DETECTING PITCHING FRAMES IN BASEBALL GAME VIDEO USING MARKOV RANDOM WALK

Chih-Yi Chiu, Po-Chih Lin[†]*, Wei-Ming Chang*[*]*, Hsin-Min Wang*[‡]*, and Shi-Nine Yang*[†]
Department of Computer Science and Information Engineering, National Chiayi University, Taiwan
[†] Department of Information and Computer Engineering, Chung Yuan Christian University, Taiwan
[*] Department of Computer Science, National Tsing Hua University, Taiwan
[‡] Institute of Information Science, Academia Sinica, Taiwan

ABSTRACT

Pitching is the starting point of an event in baseball games. Hence, locating pitching shots is a critical step in content analysis of a baseball game video. However, pitching frames vary with innings and games. Existing methods that require a great deal of effort to construct empirical rules or label training data do not capture the characteristics of various pitching frames very well. In this paper, we present an unsupervised method for pitching frame detection by using Markov random walk. A video stream is first divided into content-homogeneous shots, and these shots are merged into states through hierarchical agglomerative clustering. Then, the state with the highest visit probability according to the Markov random walk theory is deemed the set of pitching frames. Finally, a model trained on the pitching frames in the pitching state is further used to detect the remaining potential pitching frames in other states. Our experiments demonstrate that the proposed method yields satisfactory results in a variety of MLB games.

Index Terms**—**Event detection, video annotation, hierarchical agglomerative clustering, Bayesian information criterion, Markov random walk

1. INTRODUCTION

Sports video analysis has attracted increasing attention in recent years. This is because a complete game (e.g., baseball and basketball) may take several hours, and a professional sports league (e.g., MLB and NBA) has hundreds of games in a season. As a result, there are now large collections of sports videos that people need to manage and access in an efficient manner. To this end, content analysis techniques for sports videos, including scene classification, event detection, highlight extraction, game summarization, and tactic analysis, have been published in many multimedia-related journals and conferences.

In this paper, we discuss pitching frame detection, which we believe is the basis of content analysis of a baseball game video. Pitching, which is the act of throwing a baseball toward the home plate to start play, is the first state of an event in baseball games. Detecting pitching frames is not a trivial task because they vary with innings and games. As shown in Fig. 1, the variations include:

- left/right-handed batters/pitchers;
- runners/basemen/umpires;
- advertisements/viewers;
- grass/soil colors/layouts;
- teams' sport shirts;
- camera views; and
- captions superimposed by TV channels.

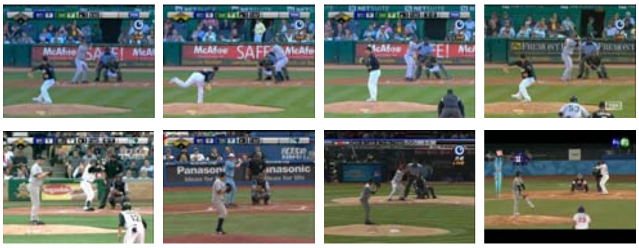

Fig. 1. Various pitching frames in innings and games

1.1. Related work

Existing pitching frame detection methods can be classified into two categories: rule-based and model-based. Rule-based methods define rules through empirical observations. For example, Chu and Wu [4] counted the ratio of the field area (grass and soil) in a frame and checked if it satisfied a predefined criterion of a pitching frame. A similar concept was also applied in [6]. The predefined rules might be disobeyed in different games. Zhang and Chang [8] employed caption recognition to check the score box. When the box information was updated, a rule-based decision tree was used to determine pitching shots. Their method is only workable for a particular score box pattern.

Model-based methods use supervised learning that needs labeled training data to learn the model of pitching frames. For example, Chang *et al*. [2] used a Bayesian classifier while Ando *et al*. [1] used a hidden Markov model (HMM). The performances of the model-based methods are highly dependent on the coverage of the training data. Even though a great deal of effort has been devoted to the collection of diverse training data, such methods might still fail to handle the cases unseen in the training data.

1.2. Overview of the proposed method

In this paper, we propose a novel unsupervised method for pitching frame detection. We first detect some frames that are deemed pitching frames, and then train a model from these pitching frames to detect the remaining pitching frames. Our method has two advantages over the above-mentioned methods. First, its model is built for every half-inning so that it can adaptively reflect the characteristics of a half-inning regardless of the variations discussed above. Second, it does not need to define heuristic rules or label diverse training data beforehand.

