Structurally Enhanced Latent Semantic Analysis for Video Object Retrieval

Fabrice Souvannavong, Lukas Hohl, Bernard Merialdo and Benoît Huet
Département Communications Multimédias
Institut Eurécom
2229, route des crêtes
06904 Sophia-Antipolis - France
(Fabrice.Souvannavong, Bernard.Merialdo, Benoit.Huet)@eurecom.fr

September 23, 2004

Abstract

The work presented in this paper aims at reducing the semantic gap between low level video features and semantic video contents. The proposed method for finding associations between segmented frame region characteristics relies on the strength of Latent Semantic Analysis (LSA). Our previous experiments [1], using color histograms and Gabor features, have rapidly shown the potential of this approach but also uncovered some of its limitation. The use of structural information is necessary, yet rarely employed for such a task. In this paper we address two important issues. The first is to verify that using structural information does indeed improve information retrieval performances, while the second concerns the manner in which this additional information is integrated within the framework. Here, we propose two methods using the structural information contained in object parts topological arrangement. The first adds structural constraints indirectly to the LSA during the preprocessing of the video, while the other includes the structure directly within the LSA. Finally, our retrieval results demonstrate that when the structure is added directly to the LSA the performance gain of combining visual (low level) and structural information is convincing.

Keywords: Latent Semantic Analysis, Region-Based Video Retrieval

1 Introduction

Multimedia digital documents are readily available, either through the Internet, private archives or digital video broadcast. Traditional text based methodologies for annotation and retrieval have shown their limit and need to be enhanced with content based analysis tools. Research aimed at providing such tools have been very active over recent years [2]. Whereas most of these approaches focus on frame or shot retrieval, we propose a framework for effective retrieval of semantic video objects. By video object we mean a semantically meaningful spatio-temporal entity in a video.

Most traditional retrieval methods fail to overcome two well known problems called synonymy and polysemy, as they exist in natural language. Synonymy causes different words describing the same object, whereas polysemy allows a word to refer to more than one object. Latent Semantic Analysis (LSA) provides a way to weaken those two problems [3]. LSA has been primarily used in the field of natural language understanding, but has recently been applied to domains such as source code analysis or computer vision. Latent Semantic Analysis has also provided very promising results in finding the semantic meaning of multimedia documents [1, 4, 5]. LSA is based on a Singular Value Decomposition (SVD) on a word by context matrix, containing the frequencies of occurrence of words in each context. One of the limitations of the LSA is that it does not take into account word order, which means it completely lacks the syntax of words. The analysis of text, using syntactical structure combined with LSA already has been studied [6, 7] and has shown improved results. For our object retrieval task, the LSA is computed over a visual dictionary where region characteristics, either structurally enhanced or not, correspond to words.

The most common representation of visual content in retrieval system relies on global low level features such as color histograms, texture descriptors or feature points, to name only a few [8, 9, 10, 11]. These techniques in their basic form are not suited for object representation as they capture information from the entire image, merging characteristics of both the object and its surrounding, in other word the object description and its surrounding environment become merged. A solution is to segment the

image in regions with homogeneous properties and use a set of low level features of each region as global representation. In such a situation, an object is then referred to as a set of regions within the entire set composing the image. Despite the obvious improvement over the global approach, region based methods still lack important characteristics in order to uniquely define objects. Indeed it is possible to find sets of regions with similar low level features yet depicting very different content. The use of relational constraints, imposed by the region adjacency of the image itself, provides a richer and more discriminative representation of video object. There has only been limited publications employing attributed relational graph to describe and index into large collection of visual data [12, 13, 14, 15] due to the increased computational complexity introduced by such approaches. Here we will show that it is possible to achieve significant performance improvement using structural constraints without increasing computational complexity.

This paper is organized as follows. The concept of adding structure to LSA and a short theoretical background on the algorithms used, are presented in Section 2. Section 3 provides the experimental results looking at several different aspects of the object retrieval task. Then it focuses on a larger scale evaluation on Video-TREC news video sequences. The conclusion and future directions are discussed in Section 4.

2 Enhancing Latent Semantic Analysis with Structural Information

As opposed to text documents there is no predefined dictionary for multimedia data. It is therefore necessary to create one to analyze the content of multimedia documents using the concept of Latent Semantic Analysis [3]. Here, we propose three distinct approaches for the construction of visual dictionaries. In the non-structural approach, each frame region of the video is assigned to a class based on its properties. This class corresponds to a ”visual” word and the set of all classes is our visual dictionary. In the case where we indirectly add structure, the clustering process which builds the different classes (words) takes structural constraints into account. Finally, in the third

case where structure is added directly to the LSA, pairs of adjacent regions classes (as in the non-structural approach) are used to define words of the structural dictionary. We shall now detail the steps leading to three different dictionary constructions.

