ROBUST SCENE CUT DETECTION BY SUPERVISED LEARNING

G. Cámara Chávez[1,2]*, M. Cord*[1]*, S. Philipp-Foliguet*[1]*, F. Precioso*[1]*, Arnaldo de A. Araújo*[2]

[1]Equipe Traitment des Images et du Signal-ENSEA
6 avenue du Ponceau 95014 Cergy-Pontoise - France
[2]Federal University of Minas Gerais
Computer Science Department
Av. Antônio Carlos 6627 31270-010 - MG - Brazil

ABSTRACT

The first step for video-content analysis, content-based video browsing and retrieval is the partitioning of a video sequence into shots. A shot is the fundamental unit of a video, it captures a continuous action from a single camera and represents a spatio-temporally coherent sequence of frames. Thus, shots are considered as the primitives for higher level content analysis, indexing and classification. Although many video shot boundary detection algorithms have been proposed in the literature, in most approaches, several parameters and thresholds have to be set in order to achieve good results. In this paper, we present a robust learning detector of sharp cuts without any threshold to set nor any pre-processing step to compensate motion or post-processing filtering to eliminate false detected transitions. The experiments, following strictly the TRECVID 2002 competition protocol, provide very good results dealing with a large amount of features thanks to our kernel-based SVM classifier method.

1. INTRODUCTION

The development of shot boundary detection algorithms was initiated some decades ago with the intention of detecting sharp cuts in video sequences. A vast majority of all works published in the area of content-based video analysis and retrieval are related in one way or another with the problem of shot boundary detection. Indeed, solving the problem of shot boundary detection is one of the principal prerequisites for revealing video content structure in a higher level.

A common approach to detect shot boundaries is computing the difference between two adjacent frames (color, motion, edge and/or texture features) and compare this difference to a preset threshold (threshold-based approach). Del Bimbo [1], Brunelli et al. [2], Lienhart [3] collect extensive reviews of this set of techniques. The main drawback of these approaches lies in detecting different kind of transitions with a unique threshold. To cope with this problem, video shot segmentation can be seen, from a different perspective, as a categorization task. There have only been a few machine learning approaches proposed to overcome this problem. Boreczky et al. [4] apply HMMs with separate states to model shot cuts, fades, dissolves, pans and zooms. Gunsel et al. [5] consider temporal video segmentation as a 2-class clustering problem (“scene change” and “no scene change”) and use K-means to cluster frame differences. Different supervised approaches were proposed by [6], [7] and [8]. Recently Ewerth et al. proposed an unsupervised approach [9].

The work presented in this paper focuses on the exploitation of features based on frame differences (histograms, projection histogram, Fourier-Mellin moments and phase correlation method). After the feature extraction step, these features are classified by *Support Vector Machines* (introduced as a machine learning method by Cortes and Vapnik [10]). Furthermore, SVM have been successfully applied in many real world problems and in several areas: text categorization [11], handwritten digit recognition [12] and object recognition [13], etc.

Most of previous works consider a low number of features because of computational and classifier limitations. Then to compensate this reduced amount of information, they need pre-processing steps, like motion compensation. Our kernel-based SVM approach can efficiently deal with a large number of features in order to get a robust classification: better handle of illumination changes and fast move problems, without any pre-processing step.

This paper is organized as follows. In section 2, we present the machine learning approach used in this work. In section 3, we detail the visual features used for classification. We evaluate the similarity measures applied for matching visual information, in section 4. In section 5, we describe our kernel-based SVM classifier. In section 6, we present the results of the proposed method. In section 7, we conclude and we present future work.

2. MACHINE LEARNING APPROACH

Statistical learning approaches have been recently introduced in multimedia information retrieval context and have been very successful [14]. For instance, discrimination methods (from statistical learning) may significantly improve the effectiveness of visual information retrieval tasks.

