PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Categorizing video shots by utilizing SVM and wavelet

Jiang, Haina, Xia, Xiquan

Haina Jiang, Xiquan Xia, "Categorizing video shots by utilizing SVM and wavelet," Proc. SPIE 8335, 2012 International Workshop on Image Processing and Optical Engineering, 83351A (15 November 2011); doi: 10.1117/12.917663

Event: 2012 International Workshop on Image Processing and Optical Engineering, 2012, Harbin, China

Categorizing video shots by utilizing SVM and wavelet

Haina JIANG*[a], Xiquan XIA[a]

[a] Chongqing College of Electronic Engineering, Chongqing, 401331, China

ABSTRACT

Shots classification plays an important role in well indexing, browsing and retrieving video content. By that, the large amount of video content can be efficiently indexed, and then, it can provide convenience for managing video. In this paper, edge features are firstly extracted by wavelet, which can not only reduce amount of shots data but also preserve the important structural properties of shots. And then, to reflect local properties of shots, ratio of edge pixels in each sub-window is calculated. After that, color moments are computed to reduce loss of global properties, which can assist edge features in well indexing shots. Finally, support vector machine (SVM), which has a very good performance on pattern recognition, is employed to classify shots. Experimental results demonstrate that this method can efficiently categorize video shots and satisfy the basic needs of shots classification.

Keywords: shots, classification, wavelet, edge, sub-window, color moments, support vector machine, decision function

1. INTRODUCTION

To help users identify shots with similar semantics to quickly browse and retrieve the relevant clips, it is necessary to study an efficient way to video shots classification. However, the video is impressive for its visual image and amount of information which makes users difficult to obtain their purpose shots in short time. Considering above, a method of shots classification is presented, which focuses on the segmented shots.

To classify video shots, various features of shots are utilized to index shots, e.g. color, texture, structure and so on. For video classification, some use one feature, e.g. Tien et.al [1] chooses the number of dominant color pixels of each frame to classify shots. Yuan [2] et.al analyzes global motions and local motions of video to distinguish shots. To improve effect of classification, more than one feature is extracted. e.g. Pallavi[3] et.al extracts candidate ball positions using features based on shape and size and identify a ball by filtering the candidates with the help of motion information for medium shots. Zhao [4] et.al adopts text and motion feature to represent text and caption in videos. In general, results produced by these methods are impressive. However, owning to influenced by different method of feature extraction and selection of classifiers, the classification method need to be further improved. For example, [5,6] employs canny operator to extract edge features to index shots and then classify them. For these methods, it is advisable to extract edges to index video shots because edges are expressions of discontinuity of local characteristics (texture, gray and structure, etc.) and contain rich information and fundamental characteristics of image. However, methods of differential operator are vulnerable to the influence of template size and noise. As for the final categorization procedure, a suitable classifier is also crucial. Take into consideration above, a new way of video classification is presented.

The rest of this paper is organized as follows: in the section 2, the methodology about feature extraction is first introduced. And then briefly describes SVM classifier and corresponding classification process. In section 3, some experiments are conducted to verify this method. In section 4, conclusions are made.

2012 International Workshop on Image Processing and Optical Engineering, edited by Hai Guo, Qun Ding,
Proc. of SPIE Vol. 8335, 83351A · © 2012 SPIE · CCC code: 0277-786X/12/$18 · doi: 10.1117/12.917663

2. METHODOLOGY

To categorize video shots, feature extraction is one of key procedures. Here, we will first review process of edge and color extraction, and then SVM classifier.

