Analysis of Vector Space Model and Spatiotemporal Segmentation for Video Indexing and Retrieval

Eric Galmar
Institut Eurécom, Département multimédia
2229 Route des Crêtes,
Sophia-Antipolis, France
eric.galmar@eurecom.fr

Benoit Huet
Institut Eurécom, Département multimédia
2229 Route des Crêtes,
Sophia-Antipolis, France
benoit.huet@eurecom.fr

ABSTRACT

Region-based video indexing systems have opened up new possibilities for the description of visual content. However, these systems are affected by spatial variations on the regions obtained from image segmentation algorithms and by the complexity of region matching techniques. In this paper, we propose to enhance these systems with the use of spatiotemporal regions. The indexing framework studied for that purpose is based on the Vector Space Model (VSM), which enables efficient and compact shot representation with count vectors. We analyse the properties of the VSM and show that shot description can be improved by considering spatio temporal representations. For evaluation, we further compare the performance of the system using the spatiotemporal and the keyframe approach. Experimental results show that the spatiotemporal approach is advantageous in terms of retrieval performance and robustness of the description.

Categories and Subject Descriptors

H.3.1 [**Information Storage And Retrieval**]: Content Analysis And Indexing—*Indexing Methods*; I.2.10 [**Vision and Scene Understanding**]: [Video Analysis]

General Terms

Algorithms,Design,Experimentation

Keywords

Region-Based Video Indexing and Retrieval, Video Analysis, Vector Space Model, Spatiotemporal segmentation

1. INTRODUCTION

Recent advances in multimedia technologies have triggered the creation of high volume video data. Textual annotation is generally unsufficient to take into account complex variations of video. Hence, the archiving of this huge information has spurred the development of efficient techniques to represent visual content. Research in content based video retrieval (CBVR) is aimed at addressing these challenges. The database is organized at shot level. Visual content of shots currently includes image features such as color, texture, shape and motion, as defined in the MPEG-7 standard. A key issue of CBVR systems is to extract and organize these features in a compact index.

Earlier video indexing systems were based on global image features [9], but these systems had limited performance since shots with different objects may share same global features. It is now well established that a good video index should capture both spatial and temporal content of the scene. This paradigm has lead to region-based systems, which objective is to bring shot representation closer to human perception. Traditionally, these regions are obtained by spatial segmentation on representative keyframes [11, 1, 2, 6, 14, 12]. The main shortcoming is that region representation remains sensitive to the segmentation, so that the comparison of indexes is affected by the spatial variations on the segmented regions. In addition, information on the temporal evolution of the regions is lost. More recently, research efforts have been carried out on spatiotemporal regions. More evidence on structure and motion can be obtained by collecting region information in multiple frames. Most methods first segment every image and then extend them to the spatiotemporal domain through the use of matching techniques [8, 6] or motion estimation [3]. For instance, in the VideoQ system, regions are initialized from color image segmentation and are then tracked using the optical flow. However, the representation is impaired from traditionnal motion estimations problems, i.e. the quality of the image segmentation and estimated flow depend on each other. Pure spatiotemporal approaches consider the shot as a single 3D volume and extract descriptors at pixel level. Then, spatiotemporal regions are built by means of grouping techniques considering simultaneously spatial and temporal information, such as mean-shift clustering of video patches [5] or graph cuts on a pixel affinity matrix [10]. The advantage of these methods is that there is no further need to track regions between frames.

In this paper, we analyse a region-based indexing system based on the Vector Space Model. This approach has proved to be successful for many content-based image applications, such as categorization [4], and video indexing and retrieval [12]. Shots are described by an index of visual terms, which are representative perceptual elements of the video. Similarity between shots can be obtained by comparing the occurrences of the visual terms. However, the quality of the shot description is often altered by variations of segmentation

CIVR'07, July 9–11, 2007, Amsterdam, The Netherlands.

algorithms. Although no segmentation algorithm can give perfect results, spatiotemporal segmentation methods provide regions that remain temporally consistent over time. It is interesting to examine how well this feature can improve shot description with the Vector Space Model. In order to favor the creation of more robust indexes without adding significant complexity to the indexing system, we propose to join spatiotemporal representation to the Vector Space Model, relying both on spatial and temporal region occurrences. This paper is organized as follows. In section 2, we introduce the video retrieval and indexing system. Then, in section 3 we present the spatiotemporal segmentation method used for describing shots. Section 4 introduces the Vector Space Model and its adaptation for spatiotemporal shot representation. Finally, in section 5 we analyze in depth the properties of the indexing model and compare spatiotemporal and traditional keyframe approaches on the task of video shot retrieval.

2. VIDEO INDEXING AND RETRIEVAL SYSTEM

In this section, we describe the system used for video shot indexing and retrieval. The video database is composed of a collection of video shots which are annotated semantically at shot level. The first task is to prepare the set of representative elements (or visual terms) that will be used to represent video shots, which we refer as the visual dictionary. To do so, shots are segmented into homogeneous volumes with the method described in section 3.1. At the same time, we store the region descriptors of the extracted regions. The overall set of descriptors is then clustered to obtain the representative elements. Within this framework, indexing of a new shot is performed in two steps. First, the shot is segmented into regions and region descriptors are extracted. Secondly, these descriptors are quantized to the nearest visual terms to obtain compact shot signatures. Search and retrieval tasks become now easy as we just need to compare shot signatures. The overall framework is shown Fig.1.

