s, represents the number of images (resp. scenes) correctly classified. N_i is the total number of test images (i.e. 660 images), and N_s the total number of scenes (i.e. 101 locations).

Table 2 displays the results we obtained when using predefined (through bins) or automatically induced (through clustering) concepts. As one can see, automatically inducing concepts with a medium grain division of the image yields the best results (the difference with the gross grain division for the S-I scenario being marginal). Another interesting point to note on table 2 is that the gross grain division method does not help generalize, over the medium grain one. In particular, the S-I and I-S scenarii in a way correspond to degenerate usages of the system, as the training and testing material are not the same. It is beneficial, in such cases, to abstract away from a strict description of images so as to be able to generalize to "new" test data. The evolution of the accuracy of the *fg-PredCon* and *gg-AutoCon-Rel* methods illustrates this: The accuracy for the I-S and S-I scenarii is better than the one for the I-I scenario for *gg-AutoCon-Rel*, whereas it is worse for *fg-PredCon* (this latter fact being also true for the *mg-PredCon* method, even though the difference is less marked, as one would expect). The *fg-PredCon* method, relying on an indexing which is very close to the original image, is not able to generalize well to new usages.

This being said, there is a difference between the I-S and S-I scenarii: The system is queried with more information in the I-S scenario than in the S-I scenario. This difference results in a performance which is, for all methods, worse for the I-S scneraio than for the other ones. We conjecture that this is why the results obtained for the *mg-AutoCon-Rel* method on S-I are not as good as the ones for I-I. There seems to be a plateau for this scenario around 0.6, an hypothesis we want to explore in future work.

We finally assessed the usefulness of spatial relationships by comparing the results obtained with the different methods that include or not such relations. These results are displayed in table 2. As one can note, except for the S-S scenario with the *mg* division, the use of spatial relations always improve the accuracy of the classifier. This justifies the framework we developed in section 2 of language model for visual graphs including automatically induced concepts and spatial relations between them.

4 Conclusion

We have introduced in this paper a new model for matching graphs derived from images. The graphs we have captured spatial relations between concepts associated with regions of an image. On a formal side, our model fits within the language modeling approach to information retrieval, and extends previous proposals based on graphs. On a more practical side, the consideration of regions and associated concepts allows to gain generality in the description of images, a generality which may take benefit when the usage of the system slightly differs from its training environment. This is likely to happen with image collections, for example, use one or several images to represent a scene. On the other hand, querying a specific location with a group of images is very promising in future application (such as mobile

localization) that allows higher accuracy rate with less computational effort comparing to video sequence.

The experiments we have conducted aim at assessing the validity of our approach with respect to these elements. In particular, we showed that integrating spatial relations to represent images led to a significant improvement in the results. The model we have proposed is able to adequately match images and sets of images represented by graphs. Furthermore, we showed that the representation based on concepts and relations between them is less sensitive to a change in the usage than one based on pixel-level features. As we conjectured, being able to abstract away from a low level description allows robustness with respect to the usage scenarios. The price to pay for this robustness is that the representation, which can be seen as recall oriented, does not provide as good results as a low level, less general representation when the usage scenario directly parallels the training one. The best strategy to follow is thus to adopt the low level representation when the usage scenario is identical to the training one, and the high level one in the other cases.

In the future, we plan on using the graph model defined here with different divergence measures. The framework we have retained is based on the Kullback-Leibler divergence. However, the Jeffrey divergence has also been used with success on image collections, and could be used to replace the Kullback-Leibler one. We also wish to investigate different possible coupling of the low level and high level representations, with the will to come up with a single representation that could be used in all cases.

Acknowledgement

This work was partially supported by the French National Agency of Research (ANR-06-MDCA-002) and Merlion Ph.D. programme from Singapore.

References

[1] Y. Chang, H. Ann, and W. Yeh. A unique-id-based matrix strategy for efficient iconic indexing of symbolic pictures. *Pattern Recognition*, 33(8):1263–1276, 2000.

[2] J. Gao, J.-Y. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 170–177, 2004.

[3] Kennedy L., Naaman, M. Ahern, S., Nair R., and Rattenbury T. How flickr helps us make sense of the world: context and content in community-contributed media collections. In *Proceedings of the 15th international Conference on Multimedia*, pages 631–640, 2007.

[4] J. Li and James Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1075–1088, 2003.

[5] J. Lim, Y. Li, Y. You, and J. Chevallet. Scene recognition with camera phones for tourist information access. In *ICME 2007, International Conference on Multimedia & Expo*, 2007.

[6] L. Maisonnasse, E. Gaussier, and J. Chevallet. Model fusion in conceptual language modeling. In *31st European Conference on Information Retrieval ECIR*, 2009.

[7] P. Mulhem and E.Debanne. A framework for mixed symbolic-based and feature-based query by example image retrieval. *International Journal for Information Technology*, 12(1):74–98, 2006.

[8] I. Ounis and Marius Pasca. Relief: Combining expressiveness and rapidity into a single system. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 266–274, 1998.

[9] John C. Platt, Mary Czerwinski, and Brent A. Field. Phototoc: Automatic clustering for browsing personal photographs. In *Proc. Fourth IEEE Pacific Rim Conference on Multimedia*, 2003.

[10] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, 1998.

[11] J. R. Smith and Chang S.-F. Visualseek: a fully automated content-based image query system. In *In Proceedings of the Fourth ACM international Conference on Multimedia*, pages 87–98, 1996.

[12] F. Song and W. B. Croft. General language model for information retrieval. In *CIKM'99*, pages 316–321, 1999.