

멀티 모달 분석을 통한 야구 동영상에서의 실시간 중요 이벤트 검출 알고리즘

*이진호, **김형국

광운대학교 전자공학과

*trampclown@kw.ac.kr, **hkim@kw.ac.kr

Rapid Detection of Important Events in Baseball Video Using multi-Modal Analysis

*Lee, Jin-Ho **Kim, Hyoung-Gook

Kwangwoon University, Department of Wireless Communications Engineering

요약

본 논문에서는 야구 동영상에서 실시간으로 중요 이벤트 장면을 검출하는 알고리즘을 제안한다. 제안하는 알고리즘은 영상 정보를 분석하여 Pitching 신과 Close Up 신을 추출하여 Play 구간을 검출하고, 오디오 정보를 분석하여 오디오 이벤트 구간을 검출한다. Play 구간의 시작인 Pitching 신을 검출하기 위해서는 오프라인 모델과 온라인 모델을 혼용하여 다양한 환경에 상관없이 높은 성능을 보일 수 있도록 하였으며, 아나운서의 억양 및 관중의 함성의 고조도가 높아지는 구간을 기반으로 검출된 오디오 이벤트 구간을 영상 정보 분석을 통해 획득된 Play 장면구간을 결합하여 중요 이벤트 장면 검출의 정확도를 높일 수 있도록 하였다. 실험에 의하면 제안하는 알고리즘은 1초의 동영상 데이터를 처리하는데 0.024초의 소요 시간이 필요하고, 0.89의 Recall과 0.975의 Precision 검출 성능을 보임을 알 수 있었다.

1. 서론

최근 타임 시프트와 같은 기술의 발전으로 사용자가 원하는 콘텐츠에 접근하여 능동적으로 시청 구간을 지정하여 볼 수 있게 되었다. 하지만, 사용자는 원하는 콘텐츠에 중요한 이벤트를 시청하기 위해서는 직접 콘텐츠를 확인해야 하는 번거로움이 있다. 특히, 스포츠는 방송 전체 시간이 다른 방송에 비해 길기 때문에 타임 시프트를 이용하여 시청하는 것은 많은 불편함이 존재한다. 이러한 기존의 타임 시프트의 문제점을 해결하기 위해 콘텐츠 내의 중요한 이벤트를 자동으로 검출해주는 중요 이벤트 검출 알고리즘이 필요하다.

스포츠 비디오에서의 중요 이벤트 검출을 위한 연구는 다양한 방향으로 진행되어 왔다. 먼저, 스포츠 이벤트의 기본 단위가 되는 플레이 구간(Play 구간)을 검출하기 위해서 단순한 특징 정보 비교에 의한 템플릿 매칭 기반 검출 방법[1], 모델을 미리 만들어 이용하는 오프라인 학습 기반 검출 방법[2], 각 비디오 콘텐츠의 특성을 반영하는 적응적 모델 기반 검출 방법[3] 등으로 분류되어 연구되었다. 하지만, 템플릿 매칭 기반 검출 방법[1]과 오프라인 학습 기반 검출 방법[2]은 다양한 경기 특성을 반영하지 못하여 검출 성능이 저하되는 문제가 있으며, 적응적 모델 기반 검출 방법[3]의 경우, 비디오 전체에 대한 분석이 선행되어야 하기 때문에 타임 시프트와 같이 방송 도중에 Play 구간이 검출되어야 하는 응용 분야에는 부적합하다. 중요 이벤트 검출 알고리즘은 크게 오디오 기반 검출 방법[4], 비디오 기반 검출 방법[5], 멀티모달 기반 검출 방법[6] 등으로 나눌 수 있다. 오디오 기반 검출 방법[4]은 중요 이벤트의 기본 단위인 Play 구간 검출 과정이 필요하며, 또한 이벤트의 시작과 끝 지점을 알 수 없는 문제점이 있다. 그 외 나머지 방법[5, 6]들은 적응적 모델 기반 검출 방법[3]과 마찬가지로, 비디오 전체에 대한 분석이 먼저 선행되어야 하는 문제점이 있다.