Our method consists of four steps. First, the video stream of a half-inning is divided into several shots. Second, hierarchical agglomerative clustering (HAC) is applied to merge similar shots into clusters based on the Bayesian information criterion (BIC). Third, the cluster that is mainly comprised of pitching frames is identified by Markov random walk. As shown in Fig. 2, the clusters are denoted as states, and the video stream is viewed as a state transition graph. By solving the Markov random walk problem, we obtain the steady visit probability of each state. Since pitching is the starting point of every baseball event, the state consisting of pitching frames should be visited the most often. In other words, the state with the highest visit probability is considered the pitching state. Finally, a model trained from the frames belonging to the pitching state is used to detect the remaining pitching frames. Note that, even though some pitching frames are not included in the pitching state, they can still be identified as pitching frames by the model.

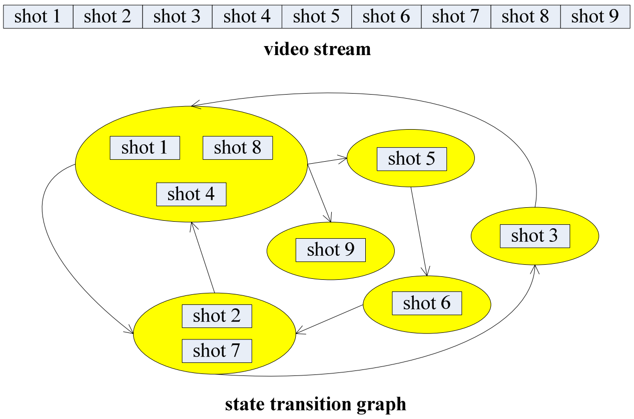

Fig. 2. The state transition graph of a video stream

The remainder of this paper is organized as follows. In Section 2, we present video shot segmentation and clustering. In Section 3, we describe our pitching frame detection method by using Markov random walk and frame classification. Section 4 details the experiment results. Section 5 contains some concluding remarks and indicates future research avenues.

2. VIDEO SHOT SEGMENTATION AND CLUSTERING

2.1. Video shot segmentation

The first step of our approach divides the video content of a half-inning into content-homogeneous shots. Note that, in this study, a shot is defined as a sequence of frames with homogeneous content, rather than a sequence of frames between two camera cuts. We exploit the self-similarity information [5] for video shot segmentation. The self-similarity information is acquired by constructing a self-similarity matrix, where the (*i*, *j*)-th element means the similarity between the *i*-th and *j*-th frames. A checkerboard kernel is used to convolute with the self-similarity matrix along its main diagonal. Local maxima in the convolution series that are greater than a predefined threshold are considered shot boundaries.

2.2. Video shot clustering

After obtaining a set of video shots, hierarchical agglomerative clustering with the Bayesian information criterion (HAC-BIC) is applied to merges similar shots in a bottom-up manner. At beginning, each shot is considered a cluster. Then, in each merging step, for any two clusters X and Y, two hypotheses are given: H_0 posits that X and Y are derived from the same Gaussian, while H_1 posits that they are derived from two distinct Gaussians. Let $Z = X \cup Y$. The difference between the BIC values of H_1 and H_0 is computed by [7]:

$$\begin{aligned}\Delta BIC(X,Y) &= BIC(H_1,Z) - BIC(H_0,Z) \\ &= \frac{n}{2}\log|\Sigma| - \frac{n_X}{2}\log|\Sigma_X| - \frac{n_Y}{2}\log|\Sigma_Y| \\ &\quad - \frac{1}{2}\eta\left(d + \frac{1}{2}d(d+1)\right)\log n,\end{aligned} \tag{1}$$

where n_X, n_Y, and n are the number of frames in X, Y, and Z, respectively; Σ_X, Σ_Y, and Σ are the covariance matrixes of X, Y, and Z, respectively; $|\Sigma|$ is the determinant of Σ; d is the dimension of the frame's feature vector; and η is a scalar controlling the merging probability of two clusters. X and Y can be merged only when $\Delta BIC(X,Y) < 0$. At each merging step, only the two clusters with the smallest ΔBIC are merged into a new cluster. The clustering process stops when none of any two clusters can be merged.

Since shot segmentation is usually imperfect, a shot might contain parts of frames with inhomogeneous content. Such shots might affect clustering and should be removed during the clustering step. To do this, we verify the trace of each shot's covariance matrix as follows. If a shot's trace (i.e., the variance) is larger than a predefined threshold, its frame content is considered inhomogeneous, and the shot is excluded in HAC-BIC. In addition, if the determinant of the covariance matrix of a shot is close to zero (i.e., singular or near singular), it is also excluded in HAC-BIC.

3. PITCHING FRAME DETECTION

3.1. Pitching cluster selection by Markov random walk

We use Markov random walk, which models a discrete-time stochastic process, to formulate transition probabilities of video shot clusters. Hereafter, a video shot cluster is denoted as a "state." The transition probabilities will converge to yield a set of steady state probabilities, each of which represents the visit frequency of a state. Thus, the state with the highest frequency is considered the pitching cluster based on the previous discussion.