2.1 Video preprocessing

We consider a video V as a finite set of frames $\{F_1, \ldots, F_n\}$, where the preprocessing is performed on key-frames representing the video content. This implies that the video is segmented into shots where the most representative frame is selected [16, 17]. Key-frames of the video V are segmented in regions R_i using the method proposed by Felzenszwalb and Huttenlocher in [18]. This algorithm was selected for its perceived computation requirement and segmentation quality ratio. Each segmented region R_i is characterized by its attributes, feature vectors that contain visual information about the region such as color, texture, size or spatial information. For this paper, the feature vector is limited to a 64 bin HSV color histogram and 24 Gabor energies of the corresponding region. Other attributes could indeed lead to better results, however for the scope of this paper we are mainly interested in identifying whether structural constraints provide performance improvements. The described method is applied to each feature, then similarity scores are merged to obtain a single value as explained in section (2.4). The methodology presented here may be extended to deal any number of low-level features with very little efforts.

2.2 Building the basic visual dictionary

The structure-less dictionary is constructed by grouping regions with similar feature vectors together as illustrated in figure (1). There are many ways to do so [19]. Here the k-means clustering algorithm [19] is employed with the Euclidean distance as similarity measure. As a result each region R_i is mapped to a cluster C_l (or class), represented by its cluster centroid. Thanks to the k-means clustering parameter k controlling the number of clusters, the dictionary size may be adjusted to our needs. In this case, each cluster represents a word for the LSA.

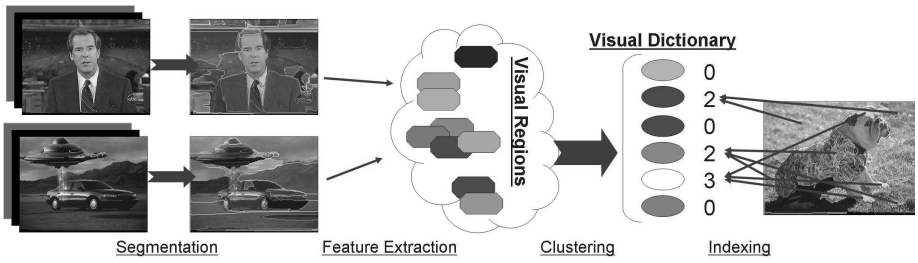


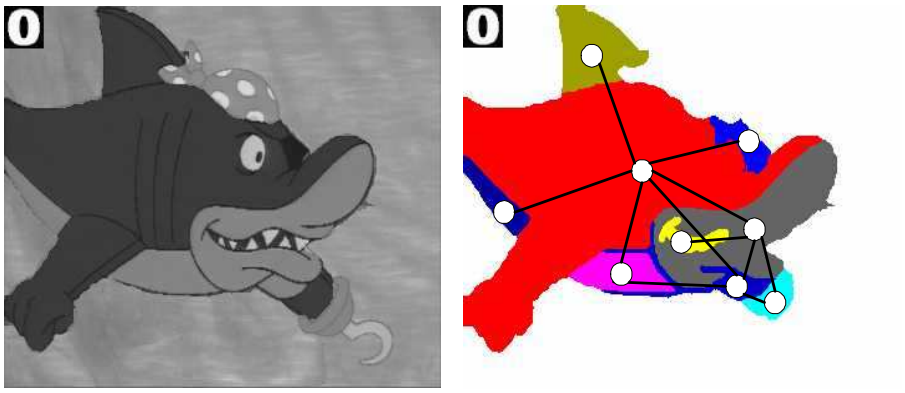
Figure 1: **Building the basic dictionary.**

2.3 Incorporating structural information

In an attempt to increase the influence of local visual information, we propose to include for each region its surrounding content. For this purpose a neighborhood graph is constructed from the segmented regions for each frame. Nodes in the graph represent segmented regions and are attributed with a vector H. Vertices between two nodes of the graph correspond to neighbor regions. We propose to define two types of neighborhood: the first obtained from adjacent neighbors, the second obtained from k-nearest neighbors. Adjacent neighbors is the natural way to create a neighborhood, two regions are said adjacent neighbors if they have at least one common boundary. However adjacent neighbors are somewhat sensitive to segmentation fluctuation. From one frame to another similar objects may have different adjacent regions due to illumination or segmentation initialization changes. In order to reduce the effect of this issue on the structural representation, we also use k-nearest neighbors defined as follows; A region R_i is a neighbor of R_j only if the distance between the barycenter of R_i and R_j is among the k smallest. This kind of neighborhood is more robust to the segmentation [13], but is only optimal for rather circular regions. Indeed, in the case of region with complex shapes using the barycenter may lead to the creation of a graph edge between two regions which are not visually really neighbors.

A segmented frame can therefore be represented as a graph $G = (V,E)$ consisting of a set of vertices $V = \{v_1, v_2, \ldots, v_n\}$ and edges $E = \{e_1, e_2, \ldots, e_m\}$, where the vertices represent the cluster number labeled regions and the edges the connectivity of the regions. For the discussion below, we also introduce $\phi_i^Q = \{h|(i,h) \in E^Q\}$ which denotes all the nodes connected to a given node i in a graph Q. As an illustration,

Figure 2(b) shows a frame containing an object segmented into regions with is corresponding relational graph based on adjacent neighbors, overlaid.



(a) (b)

Figure 2: (a) The shark object and (b) its corresponding graph of adjacent regions.