본 논문에서는 실시간으로 야구 동영상상의 중요 이벤트를 검출하는 알고리즘을 제안한다. 제안하는 방법에서는 먼저 입력되는 동영

상을 비디오와 오디오로 분리한 후에, 비디오 데이터로 플레이 시작 신(Play 시작 신)과 끝 신(Play 끝 신)의 검출을 통해 Play 구간을 자동으로 검출하게 하고, 오디오 데이터로는 SVM에 의해 미리 학습된 모델을 이용하여 오디오 이벤트 지점을 검출한다. 비디오 데이터에서 Pitching 신의 검출 성능 향상을 위해 오프라인 상에서 학습된 모델과 경기 내에서 생성된 적응적 온라인 모델을 혼합하여 사용하며, Play 끝 신은 검출된 Play 시작 신의 대표 프레임의 필드 색을 이용하여 검출하고, 검출된 Play 구간과 오디오 이벤트 구간은 중요 이벤트 구간 결정을 위한 입력으로 사용된다. 본 논문에서는 이러한 실시간 중요 이벤트 검출을 위해 다음과 같은 세 가지의 요구 사항을 정의하였다.

- 즉시성(실시간 처리) : 타임 시프트와 같은 응용 분야에서는 중요 이벤트 구간이 재생이 된 이후에 바로 이벤트 검출이 되어야 한다.
- 정확성 : 제공되는 중요 이벤트만으로도 지나간 방송의 내용을 파악할 수 있어야 한다.
- 지역성 : 지나간 방송에 대한 특성을 반영하기 힘든 타임 시프트와는 달리, 전체 방송의 특성을 반영해야 한다.

2. 실시간 야구 동영상 중요 이벤트 검출 알고리즘

야구 경기 방송은 다음 <그림 1>과 같은 경기 전 세그먼트, 야구 세그먼트 그리고 경기 후 세그먼트로 나누어진다.



그림 1. 야구 경기 방송 구조

여기서, 야구 세그먼트가 실제 선수들이 플레이하는 장면들이 담겨져

있으며, 이는 투수가 볼을 던져 타자가 받아 치는 Play구간과 그 외 다른 장면을 보여주는 Break 구간으로 나누어지며, Play구간과 Break구간은 경기가 종료될 때까지 반복되어진다. 야구 세그먼트는 Pitching 신과 Close Up 신으로 구성되며, Play 구간의 재생 시간은 영상에서 Pitching 신이 나타났을 경우, Close Up 신이 검출될 때까지 걸리는 시간이다. 그림에서 보면 알 수 있듯이, 안타나 홈런 등 득점 상황 시 Play 구간(14초)에서는 일반적인 Play 구간(4초)보다 재생 시간이 오래 지속된다. 즉, 중요한 이벤트에 해당하는 Play 구간은 재생 시간이 길고, 아나운서와 관중의 흥분소리 등 일반적인 Play구간과는 다른 오디오 특징을 가지고 있다.

본 논문에서는 언급한 야구 동영상의 특성들을 이용하여 야구 경기의 중요 이벤트 구간 검출을 위한 알고리즘 <그림2>을 제안한다.

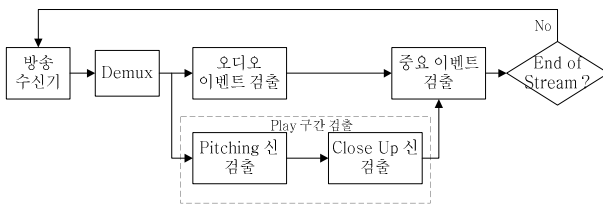


그림 2. 전체 시스템 구조

방송 수신기에서 야구 방송 동영상이 입력되면, 입력된 동영상은 DEMUX에 의해 영상신호와 오디오 신호로 각각 분리되어 신호처리가 수행된다. 오디오 신호로부터 오디오 이벤트가 검출되고, 영상 신호로부터는 Pitching 신과 Close Up 신을 검출하여 Play 구간을 결정하게 되며, 검출된 Play 구간과 오디오 이벤트는 중요 이벤트 구간을 검출하기 위한 입력으로 사용된다.

3.1. Play 구간 검출 방법

Play 구간 검출 방법은 <그림 2>에서와 같이 Pitching 신 검출과 Close Up 신 검출로 구성된다. 그 중 Pitching 신은 투수가 투구하는 마운드의 뒤쪽에서 홈을 바라보는 구도로, 거의 모든 야구 경기에서 비슷하게 촬영되며, 동일 경기 내에서 색 정보나 전체 화면 구성이 동일하다. 반면 서로 다른 경기에서는 구도는 비슷하지만, 색 정보는 필드 색이나 날씨, 조명 등에 의해서 많이 다르게 나타난다.

