

# 오디오의 Peak 특징을 이용한 동일 영화 콘텐츠 검색

정명범<sup>†</sup>, 성보경<sup>\*\*</sup>, 고일주<sup>\*\*\*</sup>

## 요 약

검색을 위해 동영상 데이터 전체를 이용하면 많은 시간과 저장 공간이 필요하다. 이를 보완하고자 기존의 동일 영화 검색은 영상 정보의 일부를 이용하여 동일한 영상 검색에 사용해 왔다. 그러나 이 방법은 같은 영상임에도 비디오 부호화기이나 해상도가 다른 경우 전혀 다른 영상으로 인식한다. 따라서 본 논문에서는 동영상의 오디오 정보를 이용하여 동일한 동영상을 찾는 알고리즘을 제안한다. 제안 방법은 부호화율, 부호화기, 샘플링 수의 변화에도 유사한 파형을 형성하는 Peak 정보를 바탕으로 데이터베이스에 색인하고, 검색한다. 논문에서는 제안 방법의 성능을 확인하기 위해 1,000개의 동영상 데이터를 검색 실험하였으며, 92.1%의 성공률을 나타내었다.

## Similar Movie Contents Retrieval Using Peak Features from Audio

Myoung-Bum Chung<sup>†</sup>, Bo-Kyung Sung<sup>\*\*</sup>, Il-Ju Ko<sup>\*\*\*</sup>

## ABSTRACT

Combing through entire video files for the purpose of recognizing and retrieving matching movies requires much time and memory space. Instead, most current similar movie-matching methods choose to analyze only a part of each movie's video-image information. Yet, these methods still share a critical problem of erroneously recognizing as being different matching videos that have been altered only in resolution or converted merely with a different codecs. This paper proposes an audio-information-based search algorithm by which similar movies can be identified. The proposed method prepares and searches through a database of movie's spectral peak information that remains relatively steady even with changes in the bit-rate, codecs, or sample-rate. The method showed a 92.1% search success rate, given a set of 1,000 video files whose audio-bit-rate had been altered or were purposefully written in a different codec.

**Key words:** Movie Retrieval(동영상 검색), Movie DNA(동영상 DNA), Audio Signal Processing(오디오 신호처리), Audio Feature Extraction(오디오 특징 추출)

## 1. 서 론

최근 컴퓨터 환경의 발달과 폭 넓은 인터넷의 확산으로 사용자들은 보다 풍부하고 깊이 있는 정보를 접할 수 있게 되었으며, 정보의 내용 또한 이미지,

오디오, 비디오 등 다양한 멀티미디어 데이터 형식으로 제공되게 되었다. 따라서 멀티미디어 데이터에 대한 사용자들의 검색 요구도 증가하였으며, 이를 위한 멀티미디어 검색 시스템에 대한 연구가 활발히 진행되고 있다. 초기의 검색 시스템은 데이터가 가지고

※ 교신저자(Corresponding Author) : 정명범, 주소 : 서울시 동작구 상도동(156-743), 전화 : 010-3125-1061, FAX : 02)882-1064, E-mail : nzin@ssu.ac.kr

접수일 : 2009년 5월 26일, 수정일 : 2009년 8월 3일

완료일 : 2009년 8월 20일

<sup>†</sup> 정희원, 숭실대학교 미디어학과 박사과정

<sup>\*\*</sup> 정희원, 숭실대학교 미디어학과 박사과정  
(E-mail : ivsinger@ssu.ac.kr)

<sup>\*\*\*</sup> 정희원, 숭실대학교 미디어학부 조교수  
(E-mail : andy@ssu.ac.kr)

※ 본 연구는 숭실대학교 교내연구비 지원으로 이루어졌음

있는 특징 그대로 시스템에 색인되어 저장되는 것이 아니라, 기존 분류되어 있는 틀에 맞추어 수작업을 통해 각각의 제목이나, 내용에 맞게 텍스트 기반으로 데이터베이스에 저장되어 왔다. 그러나 이러한 수작업의 색인은 점차 늘어나는 정보의 양을 따라가기에는 역부족이었으며, 이를 자동화하기 위한 방법으로 내용 기반의 정보 검색 시스템에 대한 연구가 진행되었다. 내용 기반의 정보 검색 시스템은 정보의 내용을 수학적으로 분석하여 구조화된 기준에 따라 대표적인 특징을 추출하고, 그러한 특징을 토대로 데이터를 체계적인 구조로 색인화 하는 것이다. 이러한 색인화는 멀티미디어 데이터의 신속하고 정확한 정보와 특징을 얻어낼 수 있고, 데이터 고유의 특징으로부터 사용자에게 적합한 내용을 제공할 수 있다.