The system that we propose in this paper deals with a statistical learning approach for video cut detection. However, our classification framework is specific. Figure 1 shows the steps of the approach. First, the feature extraction process captures different information of each frame. We extract, for every, frame in the video stream a feature vector, then a pairwise similarity measure is calculated. We test different distance metrics: $L1$ norm, cosine similarity, histogram intersection and χ^2 distance (see Sec. 4 for more details). Then, each dissimilarity feature vector (distance for each type of feature: color histogram, moments and projection histograms) is used as an input in the classifier. As soon as we use a lot of features, the dimension of the input classification space is high.

With vectors of high dimensionality, artifacts appear, known as the result of the curse of dimensionality [15]. However, with the theory of kernel functions, one can reduce this curse [16], especially if one can build a kernel function for a specific application.

Using a kernel function leads to a set of classification methods. For Pattern Recognition, statistical learning techniques such as nearest neighbors [15], support vector machines, bayes classifiers have been used. We have previously shown that the SVM classification method is highly adapted to the multimedia retrieval context [17]. Thus, we use SVM as classification method. The decision function (previously trained using a data set selected for that purpose) provides as a result the binary labels, i.e., if the frame is detected as a "cut" or "non cut".

The advantage of this approach is that all the thresholds are tuned by the classifier. Thus, the number of features do not represent an issue. Another advantage of the approach is that with many features it is possible to better describe the information content in the frame and avoid the pre-processing step. The choice of SVM as a classifier is due to the well known performance in statistical learning information retrieval.

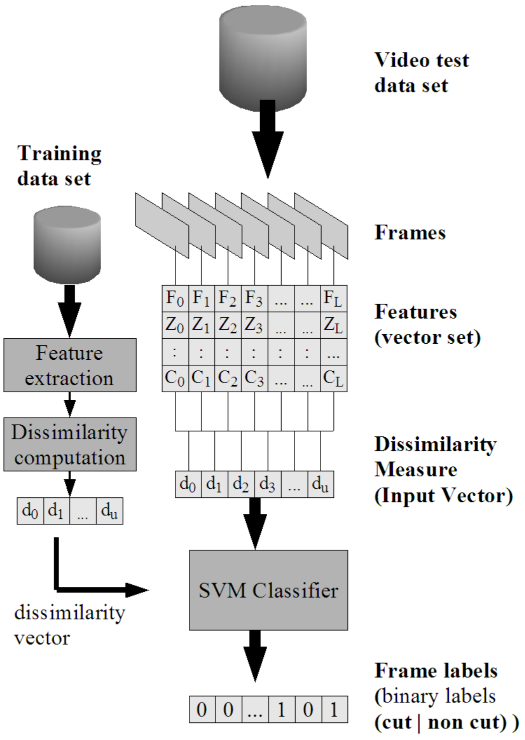


Figure 1: **Learning-based Approach for video cut detection**. Feature vectors $F_t, Z_t, \ldots C_t$ represent Fourier Mellin moments, Zernike moments, Color histogram, from frame f_t. The other features are detailed in Section 3. $d_t = D(f_t, f_{t+1})$ is the similarity distance for each feature where D is one of the similarity measure detailed in Section 4. The SVM classifier is detailed in Section 5.

3. VISUAL FEATURES

Cuts generally correspond to an abrupt change between two consecutive images in the sequence. Automatic detection is based on the information extracted from the shots (brightness, color distribution, motion, edges, etc.). Cut detection between shots with little motion and constant illumination, is usually done by looking for sharp brightness changes. However, brightness changes cannot be easily related to transition between two sots, in the presence of continuous object motion, or camera movements, or change of illumination. Thus, we need to combine different and more complex visual features to avoid such problems. In the next subsections we will review the main visual features used for shot boundary detection.

3.1 Color Histogram

Let $I(x,y)$ be a color image of size $m \times n$, which consists of three channels $I = (I_1, I_2, I_3)$, the color histogram used here is:

$$h_c(b) = \frac{1}{m\times n}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1}\begin{cases} 1 & \text{if } I(x,y) \text{ in bin } b \\ 0 & \text{otherwise}\end{cases} \qquad (1)$$

The color spaces used in this work are the RGB, HSV and opponent color (brightness-independent chromaticities space). In the case of RGB and HSV we consider 2 bins per channel.

