Latent Semantic Indexing for Video Content Modeling and Analysis

Fabrice Souvannavong, 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

Abstract

In this paper, we propose to adapt latent semantic indexing (LSI) to model video contents. This well-known technic used to describe text documents provides a rich and efficient representation of the content. We include in the original method a way to efficiently include various features extracted from the video content. In particular we focus on color and texture features. The distributions of LSI features among semantic classes is then estimated to detect concepts present in video shots. K-Nearest Neighbors and Gaussian Mixture Model classifiers are evaluated and compared. Finally, performances obtained on LSI features are compared to a direct approach based on raw features that are color histograms and Gabor's filter energies.

Keywords: *Latent Semantic Indexing, Video Content Analysis, Gaussian Mixture Model, Kernel Regression*

1 Introduction

With the growing of numeric storage facilities, many documents are now archived in huge databases or extensively shared on the Internet. The advantage of such mass storage is undeniable, however the challenging tasks of content indexing and retrieval remain unsolved, especially for video sequences, without the expensive human intervention. Many researchers are currently investigating methods to automatically analyze, organize, index and retrieve video information [1, 5]. This effort is further underlined by the emerging Mpeg-7 standard that provides a rich and common description tool of multimedia contents. It is also encouraged by Video-TREC which aims at developing video content analysis and retrieval.

One Video-TREC task focuses on the detection of high-level features in video shots; such features include *outdoors, news subject, people, building, ...* To solve this problem, we propose to model the video content with Latent Semantic Indexing. Then based on these new features, we train two classifiers to finally detect semantic concepts. Performances of the K-Nearest Neighbors and Gaussian Mixture Models classifiers are compared and provide an evaluation framework to evaluate the efficiency of Latent Semantic Indexing for video content modeling.

Latent Semantic Analysis has been proven effective for text document analysis, indexing and retrieval [2] and some extensions to audio and image features were proposed [3, 7]. In [6], we have introduced LSA to model a single video sequence for enhanced navigation. This article extends the previous work to video database modeling for high-level concept detection and present a method to improve the robustness of the system.

The next section introduces the Latent Semantic Indexing conjointly with methods to improve performances, i.e. combination of color and texture information and better robustness. Then, K-Nearest Neighbors and Gaussian

Mixture Model classifiers are presented. Next, their performance and the efficiency of LSI are discussed through experimental results. Finally, we conclude with future works.

2 Video Content Modeling

In order to efficiently describe the video content, we decided to borrow a well-known method used for text document analysis named Latent Semantic Analysis. First we detail the adaptation of LSI to our situation and then propose methods to include multiple features and to improve the robustness of LSI in our particular case.

Latent Semantic Analysis (LSA) is a theory and method for extracting and representing the contextual meaning of words by statistical computations applied to a large corpus of text. The underlying idea is that the aggregate of all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and sets of words to each other [4]. In practical, we construct the occurrence matrix A of words into documents. The singular value decomposition over A gives transformation parameters to a singular space where documents can efficiently be compared.

For video content analysis, a corpus does not naturally exist, however one can be obtained thanks to vector quantification technics. In [6], we have presented an approach on single video sequences that relies on k-means clustering to create a corpus of frame-regions. Basically, key-frames are segmented into regions and each region is represented by a set of features. They are then mapped into a codebook, obtained with the k-means algorithm, to construct the co-occurrence matrix A of codebook elements in video key-frames. Thus each frame is represented by the occurrence of codebook terms. LSI is then applied to the matrix A and provides projection parameters into the singular space where frame vectors are projected to be indexed and compared. This can be extended to model a set of video sequences; the set can be seen as a unique video where key-frames are the representative frames of shots. However the total number of key-frames is huge and the computation of the SVD is almost intractable. To overcome this, we subsample the stream and keep only 5000 key frames.

Mathematical operations are finally conducted in the following manner:

- First a codebook of frame-regions is created on a set of training videos,
- The co-occurrence matrix is constructed:
  Let A of size M by N be the co-occurrence matrix of M centroids (defining a codebook) into N key-frames (representing the video database). Its value at cell (i, j) corresponds to the number of times the region i appears in the frame j.
- Next, it is analyzed through LSA:
  The SVD decomposition gives $A = USV^t$ where

$$UU^t = VV^t = I,\ L = \min(M,N)$$

$$S \approx diag(\sigma_1,..,\sigma_L),\ \sigma_1 \geq \sigma_2 \geq ... \geq \sigma_L$$

  Then A is approximated by truncating U and V matrices to keep k factors in S corresponding to the highest singular values.

$$\hat{A} = U_k S_k V_k^t \text{ with } S_k = diag(\sigma_1,..,\sigma_k)$$

- Finally, indexing of a context of A noted $c(j)$ and a new context q is realized as follows:

$$p_{c(j)} = \text{row j of } VS$$

$$p_q = q^t U_k$$

- And to retrieve the context q in a database containing indexed contexts p_j, the cosine measure m_c is used to compare elements.

$$m_c(p_j,q) = \frac{p_q \cdot p_j}{||p_q|| . ||p_j||}$$

  The most similar elements to the query are those with the highest value of m_c.