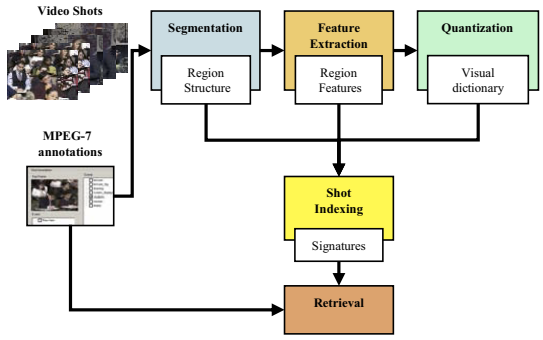


Figure 1: The Video Indexing and Retrieval System.

3. SPATIOTEMPORAL REPRESENTATION

In our framework, shot content is described by its representation into regions. Ideally, each region should emphasize one part of the perceptual visual content, while being still homogeneous with respect to the extracted features. In addition, the region extraction process should be fast enough to be applied on a large number of shots and frames in a reasonable time. Spatiotemporal region representation aims to add more robustness to the extracted regions, and thus to enhance shot description. Much more evidence on the relevance of regions can be obtained by gathering information from a sequence of frames. Regions with temporal coherence are most likely to be part of important elements of the shots, such as moving objects or static backgrounds. The feature extraction stage benefits from spatiotemporal representation, since the influence of spatial segmentation variations is reduced when considering multiple frame regions.

3.1 Efficient Spatiotemporal Segmentation

Spatiotemporal regions can be extracted efficiently with the method proposed in [7]. The approach has the ability to segment various type of scenes, including both static and dynamic contents. One important advantage compared to other spatiotemporal segmentation algorithms is the absence of computationally intensive clustering or optimization algorithms. These are implicitly replaced by splitting the segmentation process into several stages, using different graph merging algorithms adapted to the level of grouping.

The workflow is composed of 5 steps shown Fig.2. In the following, we denote by $S_{i\to j}$ the spatiotemporal segmentation from frame i to j and S_i its projection on frame i.

The segmentation is first initialized on the first frame of the shot I_0. This step sets approximately the level of spatial details. Once the segmentation initialized, the method processes frame pairs. For each new frame I_{k+1}, we create a set of over-segmented spatial regions O_{k+1}. Ideally, they correspond to a partition of the final regions S_k, except some new regions induced by motion. These regions are now spatially and temporally grouped with the current segmentation $S_{0\to k}$ to build the new segmentation $S_{0\to k+1}$. This grouping is done in three steps. To take into account region motion, we create new temporal edges linking regions in S_k to O_{k+1} by tracking feature points between frame pairs. By considering statistical properties of feature points within region couples, we can group most of dynamic regions. The linkage of remaining static regions is done with a spatiotemporal merging algorithm. The merging is performed on a pixel neighborhood, considering both local and global properties of the current segmentation $S_{0\to k}$. As the projected segmentations S_k and S_{k+1} become close after this stage, we finally compare their region adjacency graphs to check the validity of new regions.

In this way, we achieve incremental grouping of spatiotemporal regions by considering different levels of interaction between pixels, frame regions and spatiotemporal regions.

Figure 3 illustrates spatiotemporal representation on Docon and the lecture videos (cf. section 5). In Fig.3(a), the lecturer is leaving his chair, moving from the left to the right of the scene. Among the extracted regions, we can find relevant elements such as the head of the lecturer, the right part of the jacket, and background elements. In Fig.3(b) featuring an agitated shark, the regions represent the mouth, the belly, the hat and the back of the shark. Some regions can be simply characterized by their color, while others should be also described by texture. We can observe that the segmentation process absorbs small details while preserving global region properties. Thus, a certain part of inhomogeneity is tolerated, in order that the method balances the span and the homogeneity of the spatiotemporal regions.