2.3.1 Indirectly adding structure when building the dictionary

A first approach to add structural information when using LSA is to include the structural constraints within the clustering process itself. Here we are interested in clustering regions according to their attributes as well as the attributes of the neighbor regions. To this end, we used a clustering algorithm similar to k-medoid with a specific distance function $D(R_i^Q, R_j^D)$, equation (1). This distance function between regions R_i^Q of graph Q and R_j^D of graph D take the local structure into account.

$$D(R_i^Q, R_j^D) = L_2(H_i, H_j) + \frac{1}{\|\phi_i^Q\|} \sum_{k \in \phi_l^Q} \min_{l \in \phi_j^D} L_2(H_k, H_l) \tag{1}$$

where $L_2(H_i, H_j)$ is the Euclidian distance between histograms H_i and H_j. In order to deal with the different connectivity levels of nodes, the node with the least number of neighbors is ϕ_l^Q. This insures that all neighbor from ϕ_l^Q can be mapped to nodes of ϕ_j^D. Note that this also allows multiple mappings, which means that several neighbors of one node i can be mapped to the same neighbor of the node l.

As a result of the clustering described above, we get k clusters, which are built

upon structural constraints and visual features. Each region R_i belongs to one cluster C_l. Each cluster medoid represents a visual word for the Latent Semantic Analysis. Indeed, in this case we decided to perform clustering based on medoid element instead of the mean element. The reason for this choice was dictated by the fact that it is quite difficult to compute the characteristics of the average element of a cluster.

2.3.2 Adding structural constraints directly to the words of the dictionary

Another alternative to the construction of a visual dictionary containing information about the structure is proposed: the Relational LSA. Every possible unordered pair of clusters are considered as a visual word W, e.g. $C_3C_7 \equiv C_7C_3$, as illustrated in figure (3). Note that for example the cluster pair C_1C_1 is also a word of the dictionary, since two neighbor regions can fall into the same cluster C_l despite having segmented them into different regions before.

$$D_\nu = \{W_1, \ldots, W_\nu\}$$

$$(C_1C_1) \simeq W_1, (C_1C_2) \simeq W_2, \ldots, (C_kC_k) \simeq W_\nu$$

The size ν of the dictionary D_ν is also controlled by the clustering parameter k but this time indirectly.

$$\nu = \frac{k \cdot (k+1)}{2} \tag{2}$$

To be able to build these pairs of clusters (words), each region is labeled with the cluster number it belongs to (e.g. C_{14}). If two regions are adjacent, they are linked in an abstract point of view, which results in a graph G_i as described previously. Every Graph G_i is described by its adjacency matrix. The matrix is a square matrix $(n \times n)$ with both, rows and columns, representing the vertices from v_1 to v_n in an ascending order. The cell (i, j) contains the number of how many times vertex v_i is connected to vertex v_j. The matrices are symmetric to theirs diagonals.

The major drawback of this method is the creation of a dictionary with too many words, i.e. pairs of clusters. For example, let 1,000 be the number of clusters, the

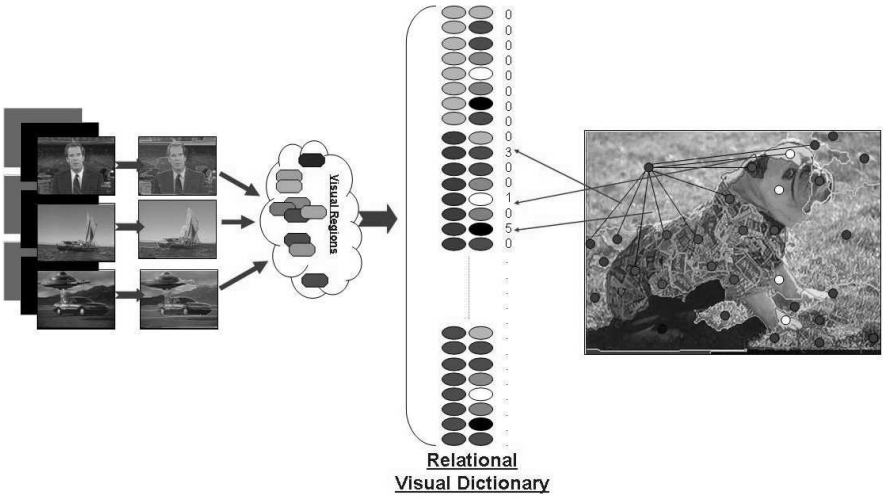


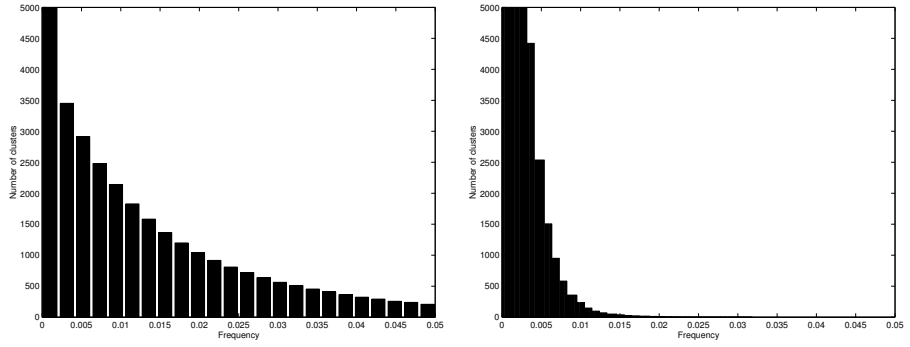
Figure 3: **Building the relational dictionary.**

