A Probabilistic Template-based Approach to Discovering Repetitive Patterns in Broadcast Videos

Peng Wang
Department of Computer Science
and Technology, Tsinghua Univ.
Beijing, 100084, China
wangppeng97@tsinghua.org.cn

Zhi-Qiang Liu
School of Creative Media
City University of Hong Kong
Hong Kong, China
zq.liu@cityu.edu.hk

Shi-Qiang Yang
Department of Computer Science
and Technology, Tsinghua Univ.
Beijing, 100084, China
yangshq@tsinghua.edu.cn

ABSTRACT

There are usually repetitive sub-segments in broadcast videos, which may be associated with high-level concepts or events, e.g., news footage, repeated scores in basketball. Unsupervised mining techniques provide generic solutions to discovering such temporal patterns in various video genres, which are currently the subject of great interests to researchers working on multimedia content analysis. In this paper, we propose a novel approach to automatically detecting repetitive patterns in a video stream. In this approach, a video stream is first transformed to a symbol sequence via the spectral clustering algorithm. After computing the transition probabilities of any two symbols in temporal evolution, we produce a set of probabilistic templates to characterize the patterns of potential interest. Finally, we verify each probabilistic template by measuring the similarities between the video sub-segments and the template. Evaluations on various sports videos show promising results.

Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems – *video*; I.5.2 [**Pattern Recognition**]: Design Methodology – *pattern analysis.*

General Terms

Algorithms, Measurement, Design, Experimentation, Theory

Keywords

Video mining, repetitive pattern discovery, probabilistic template

1. INTRODUCTION

Most broadcast videos, such as sports, news, and movies, have underlying structures of content organization, which may contain repetitive segments in the temporal evolution of a video stream [7]. For instance, the "offense-shoot" segments often repeat throughout basketball. Such repetitive segments, called patterns, are generally associated with high-level semantic concepts or events, and are helpful for video indexing, browsing, and skimming. Most previous research adopted the supervised training-recognition framework to identify repetitive semantic events present in video streams [3]. Although supervised approaches are effective in modeling the relationships between events and features, they are limited to detecting pre-defined events (patterns) in specific domains and their performances heavily rely on the quality of the training data.

To provide more generic and content-adaptive solutions to video analysis, many researchers have recently proposed unsupervised mining techniques [1]. For instance, patterns associated with offenses and defenses in tennis are discovered in an unsupervised manner [6]. Generally, most current approaches to video pattern discovery use two steps: i) decompose a video stream into a discrete symbol sequence, either by clustering [6] or probabilistic models [1]; and ii) enumerate repetitive patterns in the symbol sequence using techniques similar to that for DNA sequence analysis in bioinformatics [2].

However, there are a few major problems in such approaches. First, it is time consuming to exhaustively enumerate patterns [1], as there is no *priori* knowledge about the existing patterns. Second, most techniques in the literature require “identical matching” between any two sequences in the discrete symbol space [6]; however, it would be more effective to consider the similarities between symbols. Unlike DNA sequences where different symbols denote unique nucleotides, in a video stream there are no unique symbols (clusters). Ignoring the symbolic similarities in video mining will lead to the following problems: i) some sequences that do belong to one pattern but described by different symbols may be over-classified into different patterns; and ii) a repetitive pattern in the sequence may not be revealed, if its appearances are slightly different in its symbolic representations.

In an effort to build a more efficient and robust system for repetitive pattern discovery, in this paper, we propose a probabilistic template-based approach. In order to avoid exhaustive enumeration, in this system we first generate a set of probabilistic templates to approximate the patterns of potential interest. Then, instead of identical matching that has to be performed in the discrete symbol space, we measure the similarities between the obtained templates and the video segments *directly* in the continuous low-level feature space. This has an added benefit: the system is more tolerant of classification errors resulted from video decomposition and symbolization. Figure 1 shows that the video stream is first decomposed and symbolized by a clustering-based scheme. Then, a transition matrix is constructed by computing the temporal transitions among various symbols. From this matrix, the system uses the sub-sequences with high occurrence probabilities as the templates for finding the repetitive patterns. Each template is verified throughout the video stream to locate the segments that highly

MM’05, November 6-11, 2005, Singapore.

match with the template. Finally, the system sorts the obtained repetitive segments according to their similarity measures (values) to the template for video indexing and browsing.