동영상 검색 분야에서도 위와 같은 흐름을 따라왔다. 초기의 동영상 검색은 파일명, 파일크기, 워터마크 등을 이용한 동영상의 기본 정보만을 이용하거나, 사람의 수작업에 의해 내용 정보를 요약하여 색인해왔다. 그리고 점차 하드웨어의 처리 속도가 향상되고, 영상처리 기술이 발달됨에 따라 색인과 검색을 자동화하기 위해 동영상의 영상 정보를 이용한 내용 기반 검색 기술이 활성화 되었다. 내용 기반 검색 기술에 사용되는 영상 데이터의 특징은 기존 이미지 검색에 주로 사용 되어온 색상, 모양, 질감 등을 이용하였다[1-5]. 그리고 검색 속도를 향상시키기 위해 영상 데이터 전체를 사용하는 것이 아닌, 장면 전환점 검출, 동영상 색인 기법 등을 이용하여 영상 데이터의 일부분만을 이용하였다[6-8].

그러나 이러한 동영상의 영상을 기반으로 한 내용 기반 검색 기술은 동영상 인코딩 시 부호화기가 바뀌거나, 해상도(Resolution)가 변화하는 경우 그림 1, 그림 2와 같이 히스토그램의 형태는 비슷하나, 값이 변하여 다른 영상으로 오인식될 가능성이 있다. 그림 1은 무압축 해상도 640×480인 동영상에서 가져온 영상이며, 그림 2는 Mpeg-4 부호화기에 해상도 320×240인 동영상에서 가져온 동일 프레임의 히스토그램이다. 게다가 같은 해상도를 갖는 동일 동영상에서도 부호화기가 바뀐 경우 그림 3, 그림 4와 같이 장면 전환점이 달리 나타남을 확인할 수 있다. 그림 3은 DivX MPEG-4 Video v4 (OpenDivX) 부호화기를 사용한 것이며, 그림 4는 Microsoft MPEG-4 Video v2 부호화기를 사용한 것으로 부호화기만 달

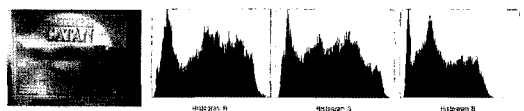


그림 1. 무압축, 640×480 해상도 영상의 히스토그램

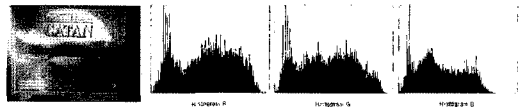


그림 2. Mpeg-4 부호화기, 320×240 해상도 영상의 히스토그램

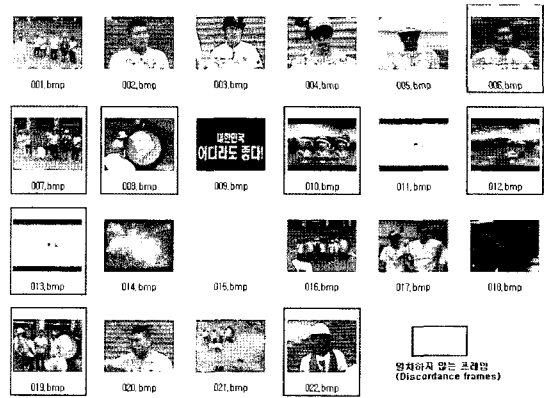


그림 3. 부호화기 DivX MPEG-4 Video v4 (OpenDivX), 해상도 320×240 pixel

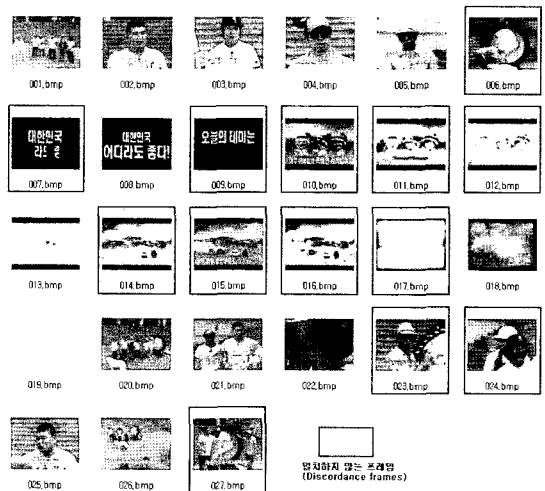


그림 4. 부호화기 Microsoft MPEG-4 Video v2, 해상도 320×240 pixel

리 사용하고, 해상도(320×240 pixel), 부호화율(768 kbps), 프레임 비율(29.97 frame/s)은 동일하게 한 것

이다. 이때, 장면 전환점 검출은 동영상 색인 기법에서 대표적으로 사용하는 Color histogram과 Texture를 이용 하였다.