	Pairs	% empty	Dictionary size without loss
k-nearest 100 clusters	5050	0.1554	4265
k-nearest 500 clusters	125250	0.7298	33847
k-nearest 1000 clusters	500500	0.8978	51161
adjacent 100 clusters	5050	0.23	3872
adjacent 500 clusters	125250	0.78	26707
adjacent 1000 clusters	500500	0.92	38935

Table 1: **Minimum size of structural dictionaries without loss.** Given values are dependent of the video and were computed on Docon's production donation to the MPEG-7 dataset.

relational dictionary reaches the size of 500,500 words. However, many words do not exist as shown in table (1) or are very rare as shown in figure (4). Thus, to keep the dictionary size low we propose to remove words that occur rarely. Given the desired number of words ν', we select pairs that occurs the most until ν' is reached. By this way, we expect to keep the most relevant pairs.

In this configuration, the LSA is also used to identify which structural information should be favored in order to obtain good generalization results. Moreover, we believe that this should improve the robustness of the method to segmentation differences among multiple views of the same object (leading to slightly different graphs).



(a) 100 clusters ⇒ dictionary size of 5050 (b) 1000 clusters ⇒ dictionary size of 500500

Figure 4: **Occurrence of cluster couples:** A bin represents the number of couples that occur at least x times. Many couples never occur or very rarely when the number of clusters is high. Given values are dependent of the video and were computed on Docon's production donation to the MPEG-7 dataset.

2.4 Latent Semantic Analysis

The LSA describes the semantic content of a context by mapping words (within this context) onto a semantic space. Singular Value Decomposition (SVD) is used to create such a semantic space. A co-occurrence matrix **A** containing words (rows) and contexts (columns) is built. The value of a cell a_{ij} of **A** contains the number of occurrence of the word i in the context j. Then, SVD is used to decompose the matrix **A** (of size $M \times N$, M words and N contexts) into three separate matrices.

$$\mathbf{A} = \mathbf{USV^T} \tag{3}$$

The matrix **U** is of size $M \times L$, the matrix **S** is of dimension $L \times L$ and the matrix **V** is $N \times L$. **U** and **V** are unitary matrices, thus $\mathbf{U^TU} = \mathbf{V^TV} = \mathbf{I_L}$. **S** is a diagonal matrix of size $L = min(M,N)$ with singular values σ_1 to σ_L, where

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_L \qquad \mathrm{S} \approx diag(\sigma_1, \sigma_2, \ldots, \sigma_L)$$

A can be approximated by reducing the size of **S** to some dimensionality of $k \times k$, where $\sigma_1, \sigma_2, \ldots, \sigma_k$ are the k highest singular values.

$$\hat{\mathbf{A}} = \mathbf{U_k S_k V_k^T} \tag{4}$$

By doing a reduction in dimensionality from L to k, the sizes of the matrices $\mathbf{U}$ and $\mathbf{V}$ have to be changed to $M \times k$ and respectively $N \times k$. Thus, k is the dimension of the resulting semantic space. To measure the result of the query, the cosine measure (*sim*) is used. The query vector $\mathbf{q}$ contains the words describing the object, in a particular frame where it appears.

$$\mathbf{q^T}\hat{\mathbf{A}} = \mathbf{q^T U_k S_k V_k^T} = (\mathbf{q^T U_k})(\mathbf{S_k V_k^T}) \tag{5}$$

Let $\mathbf{p_q} = \mathbf{q^T U_k}$ and $\mathbf{p_j}$ to be the j-th context (frame) of $(\mathbf{S_k V_k^T})$

$$sim(\mathbf{p_j}, \mathbf{q}) = \frac{\mathbf{p_q} \cdot \mathbf{p_j}}{\|\mathbf{p_q}\| \cdot \|\mathbf{p_j}\|} \tag{6}$$

The number of singular values kept k drives the LSA performance. On one hand if too many factors are kept, the noise will remain and the detection of synonyms and the polysemy of visual terms will fail. On the other hand if too few factors are kept, important information will be lost degrading performances. Unfortunately no solution has yet been found and only experiments allow to find the appropriate factor number.

When two dictionaries or more are used, for example one containing color terms through 64 bins HSV histograms and the other containing texture terms through 24 Gabor energies. The occurrence matrix A is build for each feature type. The LSA is then computed on each matrix. Similarity measures are finally independently computed for each feature type and then combined as follows:

$$sim(q,q') = w_c \times sim_{color}(q,q') + w_t \times sim_{texture}(q,q') \tag{7}$$

For simplicity $w_c = w_t = 1$ knowing that the appropriate selection of weights can be included in a training algorithm [20] for classification or a relevance feedback loop for information retrieval.