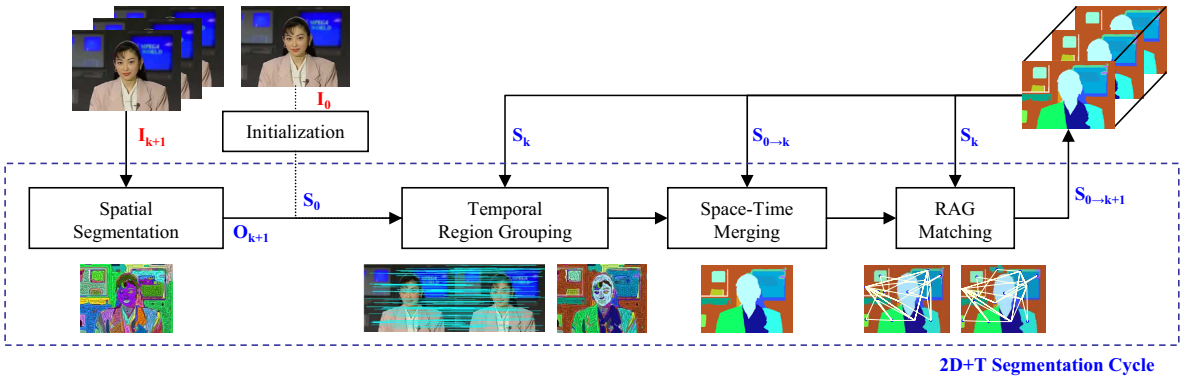
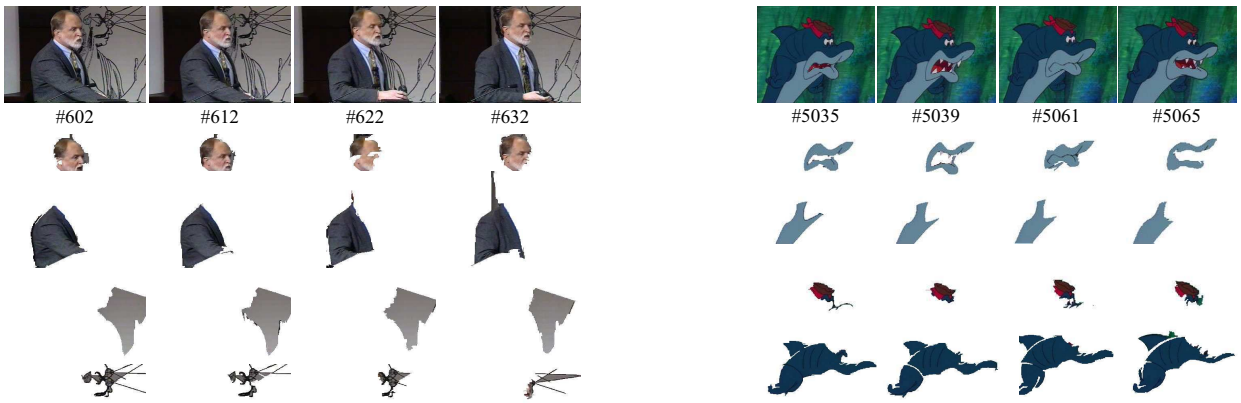


Figure 2: Spatiotemporal Segmentation Scheme.



(a) Lecture video (b) Docon video

Figure 3: Examples of spatiotemporal representation.

As we consider an automatic segmentation process, it is rather difficult to extract perceptually meaningfull regions with high accuracy. Indeed, we can notice in Fig.3(a) that part of the head is missing due to very low contrast with the white background and that the end of the jacket sleeve is cut since the microphone prevents the grouping of top and bottom jacket parts. Refining boundaries of spatiotemporal regions will induce much more processing time, whereas this will be unneeded in most cases as we do not consider to extract region and object shapes, but instead global region descriptors such as texture and color.

Compared to image segmentation methods, spatiotemporal representation brings to light both common aspects of video shots and elements that evolve temporally. In the example Fig.3(a), the lecturer elements, such as the head and the jacket are viewed in similar conditions, undergoing few temporal changes, whereas the background elements are progressively covered by the character. Less changes occur within the second shot. This illustrates that each frame region can be seen as one instance of a latent visual content, underlining spatial and temporal changes that may occur within the shot.

3.2 Feature Extraction

To evaluate the influence of the segmentation for the keyframe and spatiotemporal approaches, we use only spatial region features. Color is represented by HSV histograms with respectively 8, 4 and 4 bins for hue, saturation and value. Texture is captured by the mean energy and variance of a bank of 24 Gabor filters. For the spatiotemporal approach two ways of extracting features are considered :

- Compute a single descriptor for each spatiotemporal region. This can be seen as averaging the descriptors through time (S-extraction).
- Compute a descriptor for each projected frame region, which requires more storage for preprocessing visual terms (F-extraction).

4. VECTOR SPACE MODEL

There are usually two strategies for region-based indexing. On the one hand, structural approaches organize shots into a hierarchy of regions. In this case, regions are the elementary units of comparison [8]. On the other hand, the Vector Space Model describes the whole shot with visual terms relative to extracted region features. In the following section, we explain how the VSM can exploit both spatial and temporal redundancies existing in the shot representation.

4.1 Visual Dictionary Construction

The VSM approach comes from text document analysis research area. Similarly to a text document, one shot can

be described by a count vector, where the count values are the number of occurrences of visual terms. The reason behind this representation is that the distribution of visual terms gives information on the latent semantics of the shot. These terms are obtained by clustering region descriptors extracted over the whole database. K-means algorithm is used for clustering. The resulting clusters constitute a visual dictionary which is used to index the shots.

4.2 Indexing

The principle of the indexing process is shown Fig.4. First spatiotemporal regions are extracted for each shot using the spatiotemporal segmentation algorithm. Then for each type of feature F, region descriptors are quantized to their nearest terms in the dictionary. The assigned number of neighbours depends on the quantization error. If the descriptor is very close to its nearest term, only the nearest term is counted. Otherwise, when the descriptor is approximately distant of several terms, all these terms should be counted. Let f_r^S a region descriptor of one shot, and $(f_i^D)_{i=i_1 \dots i_K}$ the ordered K nearest visual terms of f_r. The set of counted terms $nearest(f_r^S)$ is :

$$nearest(f_r^S) = \left\{ i_k | d_F(f_r^S, f_{i_k}^D) < T d_F(f_r^S, f_{i_1}^D) \right\} \quad (1)$$

where d_F is the distance function for feature F. This soft quantization enables to deal with spatial and temporal variations of the visual content. Firstly, the closest visual term can change between different instants. Secondly, spatial parts of one region may be better described with different terms. When mapping to multiple terms, the maximum number of neighbours K should be kept small with respect to the dictionary size, in order that the count vectors still remain discriminative. In our observations, up to 10 clusters can be chosen for less than 2000 visual terms.