The opponent color representation of RGB color space is defined as: $(R+G+B, R-G, B-R-G)$. By choosing this color space, the proposed cut detection algorithm is less sensitive to lighting changes. The advantage of this representation is that the last two chromaticity axes are invariant to changes in illumination intensity and shadows.

These features are stored in vectors denoted RGBh, HSVh, R-Gh,...

3.2 Shape descriptors

As shape descriptor we use ortogonal moments like Zernike moments [18] and Fourier-Mellin moments [19].

3.2.1 Zernike moments

The Zernike moment, of order pq, is defined as :

$$Z_{pq} = \frac{p+1}{\pi}\int_0^{2\pi}\int_0^1 I(\rho,\theta)V^*_{pq}(\rho,\theta)\rho d\rho d\theta \qquad (2)$$

where $p = 0, 1, 2, \ldots, \infty$ defines de order, $I(\rho,\theta)$ is the image in polar coordinates (ρ,θ), while q is an integer depicting the angular dependence, or rotation. The Zernike polynomial V_{pq} is a set of complex polynomials which form a complete orthogonal basis set defined on the unit circle and $\{\}^*$ denotes the conjugate in complex domain [18, 20]. Moments of order 5 ($p = 5$, $p - |q| =$ even and $|q| \leq p$) are computed for each frame, and arranged in a vector denoted Z_t.

3.2.2 Fourier-Mellin moments

U_{pq} is the ortogonal Fourier-Mellin function of order p, q (uniformly distribute over the unit circle) defined as:

$$U_{pq}(\rho,\theta) = Q_p(\rho)e^{-jq\theta}, \qquad (3)$$

and the orthogonal Fourier-Mellin moments F_{pq} are defined as:

$$F_{pq} = \frac{p+1}{\pi}\int_0^{2\pi}\int_0^1 I(\rho,\theta)U_{pq}(\rho,\theta)\rho d\rho d\theta \qquad (4)$$

where $I(\rho,\theta)$ is the image in polar coordinates (ρ,θ), $q = 0, \pm 1, \pm 2, \ldots$ is the circular harmonic order, the order of the

Mellin radial transform is an integer p with $p \geq 0$. For a given degree p and circular harmonic order q, $Q_p(\rho) = 0$ has p zeros. The number of zeros in a radial polynomial corresponds to the capacity of the polynomials to describe high frequency components of the image. Therefore, for representing an image over the same level of quality, the order of p ortogonal Fourier-Mellin is always less than the order of other moments [19]. Moments of order 4 ($p = 4$ and $|q| \leq p$) are computed for each frame, all of them arranged in a vector denoted F_t.

3.3 Projection histograms

Projection is defined as an operation that maps a image into a one-dimensional array called projection histogram [21]. Two types of projection (vertical and horizontal):

$$\begin{aligned} M_h(y) &= \frac{1}{x_2 - x_1} \int_{x_1}^{x_2} I(x,y)dx & (5) \\ M_v(x) &= \frac{1}{y_2 - y_1} \int_{y_1}^{y_2} I(x,y)dy & (6) \end{aligned}$$

These features are stored in vectors denoted V_h and H_h

3.4 Phase Correlation Method (PCM)

The phase-correlation method [22] measures the motion directly from the phase correlation map (shift in the spatial domain is reflected as a phase change in the spectrum domain). This method is based on block matching: each block r in frame f_t is sought the best match in the neighbourhood around the corresponding block in frame f_{t+1}. In this work a block size of 32 × 32 was chosen. The PCM for one block is defined as:

$$\rho(r_t) = \frac{FT^{-1}\{\widehat{r_t}(\omega)\widehat{r_{t+1}}^*(\omega)\}}{\sqrt{\int |\widehat{r_t}(\omega)|^2 d\omega \int |\widehat{r_{t+1}}(\omega)|^2 d\omega}} \qquad (7)$$

where ρ is the spatial coordinate vector and ω is the spatial frequency coordinate vector, $\widehat{r_t}(\omega)$ denote the Fourier transform of block r_t, FT^{-1} denotes the inverse Fourier transform and $\{\}^*$ is the complex conjugate.