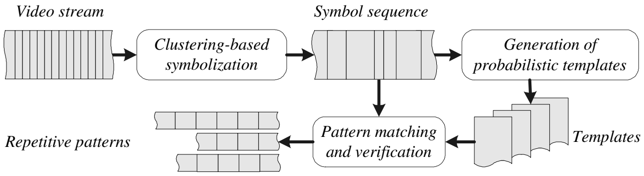
Video stream
Symbol sequence
Clustering-based symbolization
Generation of probabilistic templates
Repetitive patterns
Pattern matching and verification
Templates

Figure 1. System flowchart of the probabilistic template-based approach to pattern discovery in video streams.

In the following section, we will introduce the spectral clustering-based video decomposition and symbolization. Section 3 describes the generation of the probabilistic templates, the pattern matching and verification processes. The experiments and discussions are presented in Section 4. Section 5 concludes this paper.

2. SYMBOLIZATION OF VIDEO STREAM

The decomposition and symbolization of video stream are carried out using the clustering-based scheme; that is, video segments that have similar low-level features are grouped into clusters where each cluster is labeled with a symbol C_i. Similar to that in [5], the basic unit in our clustering scheme is 1-second video clips, with a 0.5-second overlap between two adjacent clips. For each clip, we extract a sequence of frame-based features, including 64-bin HSV color histogram, 9-bin HSV color moment, 24-bin wavelet texture, and 6-bin spatial motion activity, whose mean and standard deviation are then computed and used to characterize this clip[1].

Due to the nature of video streams, the clusters usually have complicated and irregular distributions in the feature space. We use the spectral clustering algorithm [4], which is effective in many applications such as image segmentation and multimedia data clustering [5], to decompose video streams into clusters (symbols). To further improve the clustering performance, we use the self-tuning strategy [8].

For a given video stream, the self-tuning spectral clustering algorithm is carried out as follows:

1. Form an affinity matrix A defined by $A_{ij} = \exp(-d_{ij}^2/(2\sigma_i\sigma_j))$ if $i \neq j$, and $A_{ii} = 0$, where d_{ij} is the Euclidean distance between two clips i and j, and σ_i and σ_j are their scaling factors respectively. For each clip i, σ_i is set based on the context data density of i, and in experiments its value is the mean distance from clip i to its five nearest neighbors in the feature space.
2. Construct the degree matrix D as a diagonal matrix whose i^{th} element on the diagonal is the sum of A's i^{th} row, then define the normalized affinity matrix as $L = D^{-1/2}AD^{-1/2}$.
3. Specify a search range $[n_l, n_h]$ for the most likely number of clusters in the video stream. Suppose $(x_1, \ldots, x_{n_h+1})$ are the n_h+1 largest eigenvectors of L, and $(\lambda_1, \ldots, \lambda_{n_h+1})$ are the corresponding eigenvalues. The optimal number of clusters n is estimated automatically based on the eigen-gaps between adjacent eigenvalues as:
$$n = \arg\max_{i \in [n_l, n_h]} (1 - \lambda_{i+1} / \lambda_i) \ . \qquad (1)$$
4. Form the matrix $X = [x_1x_2\ldots x_n]$ by stacking the first n eigenvectors in columns. Then construct the matrix Y by normalizing X's rows with unit lengths, i.e., $Y_{ij} = X_{ij} / (\Sigma_j X_{ij}^{2})^{1/2}$.
5. Take each row of Y as a point in $\mathbf{R}^n$, and group the rows into n clusters using the cosine-distance based K-means algorithm. The initial centers in the K-means clustering are selected as orthogonal to each other as possible.
6. Assign the i^{th} video clip to cluster C_j, if and only if the i^{th} row of Y is assigned to cluster C_j.

After the clustering, we use some simple heuristic rules, e.g., converting '$C_1C_1C_2C_1C_1$' to '$C_1C_1C_1C_1C_1$', to smooth the cluster label sequence. Finally, the adjacent clips with the same cluster label are merged into homogeneous chunks, thus the video stream is transformed to a sequence of symbols by taking each chunk as one occurrence of the corresponding symbol (cluster).