The choice of the dictionary size itself is also an important factor. Augmenting the number of clusters reduces the quantization error and leads to more accurate descriptions. However, the clustering process requires more examples and computational time.

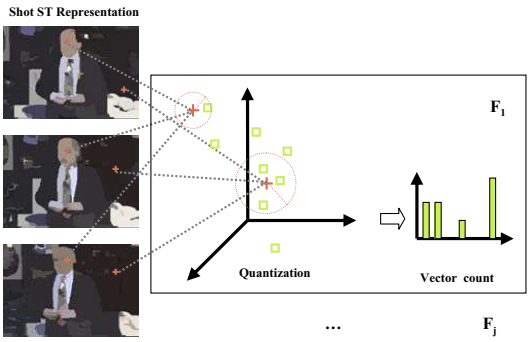


Figure 4: Vector Space Model for shot indexing.

4.3 Retrieval

Once indexed, shots can be compared very efficiently considering their signatures. The Cosine distance can naturally capture the relative proportion of common visual term occurrences between two shots. To obtain a unique similarity measure, we fuse the similarities from different modalities with a weighted sum. For simplicity, each feature is given equal weight in our experiments.

5. EXPERIMENTS

The evaluation of the spatiotemporal approach is conducted on two videos : the Docon's cartoon from the MPEG-7 dataset, and a lecture video from the open video project[1]. Each video has its own challenges for segmentation and indexing. The segmentation is more accurate in the cartoon video, but different objects can share the same environment or interact with each other (turtle, dolphin, shark), which complicates the indexing task. In the second video, we can find scenes with static content (students, screen, drawing). The lecturer is moving within the scene and is seen from close-up or wide-angle shots (lect1 and lect2). An illustration of typical annotated shots is shown Fig.5. To obtain enough occurrences of each element and variable visual content, each video is subsampled into nearly 1000 shots.

5.1 VSM Analysis

Segmentation and clustering quality are key elements of the Vector Space Model. Extracted regions should be representative of the shot, while shot count vectors should be characteristic of the visual categories. For this reason, we conducted an experiment where we study the influence of these elements on the VSM performance. More precisely, we consider two factors:

- The granularity and homogeneity of shot regions,
- The visual dictionary representativeness.

The former factor rules the segmentation quality. Segmentation algorithms usually aims to obtain homogeneous regions while limiting their number, but each method has its own way to fuse different sources of information to obtain perceptually coherent regions. Then, it is generally difficult to specify the degree of homogeneity of the regions and the granularity of the segmentation at the same time. To have more control on this factor, we choose to fix the segmentation layout utilizing a grid. In this way, we indirectly set the region homogeneity with respect to each feature, as refining the grid will lead to more homogeneous content on the average. In total, 7 scales, including 4 to 512 regions are considered.

Besides segmentation properties, visual dictionary can be depicted by the capacity of visual terms to fit each visual category. To examine underlying dictionary properties, we propose different measurements:

- The distance of one region to its nearest clusters, which we denote as *quantization error*. When indexing shots, the error depends on the distribution of the visual terms and of the number of clusters.
- The proportion of occupied bins in the count vector, which we call as *count density*, which summarizes the distribution of shot regions over visual terms.

These measures are computed on the whole video dataset and averaged on each semantic category.

Figure 6 illustrates the overall analysis results considering different dictionary sizes. Figure 6(b) shows that increasing

[1]http://www.open-video.org/

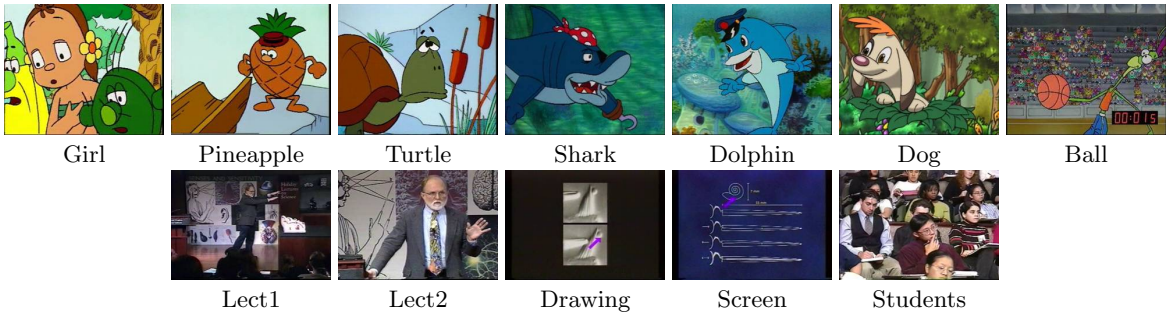


Figure 5: Examples of annotated shots of the Docon cartoon and the lecture video.