3 Experimental Results

Our proposed approaches to model a video shot thanks to latent semantic indexing are evaluated on two different tasks. First the system performance is measured in the framework of object retrieval on a set of cartoons (approximatively 10 minutes of duration) from the MPEG-7 data set. Then, its is evaluated in the context of Video-TREC feature extraction on full frames. Indeed, it would be interesting to perform the evaluation of both tasks on the same dataset, and more particularly the Video-TREC one. However, the ground truth available for Video-TREC does not feature object level annotations. Therefore, and in order to minimize the annotation effort we opted for cartoon videos.

3.1 Object retrieval

The object retrieval evaluation is conducted on Docon's production donation to the MPEG-7 dataset. Since the temporal segmentation is not available for this sequence, key-frames are selected every one second. A ground truth has then been manually established to measure the performance of the object retrieval task and 7 different objects were selected and annotated in 950 frames, see figure (5) for an illustration. The query objects are chosen as diverse as possible; some are rather simple with respect to the number of regions they consist of, while others are more complex. There are 17 to 108 possible queries per object for an overall total of 350 queries. The chosen granularity of the segmentation results in an average of about 35 regions per frame. Thus, the graphs built remain reasonably small (in term of number of nodes per graph), whereas the number of graphs (one per frame) is quite large.

Once the query is formed, the algorithm starts searching for frames which contain the query object. The query results are ordered so that the frame which most likely contains the query object (regarding the cosine measure m_c) comes first. The performance of our retrieval system is evaluated using either the standard precision vs. recall values or the mean average precision value. The mean average precision value for each object is defined as follows: we take the average precision value obtained after each relevant

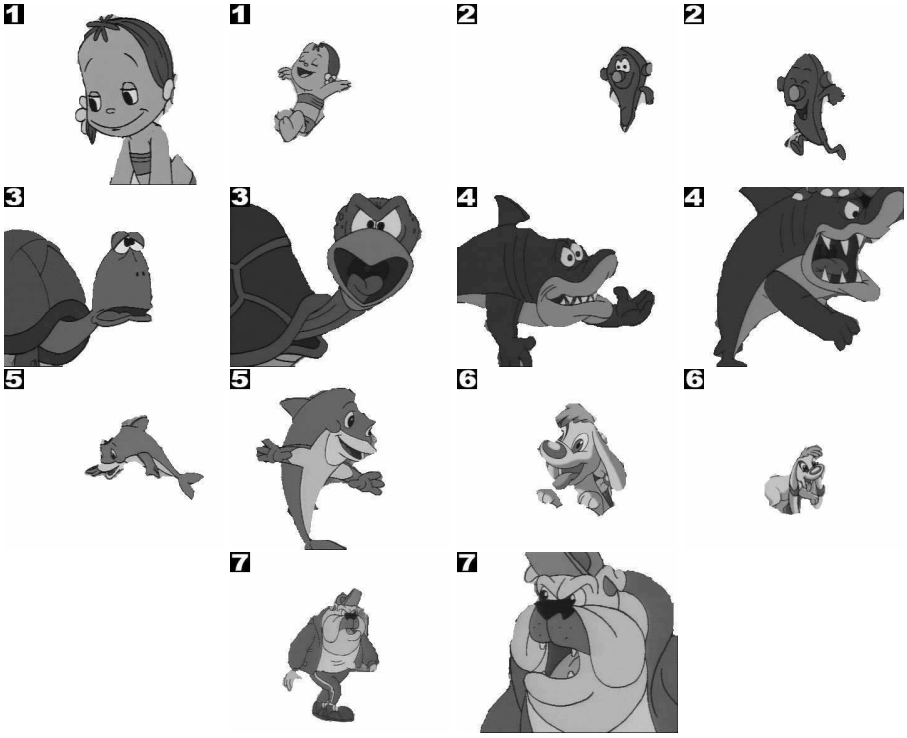
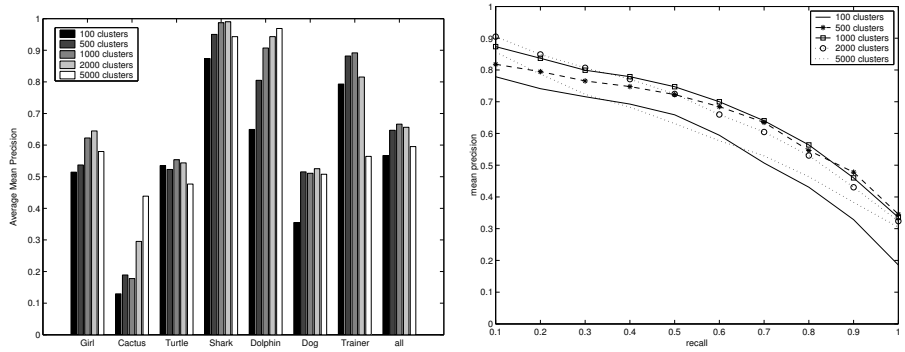


Figure 5: **The 7 query objects.** Girl, Cactus, Turtle, Shark, Dolphin, Dog and Coach

frame has been retrieved and take the mean value, over all frames retrieved. Unless stated otherwise, plots show performances of the retrieval task for the best number of factors of the LSA.