3. PROBABILISTIC TEMPLATE GENERATION AND PATTERN VERIFICATION

To discover repetitive patterns based on the symbol sequence, most previous methods used exhaustive enumeration through identical matching in the discrete symbol space. As a result, chunks with different symbols are considered as absolutely different in the matching process. In Figure 2(a), a sample sequence of 30 symbols is illustrated, where the patterns obtained by identical matching are listed in Figure 2(b).

"$C_3C_1C_2C_4C_2C_5C_1C_3C_5C_4C_2C_3C_4C_1C_3C_2C_1C_2C_3C_5C_4C_2C_5C_2C_3C_5C_4C_1C_4C_2$"

(a)

Identical Patterns

2 – *symbols*
$C_4 - C_2$
$C_2 - C_3$
$C_4 - C_1$
$C_1 - C_2$

3 – *symbols*
$C_3 - C_5 - C_4$
$C_4 - C_2 - C_5$

4 – *symbols*
$C_3 - C_5 - C_4 - C_2$
$C_2 - C_3 - C_5 - C_4$

(b)

	C_1	C_2	C_3	C_4	C_5
$P(C)$	0.167	0.267	0.200	0.200	0.167

$P(C_j \mid C_i)$	C_1	C_2	C_3	C_4	C_5
C_1	0	0.143	0.167	0.333	0.200
C_2	0.400	0	0.167	0.667	0.200
C_3	0.400	0.429	0	0	0
C_4	0.200	0.143	0167	0	0.600
C_5	0	0.286	0.500	0	0

$$\begin{aligned}&\log P(C_2 - C_3 - C_5 - C_4 - C_2)\\ &= \log P(C_2) + \log P(C_3 \mid C_2) + \log P(C_5 \mid C_3) +\\ &\log P(C_4 \mid C_5) + \log P(C_2 \mid C_4) = -0.756\end{aligned}$$

(c)

Figure 2. Illustrations of (a) an example symbol sequence; (b) the obtained patterns by identical matching; and (c) the prior probabilities and transition matrix for template generation.

However, in the case of video clustering, some symbols (clusters) are close to or even overlap with each other in the low-level feature space, while some others are far apart, i.e., there are "similarities" between symbols. Taking the symbol similarity into account may provide more reasonable pattern discovery results. For example, in Figure 2, if the symbol C_1 and C_2 are very close to each other in the low-level feature space, we may find a new pattern, e.g., the sub-sequences marked in gray color in Fig. 2 (a). We may also combine different patterns into one pattern, e.g., 'C_4-C_1' and 'C_4-C_2' in Figure 2 (b), to describe video contents that may share significant similarity.

In this section, we present a probabilistic template-based approach to video pattern discovery, which finds repetitive video segments

[1] As the frame-based features are in high dimensions, PCA-based dimension reduction is performed to get more reliable experimental results. The feature spaces referred later denote the space after dimension reduction.

without exhaustive enumeration. In addition, this method is able to avoid over-classification.

3.1 Template Generation

In the view of statistics, repetitive patterns result in large temporal transitional probabilities between the adjacent symbols. It is therefore possible to find a set of sub-sequences with high transitional probabilities, called "probabilistic templates", to give clues about the repetitive patterns in the symbol sequence.

To generate the probabilistic templates, first, we define the prior probability for each symbol C_i, and define the transitional probability from symbol C_i to symbol C_j ($1 \le i, j \le n$) as follows:

$$\begin{aligned} &P(C_i) = occur(C_i)\big/N, \\ &P(C_j \mid C_i) = occur('C_i - C_j')\big/occur(C_i), \end{aligned} \qquad (2)$$

where $occur(C)$ is the number of occurrences of a symbol (or a sub-sequence) C, and N is the length of the whole sequence. From the first-order Markov assumption, we can compute the log-probability of a sub-sequence T consisting of L symbols,

$$\log P(T) = \frac{1}{L}[\log P(C_1^T) + \sum\nolimits_{l=2}^{L} \log P(C_l^T \mid C_{l-1}^T)], \qquad (3)$$

where $C_l^T \in \{C_1,\dots, C_n\}$ is the l^{th} symbol in T. Given a pre-defined upper limit of the pattern length L_{max}, the *N-best Viterbi* algorithm is employed to find the sub-sequences that have the N-highest probability scores for each sequence length L ($2 \le L \le L_{max}$), and to form a set of template candidates.