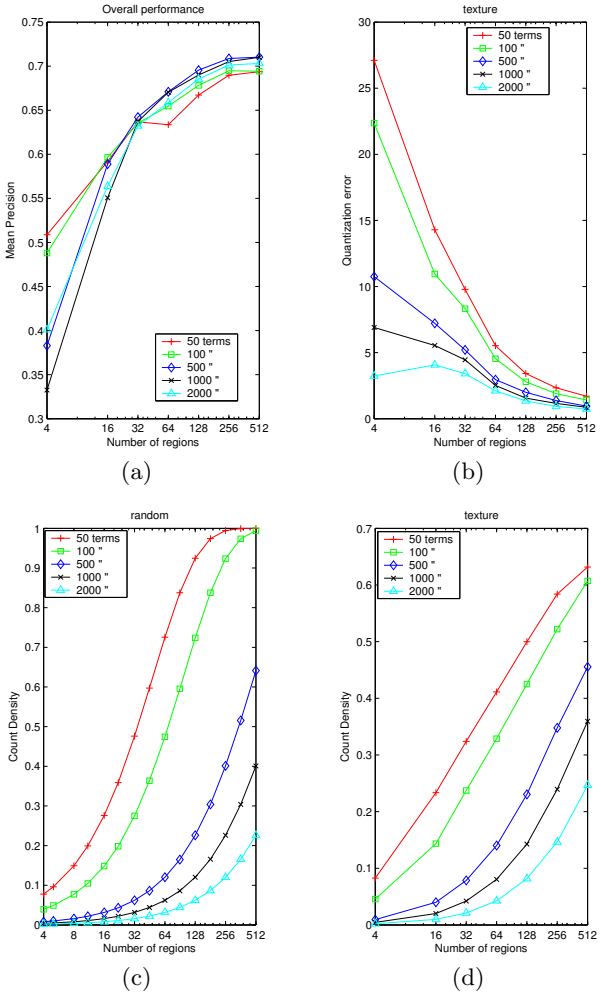


Figure 6: Analysis of the VSM for the texture modality - Example of the lecture video. (a) Overall retrieval performance. (b) Overall quantization error. (c) Exact count density for the random model. (d) Overall count density.

the number of regions and the number of terms diminish the quantization error. First, this is inherent to the clustering process : more terms results in more compact clusters. Secondly, decreasing the grid scale leads to more homogeneous regions and therefore accurate descriptors. Besides the quantization error, we analyse also the creation of the count vectors. As a reference, we compute the expected count density values when each region is indexed to a random term. In this case, the count density can be modeled as a Markov Chain. Let X_n the number of occupied bins after adding n counts, and K the dictionary size. The set of possible states is $S = 0, \ldots, K$. The Markov chain is defined by its initial value $X_0 = 0$ and its transition matrix P :

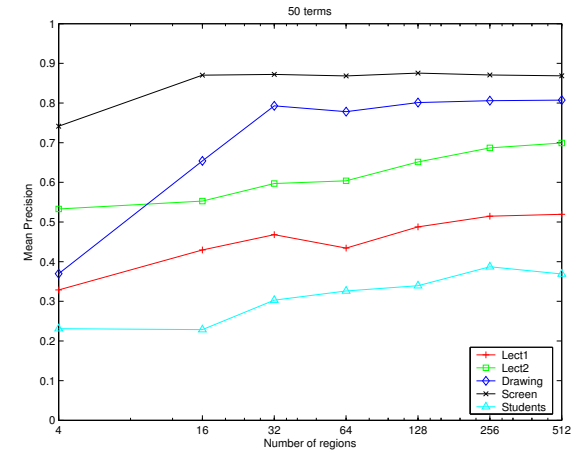
$$P_{ij} = Pr(X_{n+1} = j | X_n = i) = \begin{cases} \frac{i}{M} & \text{if } j = i \\ 1 - \frac{i}{M} & \text{if } j = i + 1 \\ 0 & \text{else} \end{cases} \quad (2)$$

The count distribution is finally given by :

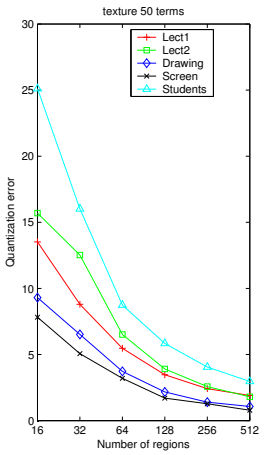
$$Pr(X_n = j) = P_{0j}^{(n)} \quad (3)$$

At this point we compare the density counts for the random model Fig.6(c) and for the dictionary of the lecture video Fig.6(d). Up to 500 terms, the count density is at least 25 percent lower than for the random model. On the contrary, when using more than 1000 clusters, the curves are very close to each other. This means that for small dictionary sizes, we obtain numerous spatial cooccurrences of the same visual terms, so that the number of these cooccurrences become important when comparying vector counts. For a large number of terms, regions are typically assigned to different clusters. In consequence, the added counts are unrelated and the density is close to the one of the Markov process. The effect of these features on the retrieval performance is shown Fig.6(a). When considering few and inhomogeneous regions, the quantization process is quite unstable leading to low and variable precision rates. Reducing the quantization error by augmenting the number of clusters does not help in this case as the count vector becomes very sparse, altering the comparison between shots. When more regions are available, the performance is less sensitive to the dictionary size. If the dictionary is small, the high number of coocurences gives dense but distinctive signatures. Otherwise, the signatures remain discriminative in spite of the reduced cooccurrences as we use accurate visual terms.