By applying a high-pass filter and performing normalised correlation this method is robust to global illumination changes [23]. We use the entropy E_r of the block r as the *goodness-of-fit* measure for each block.

The similarity metric M_t is defined by the median of all block entropies instead of the mean to prevent outliers [23].

$$M_t = \text{median}(E_r) \qquad (8)$$

4. SIMILARITY MEASURES

This section describes the similarity measures used for matching visual information. The similarity is determined as a distance between 2 extracted vectors representing one feature (for example Zernike moments: Z_t) or concatenation of several features (for example Zernike moments and color histograms: $\{Z_t, HSVh\}$). Feature vectors are considered as histograms in terms of similarity measure and thus denoted with the generic name H_t.

The distance usually used is a L_1 norm between feature vectors H_t and H_{t+1}:

$$d_t = D(f_t, f_{t+1}) = \sum_{j=0}^{u} |H_t(j) - H_{t+1}(j)| \qquad (9)$$

where $H_t(j)$ is $j-$th bin of the vector of the $t-$th frame.
The cosine dissimilarity [24] between two vectors is defined as:

$$d_t = D(f_t, f_{t+1}) = \frac{\sum_{j=0}^{u} (H_t(j) \times H_{t+1}(j))}{\sqrt{\sum_{j=0}^{u} H_t(j)} \times \sqrt{\sum_{j=0}^{u} H_{t+1}(j)}} \qquad (10)$$

Histogram intersection is defined as:

$$d_t = D(f_t, f_{t+1}) = 1 - \frac{\sum_{j=0}^{u} min(H_t(j), H_{t+1}(j))}{\sum_{j=0}^{u} H_t(j)} \qquad (11)$$

Another dissimilarity metric is χ^2:

$$d_t = D(f_t, f_{t+1}) = \sum_{j=0}^{u} \frac{(H_t(j) - H_{t+1}(j))^2}{H_t(j) + H_{t+1}(j)} \qquad (12)$$

5. SUPPORT VECTOR MACHINES

There are some learning approaches that use SVM as classifier. More recently, Qi et al. [6] transform the temporal segmentation into a multi-class categorization. For the classification task they compare different binary classifiers: $k-$nearest-neighbor classifier (KNN), the Naïve Bayes probabilistic classification, and SVM. Since its creation in 2001 TRECVID [1] has become the reference framework to propose and compare new approaches. IBM system [8] consists of extraction modules for local and global visual features. The algorithm is based on a finite state machine and the features are classified by a SVM. R. Ewerth and B. Freisleben [9] propose an unsupervised learning approach based on a sliding estimation window and an adequate measure of clustering quality. Adcock et al. [7] present an approach combining pairwise similarity and supervised classification, they used a KNN. Regarding the increase of classification methods proposed for TRECVID and the quality of their results, these approaches appear promising for the task of shot boundary detection. Based on these successful experiences we adopt a machine learning approach using a SVM as classifier.

The classification problem can be restricted to a two-class problem. The goal is, then, to separate the two classes with a function induced from available examples. We hope to produce, hence, a classifier that will properly work on unknown examples, i.e. which generalises efficiently the classes defined from the examples.

The SVM have been developed as a robust tool for classification and regression in noisy and complex domains. SVM can be used to extract valuable information from data sets and construct fast classification algorithms for massive data.

Another important characteristic of the SVM classifier is to allow a non-linear classification without requiring explicitly a non-linear algorithm thanks to kernel theory.

In kernel framework data points may be mapped into a higher dimensional feature space, where a separating hyperplane can be found. We can avoid to explicitly compute the

[1] A video retrieval algorithm competition

mapping using the kernel trick which evaluate similarities between data $K(d_t, d_s)$ in the input space.

Common kernel functions are: linear, polynomial, gaussian radial basis, gaussian with χ^2 distance (Gauss-χ^2) $K(d_t, d_s) = e^{-\chi^2(d_t,d_s)/2\sigma^2}$ and

Triangular kernel:

$$K(d_t, d_s) = \begin{cases} 1 - |u| & \text{if} \quad -1 < u < 1 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

here $u = \frac{d_t - d_s}{h}$, where w is the window width. Each kernel function results in a different type of decision boundary.