따라서 본 논문에서는 동영상의 부호화기나 해상도 변화에도 크게 영향을 받지 않는 오디오 정보를 이용한 내용 기반 검색 알고리즘을 제안한다. 오디오 정보 또한 부호화기나 부호화율에 따라 수치가 변화하기 때문에 기존 오디오 신호 처리 분야에서 사용하는 FFT(Fast Fourier Transform), MFCC(Mel-frequency Cepstral Coefficients), LPC(Linear Predictive Coding) 등을 적용하기에는 무리가 있다[9-11]. 그러나 오디오 데이터의 Peak 정보는 부호화기, 부호화율의 변화에도 유사한 파형을 형성하므로[12], 오디오 데이터의 Peak 정보로부터 구분력(unique) 있는 특징을 추출하면 동영상 검색에 효과적으로 활용 할 수 있다. 논문에서는 Peak 정보로부터 Low-peak라는 특징 색인 값을 추출하였으며, 값의 양에 따라 Level을 분류하여 검색에 사용하였다. 검색을 위한 동영상 데이터는 저작권 보호에 큰 이슈가 되고 있는 드라마, 쇼프로, 영화 등 1,000개의 파일을 사용하였으며, 부호화율과 부호화기의 변화에 대한 검색 수행 시 효과적인 결과를 얻을 수 있었다.

논문의 구성은 다음과 같다. 2장에서는 동영상의 검색에 대한 기존 연구 방법인 영상 데이터를 이용하는 방법과 오디오 신호 처리 분야에서 사용하는 FFT, MFCC 등에 대해 간략히 설명한다. 3장에서는 본 논문에서 제안하는 방법인 Low-peak 특징 추출에 관한 알고리즘을 설명한다. 그리고 4장에서는 제안 알고리즘의 성능 평가에 대한 실험과 분석을 하며, 5장에서는 결론을 제시한다.

## 2. 관련 연구

기존의 영상 데이터를 이용한 동영상 검색 방법은 전체 영상의 시퀀스 중 중요한 프레임(Key Frame)만을 추출하여 검색하는 기법이 주로 사용된다. 이 프레임은 영상들 간의 샷(Shot) 내에서 가장 효과적으로 대표할 수 있는 장면을 말하며, 이를 추출하기 위해 전체 영상 시퀀스들의 장면 전환점을 검색하여 비디오 분할을 수행하게 된다[13-15]. 이때 샷은 그림 5와 같이 하나의 카메라 동작이 끝나고 다른 카메라로 넘어가기까지의 프레임들의 집합을 의미하며,

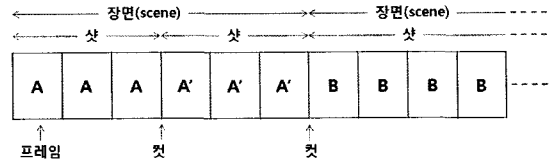


그림 5. 동영상 데이터의 영상 스퀀시 구조

샷 내부에 있는 인접한 프레임들은 높은 유사성을 갖는다.

장면 전환점 검출은 많은 연구자들에 의해 다양한 방식들이 제안 되었다. 단일 특징을 이용한 것으로는 무압축 영상의 경우 인접 프레임의 변화량을 이용하여 픽셀 값 차이를 비교하는 방법, 히스토그램 차이를 비교하는 방법, 에지를 추출하여 비교하는 방법, 인접한 블록 사이의 유사도를 비교하는 방법 등이 있으며, 압축 영상의 경우 DCT(Discrete Cosine Transform) 계수나 모션 벡터를 이용하여 공간-주파수 도메인을 사용하는 방법이 있다[16,17]. 그러나 이 같은 기존 기법들은 단일 특징만을 이용하기 때문에 특정 상황에서만 뛰어난 성능을 발휘하고 여러 상황이 포함된 비디오 시퀀스에서는 적용하기 어려운 문제점이 있다. 예를 들어 화소 단위 비교는 뉴스와 같이 고정된 배경에 카메라 움직임이 거의 없는 영상에서 좋은 결과를 보이는 반면, 스포츠 영상과 같이 객체나 카메라의 움직임이 큰 영상에서 성능이 급격히 저하된다. 또한 히스토그램 비교에서는 스포츠 영상에 좋은 성능을 보이는 반면, 뉴스와 같은 변화가 크지 않은 영상에서는 성능이 떨어진다.

이와 같은 기존의 문제점을 해결하기 위해 단일 특징을 혼합한 향상된 장면 전환점 검출 방식들이 제안 되었다. Yusoff는 다섯 가지의 장면 전환점 검출 방식을 이용하여 ROC(Receiver Operating Characteristic) 곡선과 투표 기법으로 장면 전환점을 검출하였다 [18]. Naphade와 Lee 등은 픽셀 차와 히스토그램 차를 이용하여 K-means 알고리즘으로 장면 전환점을 분류 하였으며, Dugad는 히스토그램 기법과 Likelihood Ratio를 이용하여 두 단계로 임계값을 설정하고 장면 전환점을 분류 하였다[19-21].