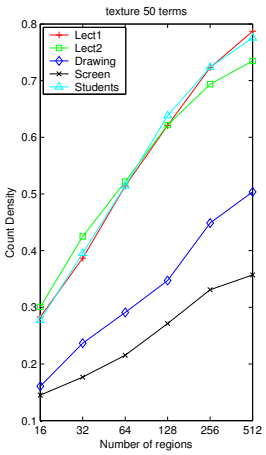
Several general trends can be drawn from this analysis. Firstly, the clustering and indexing task are more robust when considering numerous and relevant regions, boosting the accuracy of the visual terms in the quantization stage, and leading to more discriminative shot signatures thanks to the term redundancies. Secondly, the dictionary size can be choose to have intermediate count density values (0.3 to 0.5), balancing spatial redundancies and visual term accuracy.



(a) Retrieval Performance

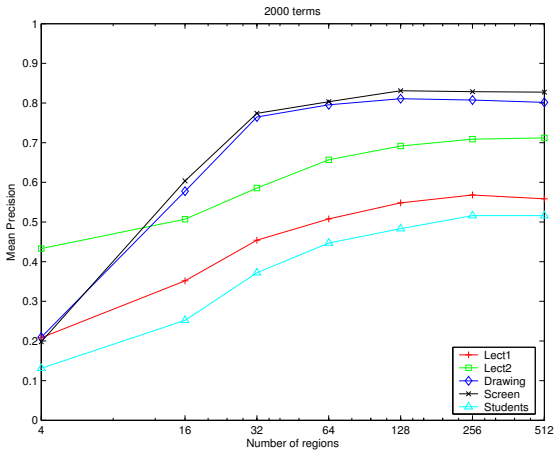


(b) Quantization error

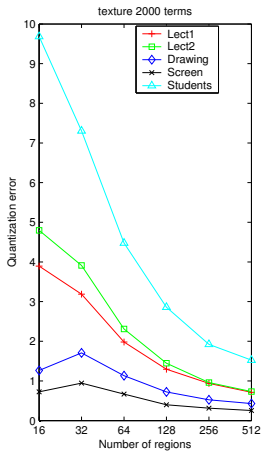


(c) Count density

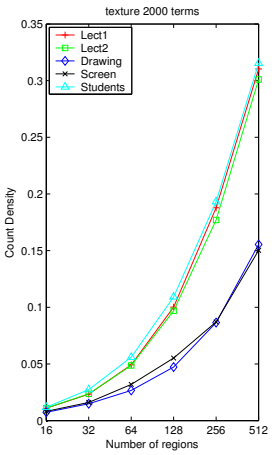
Figure 7: Analysis of the VSM for different categories with 50 visual terms - Example of the lecture video.



(a) Retrieval Performance



(b) Quantization Error



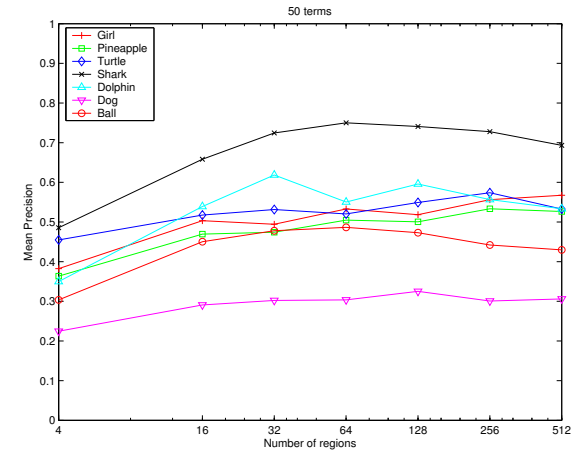
(c) Count density

Figure 8: Analysis of the VSM for different categories with 2000 visual terms - Example of the lecture video.

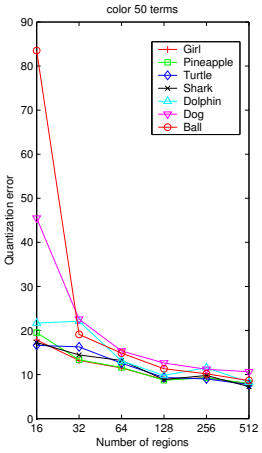
After looking at the general properties of the VSM, we now examine its behaviour with respect to different visual categories. Figures 7 and 8 show that the scenes with stable and specific visual content, such as screens, drawings are likely to be integrated efficiently in the VSM. Indeed, these categories have both small error rates Fig.7(b)-8(b) and sparse vector counts Fig.7(c)-8(c). Good retrieval results are obtained for all number of clusters, which reveals that the shots are indexed efficiently with a few category-specific terms in the dictionary. In some categories, such as students, the regions are rather inhomogeneous. As shown by the high quantization error and density of their vector counts, they are finally improperly represented in the visual dictionary. As noticed in Fig.6, augmenting the dictionary size does not lead to significant enhancement while requiring more regions. These observations are also verified in Fig.9. The shark object is clearly described by a few representative terms Fig.9(c), thus obtaining good retrieval results. In the opposite, ball and dog categories have poorer representation in the dictionary with the highest quantization errors Fig.9(b), resulting in weak performance, around 0.3 and 0.4 respectively. For the other categories, the relation between the measures and the performances is not as clearly manifested. What should be taken under consideration is the interaction between categories and the overlap of visual terms. Indeed, some categories can share same visual environment such as the couples girl-ananas or shark-turtle. The first one has large background areas in common, which is manifested through low and similar quantization error. It is also the case in the second couple, but the turtle can appear under different views and also other environments. In consequence, a query featuring a turtle may retrieve shots featuring the shark before some other shots where the turtle appears.