이러한 특성을 가지는 Pitching 신을 효과적으로 검출하기 위해 오프라인 모델과 경기 내에서 적응적으로 학습된 모델(온라인 모델)을 결합하여 Play 구간 검출에 적용한다. 오프라인 모델만은 다양한 환경에 대해 범용성이 떨어지며, 온라인 모델만은 통계적 정보를 요구하기 때문에 실시간 검출이 어려워진다. 본 논문에서는 두 모델을 혼용하여 언급했던 문제점을 보완하고 야구 경기의 Play 구간 검출 성능을 높이는 동시에 서론에서 설명했던 세 가지 요구 사항을 만족시키도록 하였다. 다음 <그림 3>는 Pitching 신 검출 알고리즘의 구조를 나타낸다.

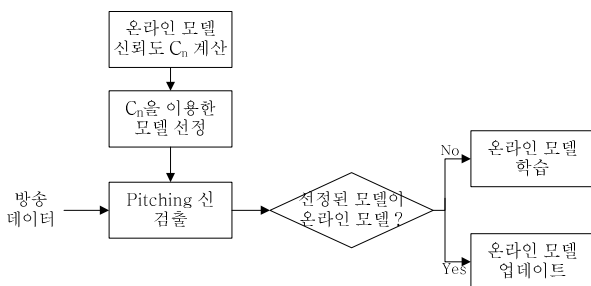


그림 3. Pitching 신 검출 알고리즘 구조

Pitching 신 검출은 온라인 모델의 신뢰도 C_n 이 특정 임계값보다 작을 때 오프라인 모델만을 이용하여 수행하고, 오프라인 모델에 의해 검출된 Pitching 신은 온라인 모델을 생성하게 된다. 생성된 온라인 모델의 신뢰도 C_n 이 특정 임계값보다 높은 경우 온라인 모델을 이용하여 Pitching 신을 검출하고, 검출한 Pitching 신을 이용하여 온라인 모델을 업데이트한다. 오프라인 모델은 실시간 검출을 위해 Linear SVM을 이용하여 생성하였으며, 위에서 언급한 Pitching 신의 특징을 반영하기 위하여 특정 벡터로 ECH(Edge Component Histogram)[7]를 사용하였다.

오프라인 모델은 온라인 신뢰도 C_n 이 특정 임계값보다 작을 때 Pitching 신을 검출하기 위해 사용되며, 검출된 Pitching 신에 해당하는 프레임을 <그림 4>과 같이 온라인 모델 학습을 위한 입력으로 사용한다.

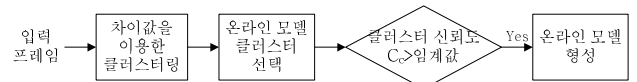


그림 4. 온라인 모델 학습 방법

온라인 모델을 생성하기 위해 먼저 입력 프레임과 이전 클러스터 간의 차이값을 계산한다. 이전 클러스터들에 대해 계산된 차이값이 특정 임계값보다 작은 클러스터가 존재할 경우 새로운 클러스터를 생성하고, 그렇지 않은 경우 해당 클러스터에 입력 프레임을 추가시킨다. 해당 프레임과 이전 생성된 클러스터들의 차이값(D_c)은 36개의 bin으로 HSV Color Histogram[8]의 유클리드 거리를 이용하였고, 이러한 클러스터링은 다른 방법보다 간단한 작업으로, Pitching 신의 해당 프레임이 입력되는 순간 바로 수행 가능하다.