장면 전환점 검출 방법들 중 주로 사용되는 방법인 히스토그램 기반 검출법은 각 프레임에서 밝기 혹은 컬러의 분포를 구한 후 인접한 두 개의 히스토그램 분포를 비교하는 방식이다. 이 방법은 객체의 빠른 이동이나, 카메라의 이동, 회전 등에 안정된 성

능을 나타낸다. 그러나 각 프레임의 밝기나 컬러 분포만을 이용하기 때문에 위치 정보를 알 수 없는 단점이 있다. 즉, 뉴스와 같이 배경 변화가 많지 않은 영상의 경우 두 프레임 사이에 히스토그램 분포가 비슷하기 때문에 장면 전환점을 놓치는 경우가 발생하기도 한다. 이를 해결하기 위해 전체 프레임을 r개의 균등한 블록으로 나누고 각 블록에 대응하는 인접 프레임의 히스토그램 분포를 비교하여 유사성을 구하는 국부 히스토그램 방식을 사용하기도 한다. 전역 히스토그램( $HD_m$ )을 구하는 계산식은 수식 (2.1)과 같다.

$$HD_m = \frac{1}{(2MN)} \sum_{k=0}^B |H_{m+1}(k) - H_m(k)| \quad (2.1)$$

여기서  $M, N$ 은 영상의 가로와 세로의 크기,  $H_m(k)$ 는  $m$ 번째 프레임의 히스토그램을 나타내며,  $B$ 는 최대 밝기 값이다. 그리고 컬러 히스토그램을 이용할 경우에는 각 RGB별 히스토그램을 구성하여 차이를 계산한다. 다음으로 국부 히스토그램을 계산하는 식은 수식 (2.2), (2.3)과 같다.

$$LHD_j = \frac{1}{2|BS|} \sum_{k=0}^B |H_{m+1}(k, j) - H_m(k, j)| \quad (0 \leq j \leq 16) \quad (2.2)$$

$$LHD_m = \frac{1}{8} \sum_{i=0}^7 \min[LHD_i] \quad (2.3)$$

$H_m(k, j)$ 는 16등분으로 분할한 각 블록의 히스토그램 값,  $|BS|$ 는 블록내의 픽셀 수,  $LHD_j$ 는 각 분할된 블록의 히스토그램 차를 나타내며,  $LHD_m$ 은 유사도가 낮은 8개의 블록을 사용하여 히스토그램의 평균을 구한 것으로 이 값이 국부 히스토그램 값이다. 각 연속되는 프레임간의  $LHD_m$  평균을 비교하여 특징 임계값을 넘어서는 위치를 장면 전환점으로 검출하는 것이 국부 히스토그램을 이용한 방법이다.

그러나 장면 전환점 검출은 부호화기가 바뀌거나, 해상도가 변화하는 경우 동일 지점을 찾지 못하는 경우가 많기 때문에 본 논문에서는 동영상의 오디오 데이터로부터 특징을 검출하는 방법을 사용한다. 기존의 오디오 신호 처리 분야에서 사용되는 특징 벡터는 FFT와 MFCC가 있으며, 이 특징들은 음성 인식, 오디오 검색 등에 쓰인다. FT(Fourier Transform)는 다양한 파형으로 이루어진 오디오 신호를 주파수 공간으로 변형한 것이며, 이러한 변형을 위한 연산 시



그림 6. MFCC 추출 과정

간문제를 해결하기 위해 고안된 변환이 FFT이다. FFT로부터 얻을 수 있는 특징은 스펙트럼의 중심을 측정하는 Spectral Centroid, 스펙트럼의 형태와 낮은 주파수 영역의 분포를 구하는 Spectral Rolloff, 연속된 스펙트럼의 변화량을 알 수 있는 Spectral Flux 등이 있다.

MFCC도 FFT를 이용한 것으로, 인간의 청각 특성을 모델링 한 방법이다. 사람의 귀는 낮은 주파수에서 작은 변화에도 민감한 반면, 높은 주파수로 갈수록 민감도가 작아지는 특성이 있다. 이러한 사람의 귀의 반응은 로그 스케일과 비슷하므로, MFCC는 그와 비슷한 멜(Mel) 스케일을 이용하여 켈스트럼 계수를 추출한 것이다. MFCC를 얻는 방법은 오디오 신호의 크기 스펙트럼을 로그 스케일(log Scale)한 후 FFT Bin을 그룹화 하여 멜 주파수(Mel-Frequency) 스케일로 변환하여 얻을 수 있다. 그림 6은 MFCC를 추출하는 일반적인 과정을 나타낸다.