2.1 Edge extracted by wavelet [7]

The two-dimensional function is smooth if its double integral is nonzero. By calculating the partial derivatives along x and y of a smoothing function $\theta(x,y)$, we can define two wavelets:

$$\psi^1(x,y)=\frac{\partial\theta(x,y)}{\partial x} \qquad (1)$$

$$\psi^2(x,y)=\frac{\partial\theta(x,y)}{\partial y} \qquad (2)$$

Given a scale s, let:

$$\psi_s{}^1(x,y)=\frac{1}{s^2}\psi^1(\frac{x}{s},\frac{y}{s}) \qquad (3)$$

$$\psi_s{}^2(x,y)=\frac{1}{s^2}\psi^2(\frac{x}{s},\frac{y}{s}) \qquad (4)$$

Then for any function $f(x,y)\in L^2(R^2)$, the wavelet transform defined with respect to $\psi^1(x,y)$ and $\psi^2(x,y)$ has two components:

$$W^1 f(s,x,y)=f*\psi_s{}^1(x,y) \qquad (5)$$

$$W^2 f(s,x,y)=f*\psi_s{}^2(x,y) \qquad (6)$$

Where, '*' stands for two-dimensional convolution, $f(x,y)$ is the target image. The two components can be written as:

$$\begin{pmatrix} W^1 f(s,x,y) \\ W^2 f(s,x,y) \end{pmatrix}=s\begin{pmatrix} \frac{\partial}{\partial x}(f*\theta_s)(x,y) \\ \frac{\partial}{\partial y}(f*\theta_s)(x,y) \end{pmatrix}=s\vec{\nabla}(f*\theta_s)(x,y) \qquad (7)$$

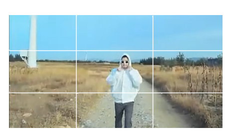
Hence, the two components of the wavelet transform are proportional to the coordinates of the gradient vector of $f(x,y)$ smoothed by $\theta_s(x,y)$ .Similar to calculation of gradient modulus, the wavelet transform modulus Mod can be computed by:

$$Mod=\sqrt{(W^1 f(s,x,y))^2+(W^2 f(s,x,y)^2} \qquad (8)$$

Then, by a given threshold T, edge feature of frame can be quickly extracted, and then, edge feature of all shots can be obtained. Here, in order to master local structure of shots, we study the edge features in sub-window instead of the whole frame. We divide each frame into six blocks show in Fig.1:

1	2
3	4
5	6

(a)



(b)

Fig.1 Sketch map of block

In each sub-window, we compute the ratio of edge pixels by:

$$\mathrm{Re}_k=\sum_{i=1}^{6} e_k \Big/ Num_k \qquad (9)$$

Where, $k(k=1,2\ldots 6)$ is the order number of block, e_k is number of edge pixel in k-th block, Num_k is number of pixel in k-th block. Then edge vector E_{i} of the i- th frame can be described by:

$$E_{\mathrm{i}} = [Re_1, Re_2, Re_3, Re_4, Re_5, Re_6]^T \tag{10}$$

To improve the accuracy, the edge vector will be normalized by:

$$NE_{\mathrm{i}} = [Re_1, Re_2, Re_3, Re_4, Re_5, Re_6]^T \big/ Max(E_{\mathrm{i}}) \tag{11}$$

Where, $Max(E_{\mathrm{i}})$ is the maximum factor of edge feature E_{i}. Then each factor value of NE_{i} is normalized in the range of $[0,1]$. To avoid of losing global property, color moments are calculated to act as supplement of edge vector.

2.2 Color moments [8]

Stricker and Orengo use three central moments to describe an image's color distribution since any color can be characterized by its moments and most information is concentrated on the low-order moments. The three color moments can be defined as:

$$E_i = \frac{1}{N}\sum_{j=1}^{N} p_{ij} \tag{12}$$

$$\sigma_i^{\ 2} = \frac{1}{N}\sum_{j=1}^{N} (p_{ij} - E_i)^2 \tag{13}$$

$$s_i^{\ 3} = \frac{1}{N}\sum_{j=1}^{N} (p_{ij} - E_i)^3 \tag{14}$$

Where, $i \in \{R,G,B\}$, j is order number of frame in given shot, E_i is mean, $\sigma_i^{\ 2}$ is standard deviation, $s_i^{\ 3}$ is skewness. p_{ij} is the color value of the i -th color component of the j-th image pixel and N is the total number of pixels in the frame.