Figure 2 (c) illustrates an example, where the prior probabilities and the transition matrix for the sequence in Figure 2 (a) are shown. The log-probability of the sub-sequence 'C_2-C_3-C_5-C_4-C_2' has the highest score for $L_{max} = 5$, and is taken as a template. It shows that with the probabilistic templates, it is able to find patterns that are not found by identical matching.

The templates are further represented in the low-level feature space. Without loss of generality, we utilize a set of Gaussian distributions to model the symbols in the feature space:

$$\Omega = \{(\mu_i, \Sigma_i) \quad i = 1,\dots,n\}, \qquad (4)$$

where μ_i and Σ_i are the mean vector and covariance matrix of the frame-based features for all the chunks of C_i throughout the video stream. As a result, we represent the template T in the feature space as follows:

$$\theta_T = [(\mu_1,\Sigma_1),\dots,(\mu_l,\Sigma_l),\dots,(\mu_L,\Sigma_L)]_T, \qquad (5)$$

where $(\mu_l, \Sigma_l) \in \Omega$. Given this representation, we perform pattern matching in the continuous feature space, which makes our approach more robust to errors caused by video symbolization.

3.2 Pattern Verification

In the pattern verification process, for a given template T of length L, we go through all the sub-sequences containing L symbols in the symbol sequence, and measure their match probabilities of T. The match probability of a sub-sequence S given the template T is defined as:

$$P(S \mid \theta_T) = \sum\nolimits_{l=1}^{L} (\log P(O_l^S \mid \mu_l^T, \Sigma_l^T) / \| C_l^S \|), \qquad (6)$$

where O_l^S is the feature sequence of all the video frames belonging to the l^{th} symbol in S, and $\|C_l^S\|$ is the number of video frames in this symbol. In our experiments, all the sub-sequences whose matching scores are greater than a pre-defined threshold are selected to form the matching set Φ_T for T. If Φ_T is null, T is removed from the template candidates; otherwise the sub-sequences in Φ_T are sorted according to their matching scores which are useful in such applications as video indexing.

Furthermore, to avoid over-classification of similar video segments into different patterns, we merge the pattern templates whose matching sets have considerable overlaps, as they are very likely associated with one pattern. Formally, the templates T_a and T_b are merged if they satisfy:

$$\| \Phi_{T_a} \cap \Phi_{T_b} \| / \min(\| \Phi_{T_a} \|, \| \Phi_{T_b} \|) > \tau, \qquad (7)$$

where $\|\Phi_T\|$ is the number of sub-sequences in the matching set Φ_T, and τ is set as 0.6 in our experiments.

4. EXPERIMENTS

To demonstrate the robustness and efficacy of the proposed approach, we tested it on several sports videos. The data set contains broadcasts of diving (65min) and basketball (70min). The experiments were carried out in three steps: i) examining the clustering performance relative to the manually labeled ground truth; ii) comparing the discovered patterns using probabilistic template and using identical matching; and iii) subjective evaluation of the pattern qualities.

Table 1. Correlations between the manually labeled ground truth (G_i) and the clustering results (C_i)

	Category in ground truth	*Clustered symbols*
Diving	*G_1: execution of dive: take-off, flight, and entry*	C_1, C_{14}
	G_2: close up of diver and degree of difficulty	C_7, C_{10}
	G_3: diver prepares to dive on springboard	C_3, C_8, C_{12}
	G_4: diver walks and rests after diving	C_2, C_5, C_{13}
	G_5: showing diver's score and rank	C_4, C_6, C_9
	G_6: remote audience view	C_{11}
Basketball	*G_1: global view of basketball court*	C_1, C_4
	G_2: foul shot from the foul line	C_2, C_{13}, C_{15}
	G_3: mid-view of passing, throwing and taping	C_7, C_{10}, C_{11}, C_{14}
	G_4: close-up of player, coach, and referee	C_6, C_9, C_5, C_{12}
	G_5: remote audience view	C_3, C_8