### 3. 오디오의 Low-peak 특징 추출

동영상이 가지고 있는 오디오 정보는 PCM 데이터로 변환 할 수 있으며, 오디오 신호 처리 분야에서는 이 데이터로부터 FFT 변환이나, MFCC 등을 이용하여 특징을 추출한다. 그러나 이 특징들은 부호화와 부호화기에 따라 그 값이 변경되므로, 같은 소리로 들리는 오디오 데이터임에도 다른 것으로 인식된다. 반면에 오디오의 Peak 정보는 부호화와 부호화기의 변화에도 유사한 파형을 형성한다. 그림 7은 부호화기 mp3, 샘플링 비율 44.1khz로 고정된 상태에서 부호화율을 각각 192kbps, 128kbps, 64kbps로 달리한 것이다. 그리고 위로부터 3번째 파형은 부호화기를 mp3가 아닌, AAC로 다르게 한 동일한 동영상의 Peak 정보이며, 4개의 파형 모두가 유사한 파형을 나타내는 것을 볼 수 있다. 따라서 오디오의 Peak 정보를 바탕으로 Peak 특징을 추출한다.

일반적으로 동영상들은 초반부에 배경 음악을 제외한 여러 가지 소리들을 가지고 있으며, 대부분의 소리는 다양한 파형을 형성한다. 이러한 소리들의 Peak 정보를 보면, 그림 8과 같이 일정 임계값(Threshold)

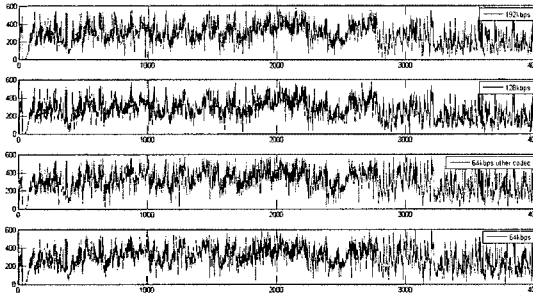


그림 7. 부호화율과 부호화기의 변화에 따른 Peak 정보

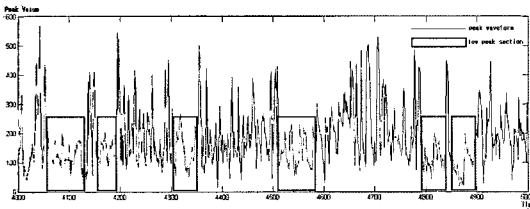


그림 8. Peak 정보의 low peak 구간

를 넘지 않는 low peak 구간들이 발생한다.

따라서 peak 정보로부터 고정된 간격으로 나누어 low peak들을 추출하면 그림 9과 같이 나타난다. 그림 9는 그림 8의 Peak를 일정 구간으로 나눈 후, 그 구간 내에 임계값보다 낮은 peak가 많은 때 0으로, 임계값보다 높은 peak가 많을 때 1을 갖도록 한 것이다. 이때 그림 9의 빨간 네모 영역과 같이 동일 동영상의 다른 부호화기, 다양한 부호화율에서도 공통된 low peak 구간을 찾는 것을 확인할 수 있다. 따라서 그림 9의 선택된 영역을 Low-peak 영역이라 할 수 있으며, Low-peak 정보를 데이터베이스에 색인한다. Low-peak 정보를 색인하는 방법은 다음의 순서를 따르며, Low-peak 검출을 위한 수식은 표 1과

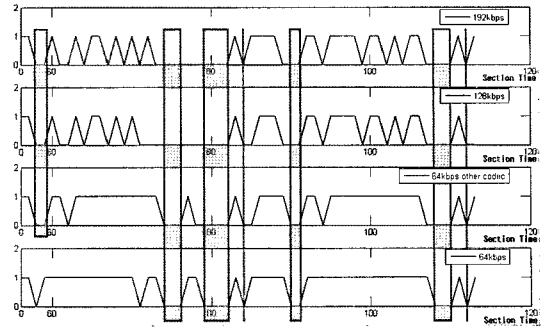


그림 9. Peak 정보를 이용한 Low-peak 영역 검출

같다.