클러스터링 과정 후, 적합한 클러스터 선별 작업이 수행된다. 본 논문에서는 가장 많은 프레임들로 구성된 클러스터를 선별하였으며, 이러한 선별 기준은 Linear SVM 분류기의 성능과 경기 내에 속하는 Pitching 신 특징 벡터의 유사성을 근거로 하였다. 클러스터 선택 후, within cluster scatter[9]를 이용하여 모델 생성에 대한 해당 클러스터의 신뢰도(C_c)를 계산한다. <그림 3>의 온라인 모델의 신뢰도 C_n 또한 within cluster scatter 값을 나타낸다. 이렇게 선택된 클러스터에 속하는 프레임들의 ECH[7]와 HSV Color Histogram [8]을 이용하여 온라인 모델을 생성한다. 생성된 온라인 모델(M_p)은 (M_e, M_c)로 표현이 가능하고, $M_e = \{M_e^0, \dots, M_e^{79}\}$ 이고 $M_c = \{M_c^0, \dots, M_c^{35}\}$ 으로써 다음 <식1>와 같이 나타낸다.

$$M_e^i = \frac{\sum_{t \in S_p} e_t(i)}{N(S_p)}, \quad M_c^i = \frac{\sum_{t \in S_p} c_t(i)}{N(S_p)} \quad \text{<식1>}$$

여기서, M_e^i 은 ECH 모델의 i 번째 값이며, M_c^i 은 HSV Color Histogram 모델의 i 번째 값을 나타낸다. 또한, S_p 은 온라인 모델 클러스터 집합을 나타내며, $e_t(i)$, $c_t(i)$ 는 각 t 오브젝트에서 추출된 ECH와 HSV Color Histogram의 i 번째 bin의 값을 나타낸다.

생성된 온라인 모델은 모델 신뢰도 C_n 이 특정 임계값보다 클 때 Pitching 신 검출에 적용하게 된다. 온라인 모델을 이용한 Pitching 신 검출은 ECH와 HSV Color Histogram간의 가중치 유클리드 거리를 사용하여 계산하고, 계산된 거리값이 특정 임계값보다 작은 경우 Pitching 신이라고 결정하게 된다. 이렇게 검출된 Pitching 신의 해당

프레임들은 <식2>의 기울기 하강 방법을 통해 온라인 모델 업데이트한다.

$$M_e^i = M_e^i + \gamma F(M_e^i, I_e^i), M_c^i = M_c^i + \gamma F(M_c^i, I_c^i); F(a, b) = (a - b) \quad \text{<식2>}$$

여기서, I_e^i 와 I_c^i 는 입력되는 프레임의 i 번째 ECH 값과 HSV Color Histogram 값을 나타낸다. Pitching 신이 검출이 완료되면 다음으로는 Close Up 신을 검출하는 작업이 수행된다. 야구 경기의 Close Up 신에서 필드 영역이 적게 나타나는 특징을 가지므로 필드 색을 이용하여 Close Up 신을 검출한다. 필드 색을 추출하는 과정은 <그림 5>과 같으며, Play 구간의 시작인 이전 Pitching 신의 대표 프레임을 이용한다.

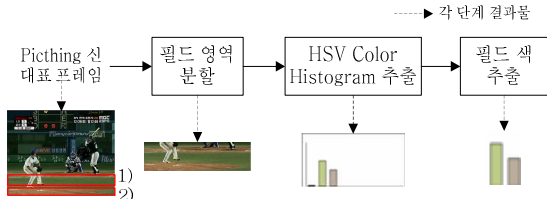


그림 5. 필드 색 추출 방법

Pitching 신 대표 프레임에서 필드 색에 속하는 1)의 잔디 영역, 2)의 흙 영역이 프레임 하단에 많이 나타나는 것을 볼 수 있으며, 이러한 프레임 특징을 이용하여 필드 색을 추출하게 된다. 먼저 입력 프레임에서 특정 크기의 하단 영역만 분할한 뒤, 분할된 하단 영역에서 HSV Color Histogram[8]을 추출하게 된다. 추출된 HSV Color Histogram의 12번째부터 23번째 빈 중에서 최대값 상위 두 개의 특징 정보를 선별한 후, 선별된 두 개의 특징 정보 평균을 계산하여 필드 색이라고 결정한다. 추출된 필드 색을 이용하여 Close Up 신을 검출하게 되며, <그림 6>은 Close Up 신 검출 알고리즘을 나타내는 그림이다.