Let P be a Markov chain characterized by an $N\times N$ matrix, where N is the number of states, and P_{ij} is the one-step transition probability from state i to state j. P is constructed as follows. First, the inter-transition probability from state i to state j, denoted as A_{ij}, is defined as:

$$A_{ij} = a_{ij} / \sum_{j=1}^{N} a_{ij} \text{ for } i \neq j, \text{ and } A_{ii} = 0 \tag{2}$$

where a_{ij} is the number of transitions from state i to state j. Next, the intra-transition probability of state i, denoted as B_{ii}, is defined as:

$$B_{ii} = size(i) / \sum_{i=1}^{N} size(i), \tag{3}$$

where $size(i)$ returns the number of frames in state i. We also define

$$B_{ij} = (1 - B_{ii})/(N-1) \quad \text{for } i \neq j. \tag{4}$$

In addition, a uniform transition probability is set for every pair of states i and j:

$$C_{ij} = 1/N. \tag{5}$$

Finally, P is obtained by:

$$P = \alpha\cdot A + \beta\cdot B + \gamma\cdot C, \tag{6}$$

where $0 \leq \alpha, \beta, \gamma \leq 1$ and $\alpha+\beta+\gamma = 1$. Note that according to Eqs. (2)-(6), A, B, C, and P are defined to be *stochastic matrixes* that have the following property:

$$\forall i, j, Z_{ij} \in [0,1] \text{ and } \forall i, \sum_{j=1}^{N} Z_{ij} = 1, \tag{7}$$

where Z is A, B, C, or P. Thus, we can compute P's principal left eigenvector, π, with the largest eigenvalue 1:

$$\pi \cdot P = 1 \cdot \pi . \tag{8}$$

π can be interpreted as a $1\times N$ vector of steady state probabilities, and the i-th entry is state i's visit frequency.

Under the assumption that the state comprising pitching frames should be visited the most often, the state with the highest visit probability in π, denoted as S, is regarded as the set of pitching frames. Since some pitching frames might distribute over states (i.e., clusters) other than S or not be included in any cluster in HAC-BIC, a model is trained on the pitching frames in S to detect the remaining potential pitching frames in other states, as will be depicted in the following subsection.

3.2. Frame classification

We model state S by a uni-Gaussian function to represent the characteristics of pitching frames of the half-inning. Given a video frame f, its similarity to S is calculated as:

$$sim(f \mid S) = \frac{\exp\left(-\frac{1}{2}(v_f - \mu_S)^T \Sigma_S^{-1} (v_f - \mu_S)\right)}{(2\pi)^{d/2} \mid \Sigma_S \mid^{1/2}}, \tag{9}$$

where v_f is the feature vector of frame f; d is the dimension of v_f; and μ_S and Σ_S are, respectively, the sample mean vector and the sample covariance matrix estimated from S. Frames whose similarities are greater than a predefined threshold are considered pitching frames.

4. EXPERIMENTS

4.1. The MLB dataset

We selected fourteen half-innings of four games from Major League Baseball (MLB) 2008 as the dataset; the four games vary in teams, stadiums, and broadcasting channels. The video data were transformed to 160×120 frame pixels and 15 frames per second. The pitching frames were manually labeled. Table 1 lists the detailed information of the dataset. Asterisks indicate the data used for training in the model-based method.

A video frame was partitioned into 4×4 non-overlapping blocks, from each block we computed the average intensities of Y, Cb, and Cr channels. Hence, each frame was represented by a 48-dimensional feature vector.

4.2. Methods compared

Two baseline methods were implemented for comparison: the rule-based method proposed by Chu and Wu [4] and the model-based method based on support vector machines (SVM) [2]. Chu and Wu's method adaptively chose the field color in the hue-luminance-saturation (HLS) color space, and computed the field ratio of each frame. For the frames whose field ratios were between two predefined thresholds, the rule of horizontal and vertical profiles was used to determine whether they were pitching frames. In the SVM method, the frame feature representation was the same as that in our method. Four half-innings (marked * in Table 1) were used to train the SVM classifier. The training accuracy was 1.00 for recall and 0.90 for precision. Our method used the following configuration in the experiments: in Eq. (1), η = 2.5; in Eq. (7), α = 0.4, β = 0.5, and γ = 0.1.