These considerations illustrate how the VSM enables to reduce the video contents to a small amount of visual terms. The construction and comparison of these terms depend most of the segmentation properties. If the regions are described accurately, compact and discriminative representation can be obtained from a large range of dictionary sizes. Otherwise the model is penalized when considering variable categories.

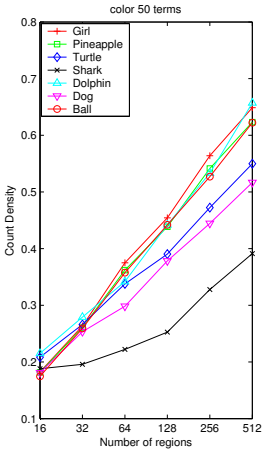
5.2 Comparison



(a) Retrieval Performance



(b) Quantization Error



(c) Count density

Figure 9: Analysis of the VSM for different categories with 50 visual terms - Example of the Docon video.

In this section, we evaluate the contribution of the spatiotemporal approach to the VSM. For this purpose, we compare the spatiotemporal representation with keyframe regions obtained from well-known segmentation algorithms: watersheding [13], which extracts homogeneous color regions and the edgeflow algorithm used in Netra-V system [6] which balances color and texture. We evaluate the algorithm on different number of frames, except for the edgeflow technique which is too computationally intensive to be used on multiple frames.

Results for the two videos are shown Fig.10(a-b). The spatiotemporal method performs well compared to the edgeflow and the watershed algorithms, and thus for every object. Regarding watershed segmentation, good results are obtained for homogeneous color objects such as shark, turtle and static rich colored scenes. However, the results can decrease dramatically for scenes with variable spatial contrast and textured areas (screen, drawing). Not suprisingly, the edgeflow method performs better on this type of scenes.

Globally, we observe gains of 7% and 12% in retrieval performance between the best image segmentation and the best spatiotemporal method, for the lecture and Docon videos respectively. The difference is justified by the fact that in the former video, the edgeflow method can give reasonably good results for categories which contain well-defined textured elements, such as the lecturer, drawing and screen. However, it fails to describe accurately scenes with more spatial variations, such as students.

Considering multiple frames (W5) or region volumes contributes to the quality of the shot indexes. In the first case the density of the shot indexes is increased, so that more common terms can be found. In comparison, in the spatiotemporal representation (STG-S), spatial and temporal variations are attenuated by extracting descriptors on the full volumes. The advantages of these two approches can be combined by extracting frame descriptors from the spatiotemporal regions (STG-F). In this way, we capture the temporal evolution of the region descriptors. We observe that the extraction process slighty enhances the retrieval performance, up to 6% with respect to STG-S methods with similar number of frames. More precisely, this improvement concerns categories whose visual content undergo important variations between scenes, such as the ball, the dog and girl scenes. Actually, the effect is to accumulate more confidence on the quantization process, as the visual terms are selected using several descriptors from the same volume. This helps to distinguish common terms that remain stable to scene changes from the others.

Another advantage of the spatiotemporal representation is that good results can be achieved considering far less regions than the other methods, as shown in table 2. This is noticeable for processing large databases.

Method	Description
STG-F*	Spatiotemporal segmentation, F-extraction
STG-S*	Spatiotemporal segmentation, S-extraction
W*	Watersheds, regions cumulated over * frames
E*	Edgeflow, regions cumulated over * frames

Table 1: Segmentation algorithms.

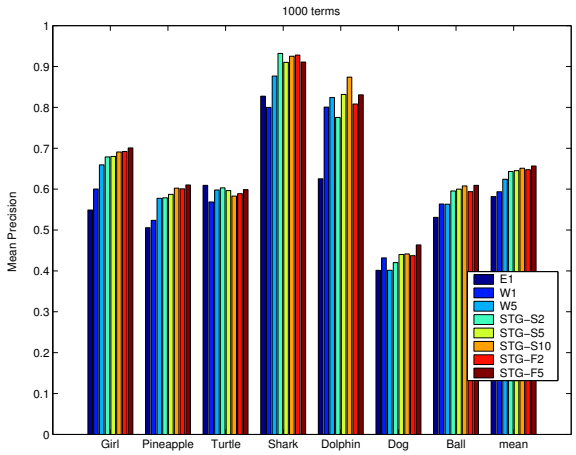
Method	E1	W1	W-5	STG-F2	STG-F5	STG-F10
Nb. reg.	90	135	672	52	88	140

Table 2: Average number of regions for the segmentation algorithms.