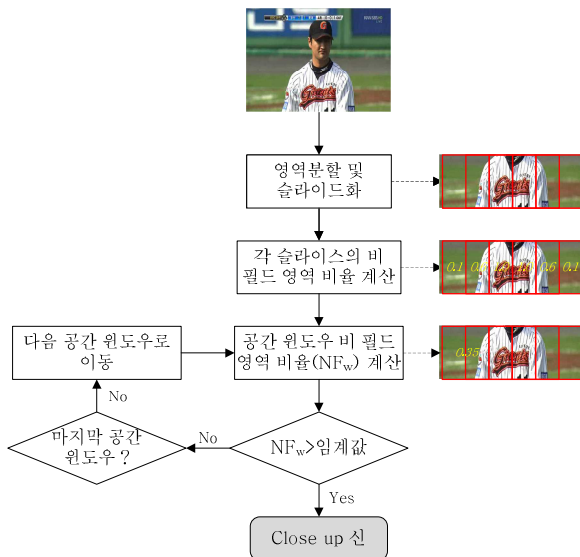


그림 6. Close Up 신 검출 방법

입력되는 프레임에서 특정 크기의 하단 영역만 분할한 뒤, 여러 개의 슬라이드로 나눈다. 나눈 각각의 슬라이스에서 <식3>를 이용하여 비 필드 영역 비율을 계산한다.

$$f \in S_n, \text{ if } D(f, F_m) \geq \epsilon_f, \quad \text{<식3>}$$

$$\text{where } D(f, F_m) = \frac{|f_s - F_m^s| + |f_v - F_m^v|}{2} + (1 - \cos|f_h - F_m^h|)$$

<식3>에서 F_m^h, F_m^s, F_m^v 는 필드 색 추출 방법에서 계산된 필드 색의

h, s, v 채널 값을 나타내고, f_h, f_s, f_v 는 각 입력 픽셀의 h, s, v 채널 값을 나타낸다. 또한 S_n 은 필드 픽셀의 집합을 나타내고, ϵ_f 는 비교 임계값을 나타낸다. HSV 색상 공간의 경우, RGB 색상 공간보다 사람의 시각 구조와 비슷한 특성을 지니므로 필드 픽셀 추출에 적합하다.

각 슬라이스의 비 필드 영역 비율을 계산하고, 계산된 비 필드 영역 비율을 이용하여 공간 윈도우 내 비 필드 영역 비율을 계산한다. 공간 윈도우는 전체 가로 크기의 30% 크기로 여러 개의 슬라이드로 구성된다. 공간 윈도우 내 비 필드 비율이 일정 임계값보다 큰 경우 해당 신을 Close Up 신이라고 결정하고, 일정 임계값보다 작은 경우에는 공간 윈도우는 한 슬라이스만큼 오른쪽으로 이동한 영역에 대해 비 필드 비율을 계산한다. 이러한 과정은 모든 공간 윈도우에 대해 수행하며, 해당 프레임의 Close Up 신 여부를 결정짓게 된다.

3.2. 중요 이벤트 구간 검출 방법

중요 이벤트가 발생한 Play 구간을 검출하기 위하여 본 논문에서는 오디오 이벤트 지점과 Play 구간의 길이를 이용한다. Play 구간 길이의 경우, 앞에서 설명한 Play 구간 검출 방법을 이용하여 계산할 수 있으므로, 본 절에서는 오디오 이벤트 지점 검출 방법에 대해 설명하도록 한다.

<그림 7>는 오디오 이벤트 검출 과정을 설명하는 그림이다. 먼저 오디오 데이터(24ms)가 입력되면 특징 정보 추출 단위로 프레임 블록킹이 수행되고, 각 오디오 프레임에 대하여 MFCC(Mel Frequency Cepstrum Coefficients), STE(Short Time Energy), 주파수 영역 특징 정보(Spectral Flux, Spectral Rolloff, Spectral Centroid) 등의 오디오 특징 정보를 추출한다. 추출된 특징 정보들은 0.96초의 세그먼트 단위로 누적되고, 누적된 특징 정보는 STE를 이용하여 묵음 구간을 필터링한다. 묵음 구간이 아닌 특징 정보는 SVM분류기를 통해 흥분(Exciting) 구간과 비 흥분(Others) 구간으로 분류하고, 마지막으로 흥분 구간에 대해서 후 처리 과정을 수행하게 된다. 후처리 과정에서는 야구 경기의 오디오 특성에 맞게 몇 초간 지속되는 흥분 구간만 선택하여 최종 오디오 이벤트 구간으로 선별하게 된다. 본 알고리즘은 입력되는 오디오 데이터에서 최종 오디오 이벤트 구간 판별까지 빠른 시간 내에 수행되므로 오디오 이벤트 검출 과정 또한 실시간 검출이 가능하다.