3.1.1 Impact of the number of clusters

To show the impact of the number of clusters chosen during video preprocessing on performances, we built several dictionaries containing non-structural visual words (as described in Section 2.2). Figure 6(a) shows for 100, 500, 1000, 2000 and 5000 clusters the best performances per object and the mean performances over all possible queries. Figure 6(b) shows the best precision and retrieval curves per number of clusters. They reveal the importance of the dictionary size on performances. A too small number of cluster removes differences while a number of cluster too high hides similarities. Indeed, in extreme cases, all regions are mapped to a single cluster or all regions are



(a) Best average mean values

(b) Best precision and recall curves

Figure 6: **Impact of the number of clusters on performances.** A number of cluster too small removes differences while a number of cluster too high hides similarities.

mapped to different clusters. However, the range that provides good performances is large enough between 500 and 2000 clusters, hence the number of cluster can be chosen empirically after few experiments.

3.1.2 Comparing indirectly added structure with the non-structural approach

In the following experiments, we compare the retrieval results either using a structure-less dictionary and a dictionary where we added the structural information within the clustering process as explained in Section 2.3.1. In both methods we use a cluster size of 528 (which also results in a dictionary size of 528) and we select k (the number of factors kept in LSA) so that we get best results (in this case k=25). Figure 7 shows the precision at given recall values for both cases. The curves represent an average over 4 objects. It shows that adding structural information to the clustering does not improve the non-structural approach, it even is doing slightly worse for recall values above 0.5. Based on this finding, we will not evaluate this approach further.

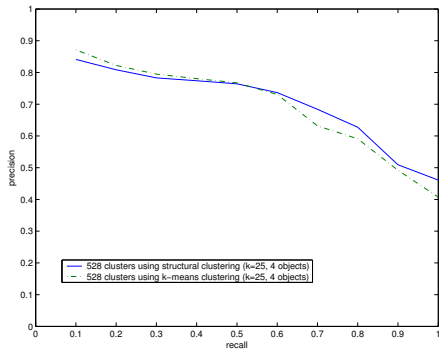
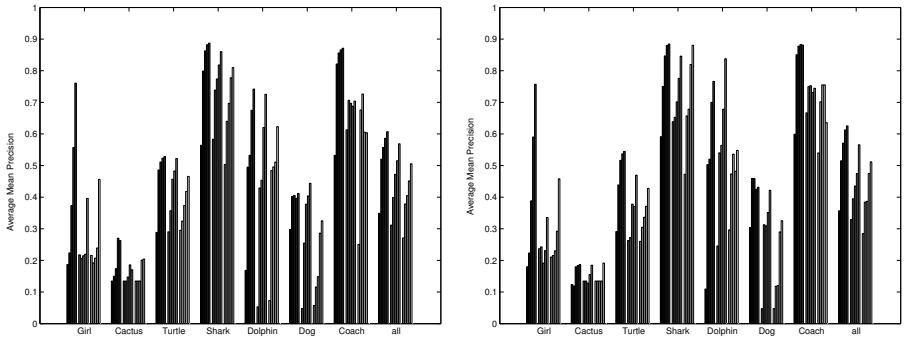


Figure 7: Retrieval performance for 4 objects queries with indirectly added structure and without.

3.1.3 Impact of LSA and the number of cluster and pairs

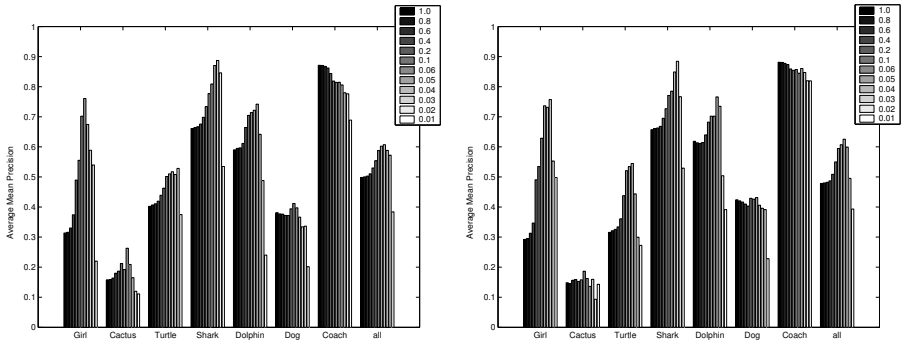
As opposed to the basic approach, the proposed Relational LSA approach does not have a dictionary size equal to the number of clusters (section 2.3.2). Both parameters (number of clusters and dictionary size) have a different effect on performances (figure 8). First, a high number of clusters leads to poor performances comparing to a smaller number of clusters. Statistically pairs occur less often than singletons, thus similar pairs are more seldom. Secondly, a high number of clusters implies that a high number of pairs are removed from the dictionary, leading to a higher loss of information. For now, it is difficult to conclude which of the two effects has the strongest impact on performances. The experiments conducted on the larger Video-TREC dataset will provide us with an answer.