First, as presented in Section 2, we have adopted the self-tuning spectral clustering algorithm to transform the continuous video stream to a discrete symbol sequence. In the clustering algorithm, we specified the search range of the number of clusters as [5, 30], from which we finally got 14 and 15 clusters for the diving and basketball videos respectively. To evaluate the clustering performance, we established the ground truth via integrating the results labeled by three graduate students who had analyzed the video content and the obtained clusters. As shown in Table 1, for one category in the ground truth, as its content often has multiple visual appearances, the clustering algorithm produced several clusters. We have manually associated these clusters with their closest semantic categories in the ground truth. We also computed the classification accuracies for all the semantic categories in the ground truth. The average error rates are below 15% (the detailed information is not listed due to the page limitation). As shown, the performance is acceptable and practicable enough for the video decomposition and symbolization.

Second, we discovered repetitive patterns based on the generated symbol sequences. To verify the performance of our probabilistic template-based algorithm, we have compared it with a well-known identical matching approach, the TEIRESIAS algorithm, which was developed for pattern discovery in biological sequences [2]. Table 2 lists the patterns discovered by identical matching (M_i) and by probabilistic templates (M_p). For the results of M_p, each [.] denotes one pattern, in which the templates have been merged using equation (7).

Table 2. Comparisons of the discovered patterns by identical matching (M_i) and probabilistic template (M_p)

	Diving	*Basketball*
M_i	C_7-C_8; C_{10}-C_{12}; C_7-C_5; C_3-C_1; C_2-C_{14}; C_8-C_1; C_5-C_9; C_2-C_6; C_5-C_{10}-C_1; C_{12}-C_7-C_{14}; C_1-C_{11}-C_{13}	C_1-C_9; C_1-C_{12}; C_4-C_6; C_4-C_7; C_1-C_5; C_{10}-C_6; C_{14}-C_9; C_2-C_5; C_{15}-C_1; C_2-C_4; C_3-C_1-C_9; C_4-C_5-C_1
M_p	[C_7-C_8, C_{10}-C_{12}, C_7-C_5]; [C_3-C_1, C_2-C_{14}, C_8-C_1]; [C_5-C_9, C_2-C_6, C_3-C_4]; [C_2-C_6-C_7, C_{12}-C_6-C_7]; [C_5-C_{10}-C_1, C_{12}-C_7-C_{14}, C_8-C_7-C_1]; C_1-C_{11}-C_{13}; C_{10}-C_3-C_1-C_{13}	[C_1-C_9, C_1-C_{12}, C_4-C_6]; [C_4-C_7, C_1-C_5]; [C_{10}-C_6, C_{14}-C_9, C_2-C_5]; [C_{15}-C_1, C_2-C_4, C_{10}-C_4]; [C_{12}-C_2, C_{11}-C_{13}]; [C_3-C_1-C_9, C_8-C_4-C_9]; [C_4-C_5-C_1, C_4-C_7-C_1]

From Table 2, we can see that the patterns discovered by M_i and M_p have some different characteristics: i) M_p generally generates fewer patterns than M_i. Taking the diving video for an example, M_i and M_p discovered 11 and 7 patterns respectively. By investigating these patterns, we noticed that some patterns found by M_i had been merged into one pattern by M_p. For instance, 'C_5-C_9' and 'C_2-C_6' are discovered by M_i as two distinct patterns (a case of over-classifications) in the diving video, although they in fact both describe the same content for "the diver walks and rests after the diving – the diver's score and rank are shown on screen." In contrast, M_p merged these two as one pattern by using the similarities between symbols in the low-level feature space, which correctly reflects the actual video content; and ii) M_p was able to find extra patterns missed by M_i. For instance, 'C_{10}-C_3-C_1-C_{13}' reported by M_p describes the segments of "first is a close up of the diver – then the diver makes some preparations on the springboard – next the diver executes the dive – finally the diver walks and rests after the diving," which is an important pattern that repeats in the diving video. Such patterns were missed by M_i because it does not appear identically in the symbol sequence, whereas M_p was able to reveal them through exploring the temporal transition probabilities among symbols based on similarities.