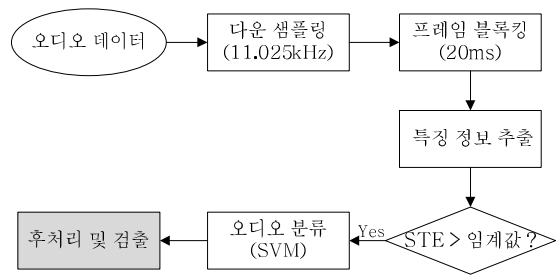


그림 7. 오디오 이벤트 검출 과정

오디오 이벤트 구간 검출과 3.1절에서 설명된 Play 구간 검출이 수행된 이후, 검출된 두 가지 정보를 이용하여 실제 중요 이벤트 구간을 정의하게 된다. 먼저, 오디오 정보를 이용하여 검출되는 이벤트 구간을 먼저 중요 이벤트 후보 구간이라고 정의하고, 검출된 Play 구간 안에 발생한 중요 이벤트 후보 구간만을 필터링하여 실제 중요 이벤트 구간으로 선택한다. 해당 Play 구간의 길이가 비교적 긴 경우 앞에서 언급했듯이, 안타나 홈런 등 흥미로운 이벤트가 발생되므로, 오디오 이벤트가 검출된 Play 구간 내에 존재하는 경우 해당 Play구간을 실제

중요 이벤트라고 결정짓게 되는 것이다.

4. 실험 및 분석

본 논문에서 제안하는 알고리즘의 효용성을 증명하기 위하여 야구 동영상 10경기에 대하여 실험을 수행하였다.

서론에서 언급한 본 논문에서 해결하고자 하는 요구 사항은 즉시성, 정확성, 지역성이다. 지역성의 경우, 제안하는 알고리즘은 재생 도중에 중요 이벤트가 검출되기 때문에 요구 사항을 만족시킨다.

즉시성의 요구 사항을 만족시키기 위해서는 동영상을 재생하면서 이벤트가 발생하는 즉시, 이벤트 검출 과정도 수행 완료되어야 하기 때문에 단위 시간의 동영상 데이터 처리 시간이 중요하다. 본 논문에서는 이를 증명하기 위하여 1초의 동영상 데이터에서 제안된 알고리즘을 수행하는 소요 시간을 측정하였으며, 소요시간은 총 0.024초였다. 이로써, 제안한 알고리즘은 동영상 재생 도중에 수행이 가능하다.

정확성의 요구 사항을 만족시키기 위해 먼저 Play 구간 검출 알고리즘의 정확도에 대한 실험을 수행하였으며, 실험결과는 <표 1>과 같다. 표에서 사용한 세 가지의 성능 측정 지표는 <식4>와 같다.

$$Recall = \frac{N_c}{N_c + N_m}, Precision = \frac{N_c}{N_c + N_f}, Accuracy = \frac{P_c}{P_t} \quad <식4>$$

여기서 N_c 는 정확하게 찾은 Play 시작 지점의 개수, N_m 은 검출이 되지 않은 Play 시작 지점의 개수이며 N_f 는 잘못 검출된 Play 시작 지점의 개수를 나타낸다. *Accuracy* 지표에서 사용된 P_c 와 P_t 는 각 Play 끝 지점이 정확하게 검출된 Play 구간의 개수와 총 Play 구간의 개수를 나타낸다. Play 끝 지점의 경우 Play 시작 지점이 정확하게 검출된 Play 구간에 대해서만 테스트를 수행하기 때문에, Play 시작 지점과는 다른 지표를 사용하고 있는 것이다.

표 1. Play 구간 검출 실험 결과

비디오	플레이 시작 지점		플레이 끝 지점
	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
STL vs. FLO	0.994	0.994	0.949
SS vs. HH	0.994	1.000	1.000
Bos vs. NY	0.963	1.000	0.924
SS vs. DS 1	0.993	1.000	0.967
DS vs. HH	0.997	0.986	0.979
SS vs. KT	1.000	0.982	0.966
HD vs. HH	1.000	0.944	0.978
SK vs. LT	1.000	0.971	0.886
KoreanSeries1	1.000	1.000	0.984
KoreanSeries7	0.969	1.000	1.000
총 평균	0.991	0.988	0.963