Our kernel-based SVM approach can thus efficiently deal with a large number of features in order to get a robust classification.

6. RESULTS

The data set used in our experiments is TRECVID-2002 Video Data Set. The shot boundary test collection contains 4 hours and 51 minutes of video. The video are mostly of a documentary/educational nature, but very varied in age, production style, and quality. At a total, there were 18 videos in MPEG-1 with a total size of 2.88 gigabytes. For all videos, shot segmentation reference data had been manually constructed by NIST.

We strictly follow the TRECVID-2002 protocol in our tests. We run our algorithm on all the TRECVID test set and provide the mean precision and the mean recall obtained. We can provide up to 10 different runs (10 different choices of parameters, features or kernels).

Precision: Among the transitions detected by the system, count the true transitions
Recall: For all possible transitions, count the detected transitions

A good detector should have high precision and high recall. $F1$ is a commonly used metric that combines precision and recall values. If both values are high then $F1$ is high.

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

Test set can only be used for test purpose, it is forbidden to use any video of the test set as a training data (Development data). Therefore our training set consists on a single video of 4197 frames (2mins. 20 secs.) with 50 cuts. This video is captured from a TV-station and is composed by a segment of commercials. We use a SVM classifier and train it with different kernels: linear, polynomial, gaussian with L2 and χ^2 distance, and triangular.

We conducted numerous experiments that provide interesting and meaningful contrast. Table. 1 shows the recall/precision measure for the best similarity measure and each kernel function. The Gaussian-χ^2 kernel provides the best results over all the other kernel functions. Furthermore, the combination *kernel/similarity distance* that performs the best result is the combination of kernel function Gaussian-χ^2 and the similarity measure χ^2 distance. Thus, our evaluation of kernel functions confirms that when distributions are used as feature vectors, a Gaussian kernel gives excellent results in comparison to distance-based techniques [17].

Kernel / Similarity measure	Recall	Precision	F1
Linear / Cos. sim.	0.92	0.88	0.90
Polynomial 3 / $L1$ norm	0.92	0.90	0.91
Gauss-$L2$ / Hist. inters.	0.92	0.89	0.90
Gauss-χ^2 / χ^2 dist.	0.94	0.90	0.92
Triangle / Cos. sim.	0.92	0.90	0.91

Table 1: Performance measure for each kernel function

Run	Features	Sim. measure
1	Ph, HSVh, Zer, Hor, Var	$L1$ norm
2	Ph, HSVh, Ver, Hor, Var	Cos. sim.
3	Ph, HSVh, Ch, Fou, Zer, Var	Cos. sim.
4	Ph, Ch, Zer, Ver, Hor, Var	Cos. sim.
5	Ph, R-Gh, HSVh, Ch, Fou, Hor, Var	Cos. sim.
6	Ph, HSVh, Ch, Fou, Zer, Hor, Var	Cos. sim.
7	Ph, Ch, Fou, Zer, Ver, Hor, Var	Hist. Int.
8	Ph, HSVh, Zer, Ver, Hor, Var	Hist. Int.
9	Ph, R-Gh, HSVh, Ch, Fou, Zer, Hor, Var	Hist. Int.
10	Ph, HSVh, Ch, Fou, Zer, Hor, Ver, Var	χ^2

Table 2: 10 best combinations of visual features

Following the TRECVID protocol, ten runs were performed. The nomenclature used for the features is as follows: RGB color histogram (Ch), HSV color histogram (HSVh), opponent color histogram (R-Gh), Zernike moments (Zer), Fourier-Mellin moments (Fou), Horizontal project histogram (Hor), Vertical projection histogram (Ver), Phase correlation (Ph) and Variance (Var). The best choice for kernel selection is the Gaussian-χ^2, thus all runs were executed using this kernel. In Table 2, we present the visual feature vectors and the corresponding similarity distances to the 10 best results.