Here, we will restrict ourselves to the RGB scheme, and moments are calculated for R, G and B components respectively. Then, color moments of the i-th frame can be expressed as:

$$C_i = [E_R, E_G, E_B, \sigma_R^{\ 2}, \sigma_G^{\ 2}, \sigma_B^{\ 2}, S_R^{\ 3}, S_G^{\ 3}, S_B^{\ 3}]^T \tag{15}$$

Similar to edge vector, color moments will be normalized by:

$$NC_i = [E_R, E_G, E_B, \sigma_R^{\ 2}, \sigma_G^{\ 2}, \sigma_B^{\ 2}, S_R^{\ 3}, S_G^{\ 3}, S_B^{\ 3}]^T \big/ Max(C_i) \tag{16}$$

Based on descriptions above, the i- th frame of given shot can be represented by the fused feature $V_i = (NE_i, NC_i)^T$, and $dim(V_i) = 15$. Repeating the same procedure, a series of vector which stands for video shots can be computed. Since shot is composed of frames recorded from similar scene and our purpose is to classify shot, it is reasonable to choose one key frame to stand for corresponding shot. Now we take a shot for example to demonstrate the choosing method.

Supposed a shot $s_i = \{f_1, f_2, \ldots, f_L\}$, f_i($i = 1,2,\ldots,L$)is frame of shot s_{i}. According to description above, each frame f_{i} can be represented by a fused vector V_i. And then, s_{i} can be expressed as $s_i = \{V_1, V_2, \ldots, V_L\}$. Next, we calculated mean vector of shot s_{i} according to:

$$\overline{V}=\frac{1}{L}\sum_{i=1}^{L}V_i \qquad (17)$$

By Hausdorff distance [9], the frame closest to $\overline{V}$ is regarded as key frame kf_i of shot s_{i}. And then, by classifying key frames, the corresponding video shots can be categorized.

2.3 SVM classifier [10]

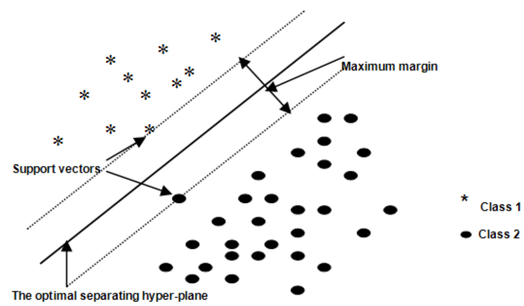


Fig.2 The scheme of SVM (linear separation)

For a supposed classification problem, its training data set is $\{x_i, y_i\}(i=1,2,...,l)$, with the input data $x_i \in \Re^n$, and the corresponding target $y_i \in \{1,-1\}$. The goal of the SVM is to get the hyper-plane that maximizes the minimum distance between any data point, as shown in Fig. 2. In feature space, SVM models take the following form:

$$\mathrm{y}(x)=\omega^T\phi(x)+b \qquad (18)$$

Where, $\phi(\cdot)$ maps the input vector $x_i(i=1,2,...,l)$ into a so-called higher dimensional feature space. b is the bias, and ω is a weight vector of the same dimension as the feature space. This problem can be transformed into a quadratic programming problem:

$$\max_{\alpha}\mathrm{Q}(\alpha)=-\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j \phi(x_i)^T\phi(x_j) \qquad (19)$$

$$s.t.\begin{cases}0\le\alpha_i\le C & \forall i\\ \sum_{i=1}^{l}\alpha_i y_i=0\end{cases} \qquad (20)$$

Where, α is Lagrangian multipliers, C is the trade-off parameter between the error and margin. After getting value of α, solution of ω and b can be gained by:

$$\omega=\sum_{\mathrm{i}=1}^{l}\alpha_i y_i\phi(x_i) \qquad (21)$$

$$\sum_{\mathrm{i}=1}^{l}\alpha_i(y_i(\omega^T\phi(x_i)+b)-1)=0 \qquad (22)$$

Where, x_i is non-zero data, and α_i is support vector. Then the final output hyper-plane decision function of SVM is:

$$F(x)=sign(\sum\alpha_i y_i\phi(x)^T\phi(x_i)+b) \qquad (23)$$

By this decision function, the process of classification can be described as follows:

Step 1: According to description above, we choose a training data set $TD = \{kf_1, kf_2, ..., kf_k\}$, where, K is the kind of video shots. The class label $c = 1$.