표를 보면 알 수 있듯이, 제안한 알고리즘은 메이저리그(STL vs. FLO, Bos vs. NY)나 국내 경기 모두에서 *Recall*은 0.991, *Precision*은 0.988, *Accuracy*는 0.963으로 높은 성능을 보임을 알 수 있고, 제안한 알고리즘의 Play 구간 검출은 다양한 환경에 영향을 받지 않음을 알 수 있다. 다음으로 정확성 요구사항을 만족을 위한 중요 이벤트 검출 알고리즘의 정확도에 실험을 수행하였다. 성능 지표는 변형된 *Recall*, *Precision*을 사용하였으며, 여기서 *Recall*은 총득점 이벤트 중 정확하게 검출된 득점 이벤트의 비이고, *Precision*은 검출한 이벤트들 중에서 중요 이벤트 집합에 정의된 이벤트의 비율을 나타낸다. 성능 측정결과, 중요 이벤트 검출 성능은 *Recall*이 0.89(141점/158점), *Precision*이

0.975(794점/815점)을 나타냈다.

Play 구간 검출과 중요 이벤트 검출의 실험을 통해 제안하는 알고리즘은 정확성의 요구 사항을 만족시킴을 알 수 있었다. 특히 *Precision*의 성능이 높기 때문에 타임 시프트와 같은 응용 분야에서 알고리즘을 통해 검출된 이벤트의 위치로 이동하는 경우, 중요 이벤트 일 가능성이 높아져서 사용자의 만족도를 높일 수 있다.

5. 결론

본 논문에서는 실시간으로 야구 동영상내의 중요 이벤트를 검출하는 알고리즘을 제안하였다. Play 구간 검출을 위해 함께 사용된 온라인 모델과 오프라인 모델은 다양한 경기 환경에서 적용하여 높은 성능을 보였으며, 멀티 모달 정보를 이용하여 중요 이벤트 구간 검출의 정확도를 높일 수 있었다. 실험 결과, 본 논문에서 제안하는 알고리즘은 1초의 동영상 데이터를 처리하는데 0.024초의 소요 시간이 필요하고, 0.89의 *Recall* 과 0.975의 *Precision* 성능을 보임을 알 수 있었다. 제안하는 알고리즘은 타임 시프트와 같이 실시간 처리가 필요한 응용 분야에 효율적으로 적용 가능할 것으로 기대된다.

6. 참고 문헌

- [1] L. Baoxin, and M. I. Sezan, "Event Detection and Summarization in Sports Video," *IEEE Workshop on Content Based Access of Image and Video Libraries*, pp.132-138, 2001.
- [2] P. Chang, M. Han, and Y. Gong, "Extract Highlights from Baseball Game Video with Hidden Markov Models," *Proceedings of International Conference on Image Processing*, pp.609-612, 2002.
- [3] J. Wu, X. Hua, J. Li, B. Zhang, and H. J. Zhang, "An Online Learning Framework for Sports Video View Classification," *Proceedings of Pacific Rim Conference on Multimedia*, pp.289-297, 2004.
- [4] I.Otsuka, R.Radharkishnan, M.Siracusa, A.Divakaran, and H.Mishima, "An Enhanced Video Summarization System Using Audio Features for a Personal Video Recorder," *IEEE Transactions on Consumer Electronics*, pp.168-172, 2006.
- [5] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic Soccer Video Analysis and Summarization," *IEEE Transactions on Image Analysis*, pp.796-807, 2003.
- [6] D. A. Sadlier, and N. E. O'Connor, "Event detection in field sports video using audio-visual features and a support vector Machine," *IEEE Transactions on Circuits and Systems for Video Technology*, pp.1225-1233, 2005.
- [7] A. Yamada, M. Pickering, S. Jeannin, L. Cieplinski, and Jens, MPEG-7 Visual Part of experimentation Model Version 9.0, ISO/IEC JTC1/SC29/WG11/N3914, 2001.
- [8] L. Zhang, F. Lin, and B. Zhang, "A CBIR Method Based on Color-Spatial Feature," *Proceedings of IEEE Region 10 Conference*, pp.166-16, 1999.
- [9] D. L. Davies, and D. W. Bouldin, "A cluster Separation Measure," *IEEE Transactions on Pattern Recognition and Machine*, pp.224-227, 1979.