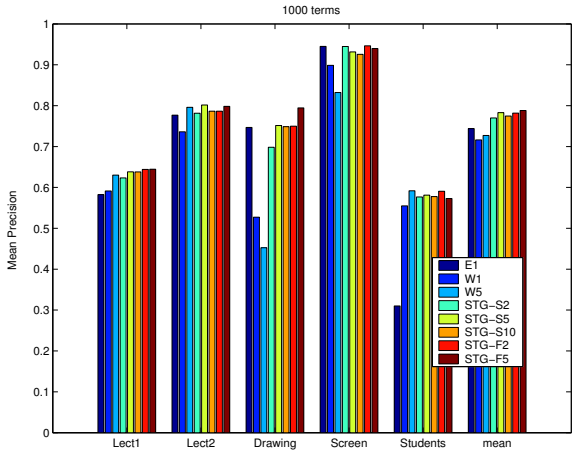
Figure 11 also points out that the performance of spatiotemporal methods does not depend much on the size of the visual dictionary. The results are averaged on all Docon's objects for several dictionary sizes. They need between 100 and 500 terms to reach their best performance, whereas a smaller dictionary can be used for watershed segmentation. This reveals that slightly more visual terms are needed to represent each spatiotemporal region. As noticed in section 4 this is not a disadvantage as it adds robustness to the scene variations.

6. CONCLUSIONS

In this paper, we have demonstrated how spatiotemporal segmentation and Vector Space Model can be combined to index video shots. Experimental results conducted on various video scenes reveal that the performance of the VSM tightly depends of the region extraction process, as the use



(a) Docon video



(b) Lecture video

Figure 10: Retrieval results for different segmentations. The dictionary contains 1000 visual terms.

of accurate region descriptors enhance the specifity of the shot signatures.

In this context, we show that the VSM benefits from spatiotemporal representation. Spatiotemporal regions emphasize relevant spatial and temporal redundancies between shots, offering more robustness to scene changes. The comparison with image segmentation methods shows that the new representation leverages retrieval performance, and that more consistent results are obtained between different visual scenes.

Future work will be to further study spatiotemporal segmentation for object-based retrieval system. Therefore, we plan to consider object spatiotemporal attributes and relationships for advanced description and comparison of video shots.

7. ACKNOWLEDGMENTS

This research was supported by the European Commission under Contract FP6 027026-K-Space.

8. REFERENCES

[1] E. Ardizzone, M. La Cascia, and D. Molinelli. Motion and color based video indexing and retrieval. In *ICPR*, pages III: 135–139, 1996.

[2] C. Carson, M. Thomas, and S. Belongie. Blobworld: a system for region-based indexing and retrieval. In *VISUAL*, pages 509–516, 1999.

[3] S. Chang, W. Chen, W. Meng, H. Sundaram, and D. Zhong. Videoq: An automatic content-based video search system using visual cues. In *ACM MM*, 1997.

[4] C. Csurka, C. Dance, L. Fan, J. Williamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV*, pages 59–74, 2004.

[5] D. DeMenthon and D. Doermann. Video retrieval using spatio-temporal descriptors. In *ACM MM*, pages 508–517, 2003.

[6] Y. Deng and B. Manjunath. Netra-v toward an object-based video representation. *IEEE Trans. CSVT*, 8(5):616–627, 1998.

[7] E. Galmar and H. Huet. Graph-based spatio-temporal region extraction. In *ICIAR*, pages 236–247, 2006.

[8] J. Lee, J. Oh, and S. Hwang. Strg-indexing, spatio-temporal region graph indexing for large video databases. In *ACM SIGMOD*, pages 718–729, 2005.

[9] W. Niblack and al. The qbic project: querying images by using color, texture, and shape. In *SPIE*, volume 1908, pages 173–187, 1993.

[10] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *Proceedings of the Sixth International Conference on Computer Vision (ICCV98)*, pages 1154–1160, Bombay, India, Jan. 1998.

[11] J. Smith and S. Chang. Visualseek, a fully automated content-based system. In *ACM Multimedia*, pages 87–98, 1996.

[12] F. Souvannavong, B. Mérialdo, and B. Huet. Region-based video content indexing and retrieval. In *CBMI*, 2005.

[13] L. Vincent and P. Soille. Watersheds in digital space: an efficient algorithm based on immersion simulations. *IEEE Trans. PAMI*, 13(6):583–598, 1991.

[14] J. Wang, J. Li, and G. WiederHold. Simplicity: Semantics-sensitive integrated matching for picture libraries. *IEEE Trans. PAMI*, 23(21):947–963, 2001.

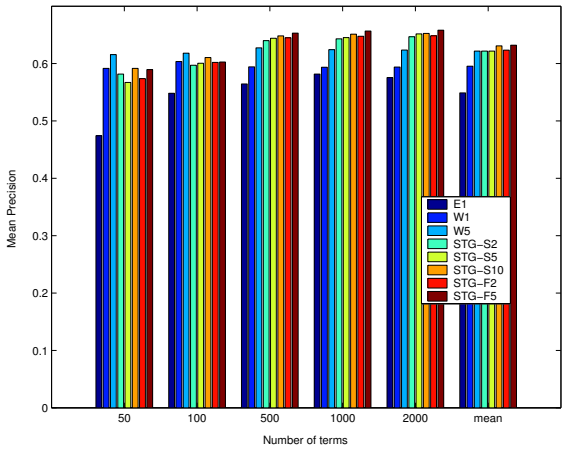


Figure 11: Retrieval results for several dictionary sizes.