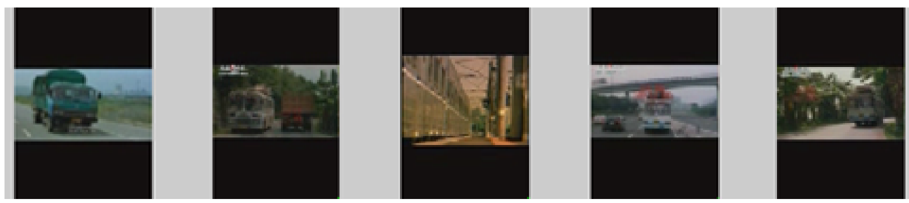
Step 2: Regarding categorize TD as a two-class classification. And then, one is marked as c, the rest ones are considered to be the other class.

Step 3: Recording the decision function f_c.

Step 4: $c = c + 1$, and repeating step 1-step 3 until $c = K$. Then all decision functions can be got. By computing decision functions, all video shots can be classified.

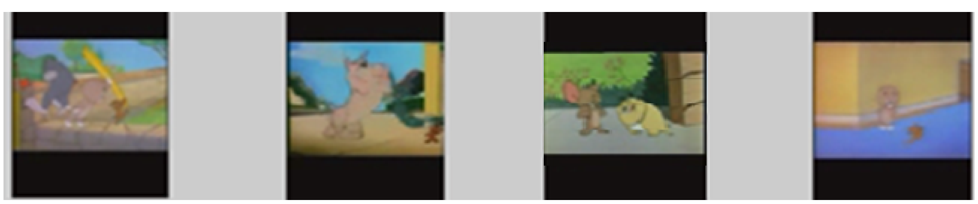
3. RESULTS

The experiment is implemented in Matlab 7.0 on a PC with AMD Sempron(tm) Processor 3100+, 1.81GHz .To demonstrate the efficiency of this classification method, classification experiments are made on a video shots database which contains four classes of shots, which are movie, sports, cartoon and advertisements. According to section 2, part of classification results are shown in Fig.3.

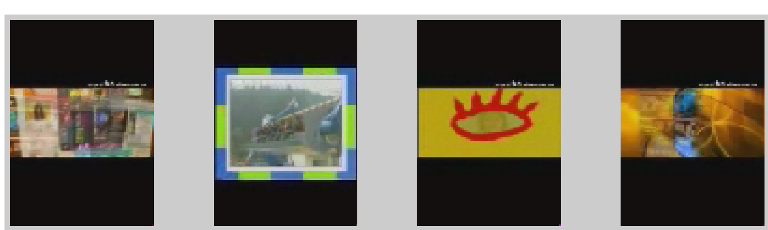


(a) Movie

(b) Sports



(c)Cartoon



(d)Advertisements

Fig. 3 Part of Classification Results

From Fig.3, it can be easily seen that video shots are classified into four classes which is in accordance with the real situation. To verify efficiency of this method in detail, we employ precision and recall[11] to evaluate the classification results:

$$precision = \frac{N_{correct}}{N_{correct} + N_{false}} \tag{24}$$

$$recall = \frac{N_{correct}}{N_{correct} + N_{miss}} \tag{25}$$

Where, for each specified kind of shot, $N_{correct}$ denotes the shot number having been correctly classified, N_{false} is the shot number having been wrongly classified, N_{miss} is the shot number having been missed. The final results are shown in table 1:

Table.1 Experiment Results of Shots Classification

Class	**shots Number**	**Tested number**	**Right number**	**Precision (%)**	**Recall (%)**
movie	210	215	192	89.3	91.4
sports	226	211	185	87.7	81.9
cartoon	320	315	289	91.7	90.3
advertisements	305	310	278	89.7	91.1

From table 1, we can find that both precision and recall of four classes change little, which demonstrate that this method of shot classification has a stable efficiency. This characteristic is significant in daily application of shots classification. What's more, it can be found that precision of this method can keep balance with its recall ratio. However, both precision and recall ratio need to be improved based on the stability of classification.

4. CONCLUSIONS

As one of important information carriers, videos are outstanding for their rich content. To efficiently utilize them, it is necessary to study a reliable method of shots classification to well manage video. In this paper, wavelet is adopted to extract edge features, which not only can rapidly detect edges but also has a certain degree of noise immunity. In this way, structure of shots can be well extracted. Moreover, to describe global property of video, color moments are calculated. By that, video content can be more reliable indexed. Finally, SVM, which is prominent for its good effects on classification for small samples, is employed to categorize video shots. The experimental results demonstrate that this method can better classify video shots and satisfy the basic needs of different scene. However, how to improve its accuracy is our future work.

ACKNOWLEDGMENTS

We would like to thank ICT Research Center, Key Laboratory of Optoelectronic Technology and System of the Education Ministry of China for its experiment equipment.

REFERENCES

[1] Tien, M.C., Chen, H.T., Y Chen,.W., Hsiao, M.H. and Lee, S.Y., "Shot Classification of Basketball Videos and its Application in Shooting Position Extraction,"Proc. ICASSP, 1085-1088 (2007).

[2] Yuan,Z., Wu, Y. and Wang, G.Y., " Motion-information-based video retrieval system using rough pre-classification," Transations on Rough Sets. Papers 4100, 306-333 (2006).

[3] Pallavi,V., Mukherjee, J., Majumdar, A. K. and Sural, S., "Ball detection from broadcast soccer videos using static and dynamic features," Journal of Visual Communication and Image Representation. Papers 19(7), 426-436 (2008).
[4] Zhao,X., Lin, K.H., Fu, Y., Hu, Y.X., Liu, Y.C. and Huang, T.S., "Text from corners: a novel approach to detect text and caption in videos," IEEE Transactions on Image Processing. Papers 20(3), 790-799 (2011).
[5] YU, J.Q. and WANG, N., "Shot Classification for Soccer Video Based on Sub-window Region," Journal of Image and Graphics. Papers 13(7), 1347-1352 (2008).
[6] ZHOU,Y.H., CAO,Y.D., LI, J. and Zhang, H.X., "Soccer Video Shot Classif ication Method Based on Color and Edge Distribution," Transactions of Beijing Institute of Technology. Papers 25(12), 1079-1082 (2005).
[7] Mallat, S. and Huang, W.L., "Singularity detection and processing with wavelets," IEEE Transaction Information Theory. Papers 38 (2), 617-643 (1992).
[8] Stricker, M. and Orengo, M., "Similarity of Color Images," In SPIE Storage and Retrieval for Still Image and Video Databases III 2420, 381-392 (1995).
[9] Huttenlocher, D.P., Klanderman, G.A. and Rucklidge, W.J., "Comparing images using the Hausdorff distance," IEEE Transactions on PAMI. Papers 15(9), 850-863 (1993).
[10] Wang, X.Y., Chen, J.W. and Yang, H.Y., "A new integrated SVM classifiers for relevance feedback content-based image retrieval using EM parameter estimation," Applied Soft Computing. Papers 11, 2787–2804 (2011).
[11] Zhao, S.W., Zhuo, L., Xiao, Z. and Shen, L.S., "A Data-Mining Based Video Shot Classification Method," In Proceedings of the 2009 International Conference on Image and Signal Processing. Proc. ICISP, 1-4 (2009).