The number of factors k to keep is crucial and difficult to choose since we do not really want to reduce the dimension for compression purpose but to create induction rules and improve the comparison task. This simplification provides a least squared approximation of the original matrix, therefore it can be seen as a filter that removes the noisy part of the co-occurrence matrix. A threshold has to

be defined to effectively remove noise while keeping the integrity of word equivalences. It is selected by testing several values on test data.

In the particular situation of video content, many features can be extracted. Three methods were evaluated to consider multiple features. They could be combined at the origin, before the creation of the codebook, or independent codebooks could be merged to create a single occurrence matrix, or the LSI is applied to each feature and the similarity measure is modified to combine outputs from each singular space. In [6], we retained that equivalent performances were obtained when features were combined just before or after LSI. The latter solution being the most flexible was kept for our task. Indeed features can easily be weighted and new features added without the need to redo all computation tasks.

However, the LSI does not reveal as performant for many videos as for one. The occurrence information in each frame is to weak compared to the noise inherent to the codebook and this effect is further emphasized when many videos are implied. To come over codebook instability, we match a region to its k-nearest elements in the codebook. This one to many relationship allows to inject more occurrence information for each key-frame and to deal with the sub-optimality of the codebook.

3 Feature Detection

This year two estimators are evaluated. We go on with Gaussian Mixture Model and introduce K-Nearest Neighbors that does not assume any distribution shape of the data. Input features are the projected vectors of color and texture features in their respective singular space where they are normalized.

Let assume that the distribution of each feature can be modeled by a mixture of Gaussians. The Mahalanobis distance remains valid to compare projected vectors since they are normalized. The classical Expectation-Maximization algorithm trains mixture of ten Gaussians assumed to have a diagonal covariance. The final ranking of shots containing a feature, denoted F_x, is then based on the likelihood value computed on the corresponding mixture. An other solution consists on training two mixtures for each feature, one for positive samples (P_p), i.e. that contains F_x and one for negative samples (P_n), i.e. that does not contain F_x. We compute a detection score as:

$$Ds(shot_i) = P_p(shot_i)/[P_p(shot_i) + p_n(shot_i)]$$

.

Since we have no information about the distribution shape of the data, we find natural to compare the performance of GMM with K-NN classifier. Given a shot i, its 20 nearest neighbors in the training set are identified. Then for each feature, it inherits the detection score as follows:

$$Ds(shot_i) = \sum_{k=0}^{k=20} sim(shot_i, shot_k) * Ds(trshot_k)$$

. Detection scores for training shots, $trshot_k$, are either 1 if F_x was annotated or 0 if not.

4 Experiments

Our final participation to Video-TREC includes 6 runs to compare both classifiers and the effect of LSI over raw features.

5 Conclusion

We have presented Latent Semantic Indexing to efficiently model video contents. It gives an efficient representation of key-frame (thus shot) content. However the proposed adaptation relies on the creation of a codebook, operation that is often sub-optimal. To overcome this problem, we have introduced a method that improves noise robustness by matching a frame-region to k codebook elements. We then used these features to train two classifiers: Gaussian Mixture Model and K-Nearest Neighbors, the first models semantic classes with mixture of Gaussians whereas the second makes any assumption about feature distribution in classes. Finally classifiers were compared and used to evaluate the gain obtained with LSI.

Future works will take several directions. One disadvantage of Latent Semantic Indexing, as presented, is the lost of spatial information. Thus, efforts will be conducted to include spatial relationship between regions. On the other hand, we do not take advantage of the whole video content. New features will be included such as object and camera motion, text and audio. Moreover a shot can be represented by all its frame instead of only its key-frame.

References

[1] Shih-Fu Chang, W. Chen, H.J. Meng, H. Sundaram, and Di Zhong. A fully automated content-based video search engine supporting spatiotemporal queries. In *IEEE Transactions on Circuits and Systems for Video Technology*, volume 8, pages 602– 615, 1998.

[2] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[3] Mikko Kurimo. Indexing audio documents by using latent semantic analysis and som. In Erkki Oja and Samuel Kaski, editors, *Kohonen Maps*, pages 363–374. Elsevier, 1999.

[4] Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.

[5] M.R. Naphade, T. Kristjansson, B. Frey, and T.S. Huang. Probabilistic multimedia objects (multijects): a novel approach to video indexing and retrieval. In *IEEE International Conference on Image Processing*, volume 3, pages 536–540, 1998.

[6] Fabrice Souvannavong, Bernard Merialdo, and Benoît Huet. Video content modeling with latent semantic analysis. In *Third International Workshop on Content-Based Multimedia Indexing*, 2003.

[7] Rong Zhao and William I Grosky. From features to semantics: Some preliminary results. In *International Conference on Multimedia and Expo*, 2000.