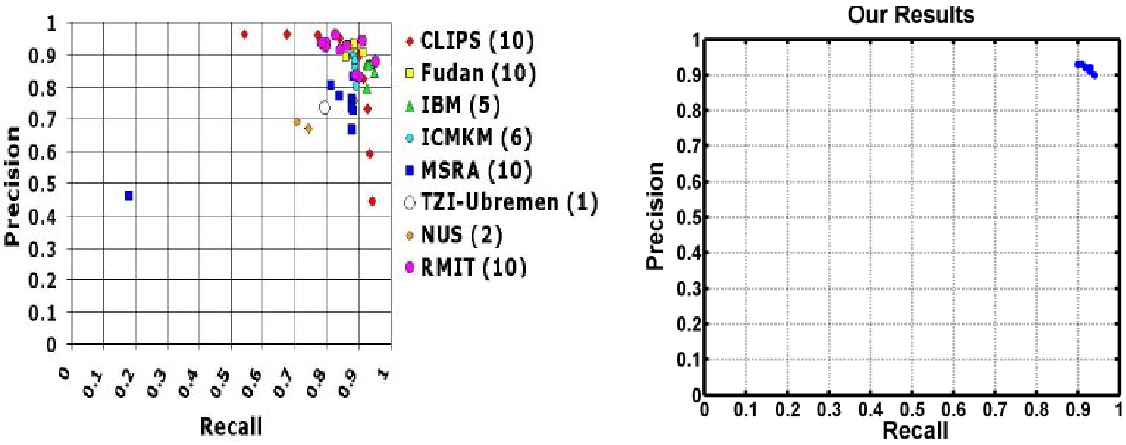
In Figure 2(a) we show the results that were obtained in the official contest of TRECVID-2002 and compare it with the results of our ten runs Figure 2(b). As shown in the figure the accuracy and robustness of our approach is very efficient. Hence, the capacity of generalisation of our classifier is proven and the combination of the selected features performs good results without any pre-processing or post-processing.

7. CONCLUSION AND FUTURE WORKS

This paper considers cut detection from a supervised classification perspective. Previous detecting cut classification approaches consider few visual features because of computational limitations. As a consequence of this lack of visual information, these methods need pre-processing and post-processing steps, in order to simplify the detection in case of illumination changes, fast moving objects or camera motion.

We propose to implement a kernel-based SVM classifier which can deal with large feature vectors. Hence, we combine a large number of visual features and avoid any pre-processing or post-precessing step. We present a supervised statistical learning approach, requiring a small training set. Thus, we do not have to set any threshold as many methods proposed in the framework of TRECVID.

We compare our algorithm to the latest results publically available. Our method shows excellent performance on the 2002 TREC Video Track Shot Classification Task in terms of



(a) (b)

Figure 2: Precion/Recall measure of performance. (a) show the official results for TRECVID 2002 [25], (b) show our ten runs results

precision and recall.

To confirm the efficiency of our approach, we are going to participate to the TRECVID-2006 competition. The next step is to extend our algorithm for gradual transition detection. For that purpose new features will be necessary. This will not be an issue for our kernel-based algorithm which can deal with high order features. We expect our learning-based approach be able to detect cuts and gradual transitions.

8. ACKNOWLEDGMENTS

The authors are grateful to MUSCLE Network of Excellence, CNPq and CAPES for the financial support of this work.

REFERENCES

[1] A. del Bimbo, *Visual Information Retrieval*, Morgan Kaufmann, San Francisco, California, 1999.

[2] R. Brunelli, O. Mich, and C.M. Modena, "A survey on the automatic indexing of video data," *Journal of Visual Communication and Image Representation*, vol. 10, pp. 78–112, 1999.

[3] R. Lienhart, C. Kuhmunch, and W. Effelsberg, "On the detection and recognition of television commercials," *IEEE International Conference on Multimedia Computing and Systems '97. Proceedings*, pp. 509 – 516, 1997.