Table 1. The MLB dataset used in the experiments

Game	Inning	Info. (* means training data)
2008/06/10 NYY vs. OAK	1 top	16170 frames / 3180 pitching frames (*)
	2 bottom	5820 frames / 2085 pitching frames (*)
	3 top	8790 frames / 2070 pitching frames
	4 bottom	7380 frames / 1530 pitching frames
	5 top	9285 frames / 2235 pitching frames
	5 bottom	4335 frames / 1785 pitching frames
2008/06/16 BOS vs. PHI	1 top	7095 frames / 1260 pitching frames (*)
	2 bottom	10815 frames / 2010 pitching frames (*)
	3 top	4785 frames / 1035 pitching frames
	4 bottom	5070 frames / 1650 pitching frames
2008/06/18 SD vs. NYY	3 top	20160 frames / 7440 pitching frames
	4 bottom	6210 frames / 1980 pitching frames
2008/06/24 NYY vs. PIT	6 bottom	7650 frames / 2790 pitching frames
	7 top	8535 frames / 2160 pitching frames

4.3. Experiment results

The experiment results are summarized in Table 2. **R** and **P** are the abbreviations of recall and precision rates, respectively. Overall speaking, the SVM method outperforms Chu and Wu's method in Games 2008/06/10 and 2008/06/16, where parts of games were used for training. In Games 2008/06/18 and 2008/06/24, however, the SVM method does not perform as well as Chu and Wu's method. The results show that both methods do not have a good generalization ability.

From Table 2, it is clear that our method demonstrates a satisfactory result over the two baseline methods. Basically, our method can also be categorized as a rule-based method. The only assumption, however, is that pitching is the starting point of every baseball event such that its state should be visited the most often. This simple assumption has shown its generalization in a variety of innings and games. In addition, our method is an unsupervised method; it does not require a great deal of effort to manually label diverse training data.

5. CONCLUSION AND FUTURE WORK

We have proposed an unsupervised method for pitching frame detection. A video stream of a half-inning is first segmented into content-homogeneous shots. Then, the shots are clustered into states (i.e., clusters) through the HAC-BIC technique. Finally, the state associated with the pitching frames is determined by the Markov random walk theory under the assumption that the state with the highest visit probability corresponds to the pitching frames. Our experiment results demonstrate that the proposed method yields a relatively robust performance in the MLB dataset compared to two baseline methods. In our future research, we will utilize this work to analyze more baseball events.

6. ACKNOWLEDGEMENT

This work was supported in part by the National Science Council of Taiwan under Grants: NSC 98-2218-E-415-003 and NSC 98-2631-H-001-013.

Table 2. The results of pitching frame detection

Game	Inning	Chu & Wu		SVM		Our	
		R	P	R	P	R	P
2008/06/10 NYY vs. OAK	1 top	0.94	1.00	–	–	0.95	0.93
	2 bottom	0.75	0.77	–	–	1.00	0.94
	3 top	0.95	1.00	1.00	0.99	0.91	1.00
	4 bottom	0.69	0.73	0.89	0.82	0.82	1.00
	5 top	0.94	1.00	0.91	0.97	0.92	1.00
	5 bottom	0.70	0.78	0.83	0.84	0.87	1.00
2008/06/16 BOS vs. PHI	1 top	1.00	0.80	–	–	0.99	0.99
	2 bottom	0.95	0.82	–	–	0.98	0.99
	3 top	1.00	0.74	1.00	0.96	1.00	0.99
	4 bottom	0.95	0.88	1.00	0.93	0.95	0.99
2008/06/18 SD vs. NYY	3 top	1.00	0.92	1.00	0.84	0.98	0.99
	4 bottom	1.00	0.83	1.00	0.79	0.94	1.00
2008/06/24 NYY vs. PIT	6 bottom	0.98	0.93	0.96	0.85	1.00	0.98
	7 top	0.98	0.90	1.00	0.65	1.00	0.99

7. REFERENCES

[1] R. Ando, K. Shinoda, S. Furui, and T. Mochizuki, "A robust scene recognition system for baseball broadcast using data-driven approach," *ACM International Conference on Image and Video Retrieval (CIVR)*, 2007.

[2] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[3] P. Chang, M. Han, and Y. Gong, "Extract highlights from baseball game video with hidden Markov Models," *IEEE International Conference on Image Processing (ICIP)*, 2002.

[4] W. T. Chu and J. L. Wu, "Explicit semantic events detection and development of realistic applications for broadcasting baseball videos," *Multimedia Tools and Applications*, Vol. 38, No. 1, pp. 27-50, 2008.

[5] J. Foote and M. Cooper, "Media Segmentation using Self-Similarity Decomposition," *SPIE Storage and Retrieval for Multimedia Databases*, 2003.

[6] C. C. Lien, C. L. Chiang, and C. H. Lee, "Scene-based event detection for baseball videos," *Journal of Visual Communication and Image Representation*, Vol. 18, No. 1, pp. 1-14, 2007.

[7] M. Nishida and T. Kawahara, "Unsupervised speaker indexing using speaker model selection based on Bayesian information criterion," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003.

[8] D. Zhang and S. F. Chang, "Event detection in baseball video using superimposed caption recognition," *ACM International Conference on Multimedia (ACM-MM)*, 2002.