Table 3. Subjective evaluation of the obtained patterns

Video	*Reasonableness*		*Completeness*		*Informativeness*	
	M_i	M_p	M_i	M_p	M_i	M_p
Diving	72.7	71.4	27.3	52.9	62.9	68.3
Basketball	66.7	67.1	33.3	47.1	58.3	62.9

Finally, we carried out a subjective evaluation of the discovered patterns. To completely inspect the subjective quality of a given pattern, we asked five participants to evaluate whether the pattern really repeats throughout the video stream (*reasonableness*), whether its instances are picked out completely (*completeness*), and whether it provides useful information for video understanding (*informativeness*). The five participants were asked to give scores good (100), neutral (50), and bad (0) for the three qualities, respectively. We then averaged the scores over various participants to get a general evaluation for each obtained pattern.

Table 3 shows the detailed subjective evaluation results: i) in terms of *reasonableness*, the participants felt satisfied with the patterns discovered by both M_i and M_p. It shows that either by M_i or M_p, the reported patterns did repeat throughout the video streams; ii) in terms of *completeness*, they considered M_p performs much better than M_i. This demonstrates indeed that M_p was able to identify the segments for a specific pattern more completely. This is because M_p resolves the over-classification problem by pattern verification in the low-level feature space, whereas M_i usually splits segments of similar contents into distinct patterns; and iii) in terms of *informativeness*, the performances of M_p are again better than M_i, as M_p is able to find patterns that are otherwise missed by M_i and represent important, repetitive contents in video streams. Our system is able to provide the user (audience) with more useful information. Generally, the results for diving are better than basketball, as diving is far more structured than basketball and it is easier to reach a satisfactory result to the user.

5. CONCLUSIONS

In this paper, we have proposed an effective and robust approach to discovering repetitive segments from video streams. In this approach, the self-tuning spectral clustering is first employed to decompose the video stream into a symbol sequence, based on which a set of probabilistic templates are generated by exploiting the temporal transitions among various symbols. The obtained templates are verified throughout the video stream in the low-level feature space to detect and locate the repetitive patterns Experiments on sports videos show that our approach achieves encouraging results in practice. To make further improvements we are now working on automatic selection of the proper number of templates in the unsupervised mining; and carry out theoretic analysis and experimental evaluations to make the method applicable to more video genres and other media contents, which will be reported in our future papers.

6. ACKNOWLEDGMENTS

This research has been supported in part by research grants from National Natural Science Foundation of China No. 60273008, City University of Hong Kong No.7001679, and Hong Kong RGC CityU 1062/02E.

7. REFERENCES

[1] Divakaran, A., Peker, K., Chang, S.-F., Radhakrishnan, R., and Xie, L. Video mining: pattern discovery versus pattern recognition, *ICIP'04*, 2379-2382, 2004.

[2] http://cbcsrv.watson.ibm.com/Tspd.html

[3] Leonardi, R., Migliorati, P., and Prandini, M. Semantic indexing of soccer audio-visual sequences: a multimodal approach based on controlled Markov chains, *CSVT*, 14(5):634-643, 2004.

[4] Ng, A. Y., Jordan, M. I., and Weiss, Y. On spectral clustering: analysis and an algorithm, *NIPS'01*, 849-856, 2001.

[5] Radhakrishnan, R., Divakaran, A., and Xiong, Z. A time series clustering based framework for multimedia mining and summarization using audio features, *ACM MIR'04*, 157-164, 2004.

[6] Wang, P., Cai, R., and Yang, S.-Q. A tennis video indexing approach through pattern discovery in interactive process, *IEEE PCM*, *LNCS*, Tokyo, Japan, Dec. 2004, 3331:49-56.

[7] Yeung, M., Boon-Lock Yeo. Time-constrained clustering for segmentation of video into story units, *ICPR'96*, 3:375-380, 1996.

[8] Zelnik-Manor, L., and Perona, P. Self-tuning spectral clustering, *NIPS'05*, 1601-1608, 2005.