[4] J.S. Boreczky and L.D. Wilcox, "A hidden markov model framework for video segmentation using audio and image features," in *ICASSP'98*, 1998, vol. 6, pp. 3741–3744.

[5] B. Gunsel, A. Fernan, and A. Tekalp, "Temporal video segmentation using unsupervised clustering and semantic object tracking," *Journal of Electronic Imaging*, pp. 592–604, 1998.

[6] Y. Qi, T. Liu, and A. Hauptmann, "Supervised classification of video shot segmentation," in *IEEE Conference on Multimedia & Expo (ICME'03)*, Baltimore, MD, July 6-9 2003.

[7] John Adcock, Andreas Gingensohn, Matthew Cooper, Ting Liu, Lynn Wilcox, and Eleanor Rieffel, "Fxpal experiments for trecvid 2004," in *TREC Video Retrieval Evaluation Online Proceedings: TRECVID 2004*, 2004.

[8] Arnon Amir, Janne Argillander, Marco Berg, Shih-Fu Chang, Martin Franz, Winston Hsu, G. Uyengar, J. Kender, L. Kennedy, C. Lin, M. Naphade, A. Natsev, J. Smith, J. Tesic, G. Wu, R. Yan, and D. Zhang, "Ibm research trecvid-2004 video retrieval system," in *TREC Video Retrieval Evaluation Online Proceedings: TRECVID 2004*, 2004.

[9] Ralph Ewerth and Bernd Freisleben, "Video cut detection without thresholds," in *Proc. of 11th Workshop on Signals, Systems and Image Processing*, Poznan, Poland, 2004, pp. 227–230, PTETiS.

[10] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[11] T. Joachims, "Text categorization witt support vector machines," in *Proceedings of the European Conference on Machine Learning*, 1998.

[12] V. Vapnik, *Statistical Learning Theory*, Wiley, 1998.

[13] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Proceedings of the International Conference on Computer Vision*, 1998.

[14] Simon Tong, *Active Learning: Theory and Applications*, Ph.D. thesis, Stanford University, 2001.

[15] T. Hastie, R. Tibshirani, and J. Friedman, *The Element of Statistical Learning*, Springer, 2001.

[16] A. Smola and B. Scholkopf], *Learning with kernels*, MIT Press, Cambridge, MA., 2002.

[17] P.H. Gosselin and M. Cord, "A comparison of active classification methods for content- based image retrieval," in *International Workshop on Computer Vision meets Databases (CVDB), ACM Sigmod*, Paris, France, June 2004, pp. 51–58.

[18] C. Kan and M.D. Srinath, "Combined features of cubic b-spline wavelet moments and zernike moments for invariant pattern recognition," in *International Conference on Information Technology: Coding and Computing.*, 2001, pp. 511–515.

[19] C. Kan and M.D. Srinath, "Invariant character recognition with zernike and orthogonal fourier-mellin moments," *Pattern Recogntion*, vol. 35, pp. 143–154, 2002.

[20] K. Whoi-Yul and K. Yong-Sung, "A region-based shape descriptor using zernike moments," *Image Communication*, vol. 16, no. 95-102, 2000.

[21] O.D. Trier, A.K. Jain, and T. Taxt, "Feature extraction methods for character recognition - a survey," *Pattern Recognition*, vol. 29, pp. 641–662, 1996.

[22] James Ze Wang, "Methodological review - wavelets and imaging informatics : A review of the literature," *Journal of Biomedical Informatics*, pp. 129–141, July 2001, Avaliable on `http://www.idealibrary.com`.

[23] S. V. Porter, M. Mirmehdi, and B. T. Thomas, "Temporal video segmentation and classification of edit effects," *Image and Vision Computing*, vol. 21, no. 13-14, pp. 1097–1106, December 2003.

[24] G. Salton, *Automatic Text Processing*, Addison-Wesley Longman Publishing, 1989, Chapter 9.

[25] A.F. Smeaton and P. Over, "The trec-2002 video track report," in *The Eleventh Text REtrieval Conference (TREC 2002)*, 2002, `http://trec.nist.gov//pubs/trec11/papers/VIDEO.OVER.pdf`.