Figure (9) shows the effect of the number of factors kept for the LSA on retrieval performance. For this purpose, performances of the system are computed with respect to different values of the projection size. It is important to notice that even if no method exists to find the optimal number of factors used for the projection (section (2.4)) we can approximatively select an empiric value after few experiments since the maximum is obtained in a stable area.



(a) k-nearest neighbors (b) adjacent neighbors

Figure 8: **Impact of the number of clusters and the dictionary size on retrieval performances.** For cluster sizes of 100, 500 and 1000 we select 100, 500, 1000, 2000 or 5000 words. Approaches using structure requires an initial number of cluster small enough to avoid hiding similarities.



(a) k-nearest neighbors (b) adjacent neighbors

Figure 9: **Impact of the number of factors of the LSA.** Performances are stable at their maximum when 50% of factors are kept.

3.1.4 Comparing directly structure enhanced words with non-structural words

In this section, the proposed Relational LSA approach is compared to the standard LSA. For a cluster size of 100, we compare two different ways of defining the visual words used for LSA. In the non-structural case, each cluster label represents one word, leading to a dictionary size 100. In the structural case, every possible pair of cluster label is defining a word (as explained in Section 2.3.2), so that the number of words in the dictionary is 5050. In this case, the dictionary size can be reduced to 5000

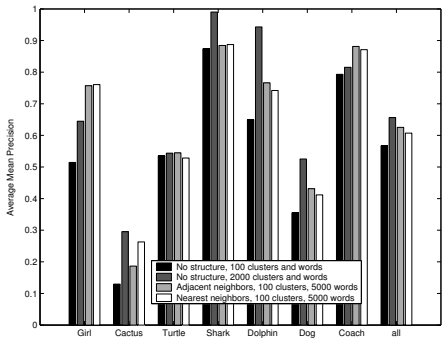


Figure 10: **Retrieval performances of proposed approaches** On complex objects (namely Girl and Coach), methods using structure provide the best performances.

without loss of information by removing non occuring pairs. Figure (10) shows the results for both approaches upon the two types of neighborhood (adjacent and k-nearest neighbors). Both methods using structure outperform the basic approach with a major improvement for a fixed number of clusters (100).

Now if we consider the best parameters, i.e.: the number of clusters and the dictionary size, for each method, results are slightly different. The mean results on average over all considered objects (350 queries) show that the basic approach with 2000 words does better than others. However, a closer look at performances on individual objects, shows that for complex objects (such as Girl and Coach) the Relational LSA does better regardless of the type of structure employed. In the contrary, on simple objects (such as the Shark, Dolphin and Dog) the basic method achieves the best retrieval results. This finding is rather coherent with our model. Indeed simple objects have very few regions and thus pairs. The extreme case were the object is composed of one single region only is a good example of query where neighborhood information is of little or no use. Additionally, as the number of object regions increases more pairs are available. In a way, constraints are relaxed, thus complex objects favor the used of relational information.

3.2 Video-TREC feature extraction

Our system is also evaluated in the context of Video-TREC [2]. One of the task at hand is to detect the semantic content of video shots. The evaluation requires annotated data. In June 2003, Video-TREC has launched a collaborative effort to annotate video

sequences in order to build a labeled reference database. The database is composed of about 63 hours of news videos that are segmented into shots. These shots were annotated with items in a list of 133 labels which root concepts are the event taking place, the context of the scene and objects involved. The tool described in [21] was used for this time-consuming task. For the purpose of this paper, we have selected 10 features among those 133 items to evaluate the performances of proposed approaches: *standing person, basketball, weather news, flower, cityscape* and *news person*. Both simple and complex semantic features were retained to evaluate our system. We used 28,000 shots for the training set and 18,000 for the test set. For each feature, test shots are ordered with respect to their detection score value. Next the average precision at 2,000 shots is computed to characterize the performance of the system for each feature. In [22, 23], we have proposed several approaches to estimate shot semantic features and compute their detection score. The k-nearest neighbors classifier on LSA features gave better performances for the semantic classification task than Gaussian mixture models and neural nets. Strong of this result we employ the k-nearest neighbors classifier to estimate the semantic content of shots.

Let N_s be the neighborhood of a shot s in the training set L, i.e. the k-nearest neighbors of s in the training set, and $y_i \in \{0,1\}^l$ the semantic value of the neighbor i. The detection score is a vector defined as:

$$d_L(s) = \sum_{N_s} sim(s,n_i) * y_{n_i} \tag{8}$$

We experimented several forms of the estimator: we normalized by $\sum_{N_s} sim(s,n_i)$ or used $y_i \in \{-1,1\}^l$ and equation (8) gave the best performances. Indeed we are computing a detection score to order shots. The lack of normalization favors shots that have really close neighbors, and thus shots for which the estimation is the most reliable.

Due to computation requirements experiments are not as extensive as on Docon's production donation to the MPEG-7 dataset. Moreover the dictionary size is limited to 2,000 words. Figure (11) shows the average precision obtained for each concept. Like in previous experiments on cartoons, Relational LSA performances decrease when the

number of cluster increase. However this is not the case for the concept *weather news* that occurs frequently in the dataset. This suggests that the truncation of the dictionary as presented in section (2.3.2) has an important impact on performances that is stronger than the effect of the dictionary size. Therefore, it appears to be better to design the system with a small number of clusters to limit the side effects of the truncation. The loss of information during the quantification process is then compensated by the use of the neighborhood. This is observed on the presented figure since a performance gain is perceived for many concepts.