- (1) 동영상 파일의 시작 부분으로부터 일정 길이를 선택하여 peak 정보 추출
- (2) peak 정보로부터 고정된 구간을 나누어 구간별 low peak 값 측정
- (3) 측정된 low peak 값으로부터 Low-peak 위치 결정

표 1에 의해 동영상의 Peak 정보로부터 Low-peak 영역을 데이터베이스에 색인할 수 있으나, 동영상의 오디오 정보들 중에 대화가 없거나, 파형의 정보가 큰 변화가 없는 경우 Low-peak 영역이 많이 발생할 수 있다. 따라서 이 경우까지 보완할 수 있게 Low-peak 영역이 일정치 이상인 경우 Level을 두어 임계값을 낮춘 후 2차 Low-peak 영역을 검출하여 데이터베이스에 색인한다. 2차 Low-peak 영역 검출을 위해 표 1에 표 2를 추가하여 적용한다.

이때 표 1에서 Low-peak 검출은 시간 영역을 색인하는 것이기 때문에 부호화율 변환 시 생기는 파형의 밀립이나 당겨지는 현상을 최소화 시켜야 한다. 이는 오디오 데이터의 최소 파형 모형을 이용하여

표 1. Low-peak 검출 과정

<p>(1) n개의 peak 데이터 :</p> $peak[n] = \max\{\{1764*(n-1)+1\}, \{1764*(n-1)+2\}, \dots, \{1764*(n-1)+1764\}\} - \min\{\{1764*(n-1)+1\}, \{1764*(n-1)+2\}, \dots, \{1764*(n-1)+1764\}\}$ <p>(2) <math>g[m] = \sum_{n=1}^i f(peak[n])</math>      <math>\begin{cases} f(peak[n]) = 1 (peak[n] &lt; \alpha) \\ f(peak[n]) = 0 (peak[n] \geq \alpha) \end{cases}</math></p> <p><math>i</math> : 고정 간격, <math>m</math> : 고정 간격에 의한 색인  <math>g[m]</math> : 고정 간격의 low peak 개수, <math>\alpha</math> : 임계값</p> <p>(3) <math>LC = \sum_{m=1}^j h(g[m])</math>      <math>\begin{cases} h(g[m]) = 1 (g[m] &gt; \beta) \\ h(g[m]) = 0 (g[m] \leq \beta) \end{cases}</math></p> <p><math>j</math> : 전체 길이/고정간격(<math>n/i</math>), <math>h(g[m])</math> : Low-peak 영역 결정 함수  <math>LC</math> : Low-peak 영역 값, <math>\beta</math> : 임계값</p>
--

표 2. 2차 Low-peak 검출을 위한 수식

$$(4) \begin{cases} \in dex : m, level & (LC < \gamma) \\ (\alpha - k, go(2), (3), (4) & (LC \geq \gamma) \end{cases}$$

$\gamma$  : 임계값

시작점을 검출함으로써 그 현상을 방지 할 수 있다[22]. 그림 10은 그림 7의 부호화율과 부호화기 변화에 대한 예시로부터 시작점을 검출한 것이다. 부호화율이나 부호화기에 의해 재 변환을 한 경우 그림 10에서처럼 시작점이 점점 밀려나감을 확인할 수 있다. 그림 10의 가장 위에 파형정보는 mp3, 44.1kHz, 192kbps의 오디오 데이터이며, 오디오의 소리가 처음으로 나타나는 시작점은 peak 위치 2640 프레임 부분이다. 반면 동일 동영상의 mp3, 44.1kHz, 128kbps는 3520 프레임, mp3, 44.1kHz, 64kbps는 5280 프레임, 다른 부호화기를 사용한 AAC, 44.1kHz, 64kbps는 4400 프레임로 부호화율, 부호화기 변화에 따라 시작점이 달라진다.

시작점 검출과 제안 방법으로부터 얻은 Low-peak 검출 값은 동영상 검색을 위한 데이터베이스 색인에 사용되며, 논문에서의 데이터베이스 색인 값은 mp3, 44.1kHz, 64kbps를 사용한다. 부호화기로 mp3를 사용하는 것은 업로드 파일의 대부분이 용량을 많이 차지하는 AC3, AAC보다 용량을 적게 차지하는 mp3를 주로 사용하기 때문이며, 샘플링 수 또한 대부분의 파일들이 표준으로 44.1kHz를 사용하기

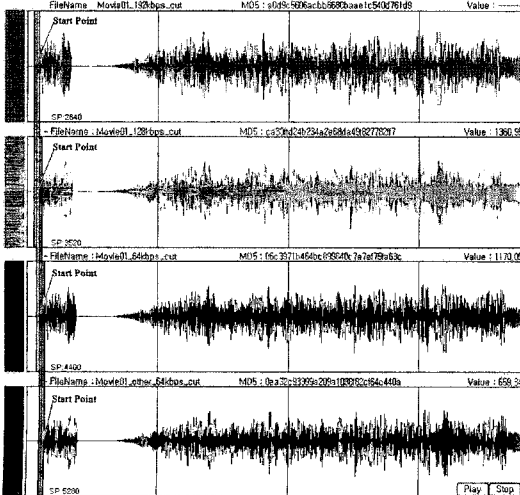


그림 10. 부호화율, 부호화기에 의한 시작점의 위치 변화

때문이다. 그리고 부호화율 64kbps 값은 128kbps, 192kbps에 비해 가장 작은 Low-peak 색인을 하기 때문에, 검색 질의문(query)을 생성 시 최소한의 값을 받기 위함이다.

데이터베이스의 색인된 동영상 데이터로부터 검색을 하기 위한 질의문은 동일한 Low-peak 검출 방법을 이용한다. 그러나 부호화율과 부호화기에 의해 peak 파형의 변화가 생겨 Low-peak 개수 값이 변할 수 있기 때문에 추출한 Level을 1차 Low-peak 값으로, 추출 Level의 한 단계 아래인 값을 2차 Low-peak 값으로 한 후 검색에 적용한다.

#### 4. 실험 및 분석

오디오 데이터의 Peak 특징을 이용한 동일 동영상 검색 실험은 다음과 같이 이루어졌다. 뮤직비디오, 드라마, TV 오락프로 등을 무작위로 1,000개의 동영상을 수집하였으며, 수집한 동영상의 오디오 정보로부터 제안한 방법으로 mp3, 44.1kHz, 64kbps 음질로부터 Peak 특징을 데이터베이스에 색인하였다. 그 후 색인한 데이터베이스로부터 검색을 위한 질의문은 Peak 특징이 구분력 있는 특징임을 확인하기 위해 먼저 동일한 64kbps 음질을 검색에 사용하였다. 표 3은 동일한 64kbps 음질 사용 시 나타난 결과이다.

표 3에서 동일한 음질에 대한 검색 결과는 100%의 정확성을 나타냈으며, 제안한 Peak 특징이 구분력 있는 특징임을 확인할 수 있다.

다음으로 부호화율, 부호화기, 샘플링 수가 다른 동영상에도 제안 방법의 유효성을 검증하기 위한 세 가지 실험을 하였다. 첫째 실험은 부호화기, 샘플링 수를 mp3, 44.1kHz로 고정하고 부호화율을 변화한 것으로, 질의문은 192kbps, 128kbps 음질을 갖는 동일한 동영상 각 1,000개를 사용하였다. 데이터베이스에 색인된 Low-peak 값이 질의문보다 작으므로, 데이터베이스에 색인된 Low-peak 값과 일치하거나, 일치하는 데이터가 없는 경우 일정량의 오차가 있는 데이터를 검색 결과로 나타나게 하였다. 질의문으로

표 3. 오디오의 Peak 특징의 구분력 유효성 실험 결과

	Correct	Error	No Find	정확도
총계	1000	0	0	100 %

부터 얻은 결과의 일치 여부는 입력 데이터와 결과 데이터의 동영상 제목이 같은지를 비교 판단 하였다. 표 4는 부호화를 변화에 따른 제안 방법의 실험 결과이다.

두 번째 실험은 샘플링 수, 부호화 율을 44.1kHz, 64kbps로 고정하고 부호화기를 변화한 것으로, 질의 문은 MP4, AAC 부호화기로 인코딩한 동일한 동영상 각 1,000개를 사용하였다. 이때, AC3는 5.1 채널을 기반으로 한 입체음향 지원 포맷이므로 질의문 생성 시 2채널로 재인코딩을 하여 사용하였다. 표 5는 부호화기 변화에 따른 제안 방법의 실험 결과이다.

마지막 실험으로 부호화기, 부호화 율을 mp3, 64kbps로 고정하고 샘플링 수를 변화한 것으로, 질의 문은 22,050Hz로 샘플링한 동일한 동영상 1,000개를 사용하였다. 이 샘플링 수는 실제 동영상 파일, 오디오 파일에서는 주로 사용되지 않지만, 파일 용량을 줄이기 위한 방법으로 종종 사용되기 때문에 추가 실험을 한 것으로, 실험 결과는 표 6과 같다.

각각의 실험으로부터 부호화 율을 다르게 한 경우

표 4. 오디오의 Peak 특징을 이용한 부호화율 변화에 따른 동일 동영상 검색 실험 결과

	Correct	Error	No Find	정확도
mp3, 44.1kHz 192kbps Audio	904	95	1	90.4%
mp3, 44.1kHz, 128kbps Audio	900	99	1	90.0%
총 계	1804	194	2	90.2%

표 5. 오디오의 Peak 특징을 이용한 부호화기 변화에 따른 동일 동영상 검색 실험 결과

	Correct	Error	No Find	정확도
AC3, 44.1kHz 64kbps Audio	964	35	1	96.4%
AAC, 44.1kHz 64kbps Audio	937	62	1	93.7%
총 계	1901	97	2	95.1%