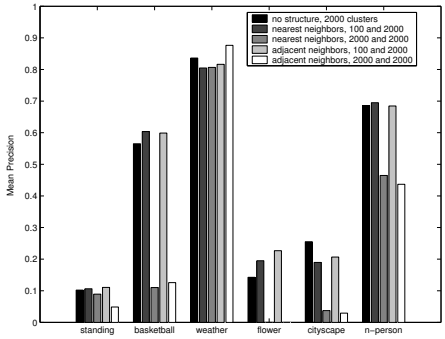


Figure 11: **Video-TREC feature detection.**

4 Conclusion And Future Work

In this paper we have presented two methods for enhancing a LSA based video object retrieval system with structural constraints (either direct or indirect) obtained from the object visual properties. The methods were compared to a similar method [1] which did not make use of the relational information between adjacent regions. Our results show the importance of structural constraints for region based object representation. This is demonstrated in the case where the structure is added directly in building the words, an approach we refer to as Relational LSA, where substantial performance increase (over 5%) is achieved when a common number of region categories is used. We have seen that the structure is of main importance when dealing with sophisticated objects in which case the presented method outperforms the basic approach by 10%. Further experiments on the task of feature detection on TV news videos have shown

the beneficial influence of the structure upon retrieval accuracy.

In this paper, we have seen how the relational LSA was able to improve performances for complex objects. We currently looking at a way to combine several approaches to obtain the best results regardless of the complexity of the query.

References

[1] F. Souvannavong, B. Merialdo, and B. Huet, "Video content modeling with latent semantic analysis," in *Third International Workshop on Content-Based Multimedia Indexing*, 2003.

[2] http://www-nlpir.nist.gov/projects/trecvid/.

[3] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society of Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[4] R. Zhao and W. I. Grosky, *Video Shot Detection Using Color Anglogram and Latent Semantic Indexing: From Contents to Semantics*. CRC Press, 2003.

[5] F. Monay and D. Gatica-Perez, "On image auto-annotation with latent space models," *ACM Int. Conf. on Multimedia*, 2003.

[6] P. Wiemer-Hastings, "Adding syntactic information to lsa," in *Proceedings of the Twenty-second Annual Conference of the Cognitive Science Society*, 2000, pp. 989–993.

[7] T. Landauer, D. Laham, B. Rehder, and M. Schreiner, "How well can passage meaning be derived without using word order," in *Proceedings of the 19th annual meeting of the Cognitive Science Society*, 1997, pp. 412–417.

[8] M. Swain and D. Ballard, "Indexing via colour histograms," *Third International Conference on Computer Vision*, pp. 390–393, 1990.

[9] M. Flickner, H. Sawhney, and al., "Query by image and video content: the qbic system," *IEEE Computer*, vol. 28, no. 9, pp. 23–32, 1995.

[10] A. Pentland, R. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233–254, 1996.

[11] G. Gimelfarb and A. Jain, "On retrieving textured images from an image database," *Pattern Recognition*, vol. 29, no. 9, pp. 1461–1483, 1996.

[12] K. Shearer, S. Venkatesh, and H. Bunke, "An efficient least common subgraph algorithm for video indexing," *International Conference on Pattern Recognition*, vol. 2, pp. 1241–1243, 1998.

[13] B. Huet and E. Hancock, "Line pattern retrieval using relational histograms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1363–1370, December 1999.

[14] K. Sengupta and K. Boyer, "Organizing large structural modelbases," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995.

[15] B. Messmer and H. Bunke, "A new algorithm for error-tolerant subgraph isomorphism detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.

[16] I. Koprinska and S. Carrato, "Temporal video segmentation: A survey," *Signal Processing: Image Communication*, vol. 16, pp. 451–460, 2001.

[17] G. M. Quénot, "Trec-10 shot boundary detection task: Cipls system description and evaluation," in *The 10th Text REtrieval Conference (TREC)*, 2000.

[18] P. Felzenszwalb and D. Huttenlocher, "Efficiently computing a good segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1998, pp. 98–104.

[19] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice Hall, 1988.

[20] K. Kira and L. Rendell, "A practical approach to feature selection," in *Proceedings of the 9 International Conference on Machine Learning*, 1992, pp. 249–256.

[21] C.-Y. Lin, B. L. Tseng, and J. R. Smith, "Video collaborative annotation forum: Establishing ground-truth labels on large multimedia datasets," in *Proceedings of the TRECVID 2003 Workshop*, 2003.

[22] F. Souvannavong, B. Merialdo, and B. Huet, "Latent semantic indexing for video content modeling and analysis," in *The 12th Text REtrieval Conference (TREC)*, 2003.

[23] F. Souvannavong, B. Merialdo, and B. Huet, "Latent semantic analysis for semantic content detection of video shots," in *International Conference on Multimedia and Expo*, 2004.