표 6. 오디오의 Peak 특징을 이용한 샘플링 수 변화에 따른 동일 동영상 검색 실험 결과

	Correct	Error	No Find	정확도
mp3, 22.05kHz 64kbps Audio	898	101	1	89.8 %

90.2%, 부호화기를 다르게 한 경우 95.1%, 샘플링 수를 변화한 경우 89.8%의 정확성을 나타내었다. 각각의 실험 결과를 비교하면 부호화기를 다르게 한 경우가 가장 높은 성공률을 보였는데, 이는 제안 방법이 부호화기에 대하여 부호화 율, 샘플링 수 등에 비해 강인하다는 것을 예상할 수 있다. 각각의 실험을 모두 종합하면 전체적으로 약 92.1%의 성능을 보였다. 이러한 성능은 저작권 보호를 위해 수작업으로 판단하는 현재의 시스템에 적용할 경우 기존 수작업 비용의 9/10를 절감 할 수 있다.

실험 결과에서 1개씩 No Find된 영상은 동일한 동영상으로, 데이터베이스에 색인된 Low-peak 개수와 질의문에 색인된 Low-peak 개수가 큰 차이를 나타내어 오차의 범위를 벗어났기 때문이다.

## 5. 결 론

본 논문에서는 동일 동영상 검색을 위한 오디오 정보를 이용한 알고리즘을 제안하였다. 동일 동영상 검색을 위해서는 해상도, 부호화기 변화에 따라 색인 지점이 달라지는 영상에 비해, 유사한 파형을 형성하는 오디오 데이터로부터 Peak 특징을 사용하는 것이 보다 효과적이다. Peak 특징을 데이터베이스에 색인하는 것은 데이터베이스 구축 시 기존의 영상 데이터를 저장하는 것에 비해 저장 공간을 절약할 수 있으며, 영상 검색을 위한 시간도 단축할 수 있을 것이다.

제안 알고리즘은 저작권 보호를 요청 받은 동영상 데이터들로부터 Peak 특징을 데이터베이스에 색인한 후, 사용자가 업로드한 동영상의 Peak 특징을 추출하여 비교함으로써 자동으로 서비스 유무를 판단하게 할 수 있다. 따라서 Youtube, E-Donky 등과 같이 동영상을 공유하는 서비스에서 저작권 보호를 위한 시스템 구축에 활용할 수 있을 것이다. 추후 연구로는 Error Movie에 대한 정확성을 높이고, No Find Movie의 색인을 위해 또 다른 Peak 특징 추출 연구가 필요하다.

## 참 고 문 헌

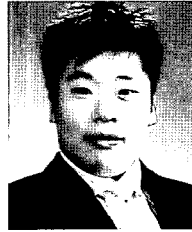
- [1] M.J. Swain and D.H. Ballard, "Color indexing," *International Journal of Computer Vision*, Vol.7, No.1, pp. 11-32, 1991.

- [2] L.M. Kaplan, R. Murenzi, and K.R. Namuduri, "Fast Texture Database Retrieval Using Extended Fractal Features," *SPIE proc. Storage and Retrieval for Image and Video Database*, Vol.3312, pp. 162-175, 1998.
- [3] J. Huang, S.R. Kumar, M. Mitra, W.J. Zhu, and R. Zabih, "Image Indexing Using Color Corrograms," *IEEE Proc. Computer Vision and Pattern Recognition*, pp. 744-749, 1997.
- [4] R. Mehrotra and J.E. Gray, "Similar-shape Retrieval in Shape Data Management," *IEEE Computer*, Vol.28, No.9, pp. 57-62, 1995.
- [5] B.S. Manjunath and W.Y. Ma, "Texture Features for Browsing and Retrieval of Image data," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.18, pp. 837-842, 1996.
- [6] S.W. Smoliar and H.J. Zhang, "Content-based Video Indexing and Retrieval," *IEEE Multimedia*, Vol.1, pp. 62-72, 1994.
- [7] W.A.C. Faernando, C.N. Canagarajah, and D.R. Bull, "Scene Change Detection Algorithms for Content-based Video Indexing and Retrieval," *Electronics & Communication Engineering Journal*, Vol.13, pp. 117-126, 2001.
- [8] J. Meng, Y. Juan, and S.F. Chang, "Scene Change Detection in a MPEG Compressed Video Sequence," *IS&T/SPIE Symposium Proceeding*, Vol.2419, pp. 14-25, 1995.
- [9] M.F. McKinney and J. Breebaart, "Features for Audio and Music Classification," *International Symposium on Music Information Retrieval*, pp. 151-158, 2003.
- [10] G. Tzanetakis and P. Cook, "Multifeature Audio Segmentation for Browsing and Annotation," *Workshop on Applications of Signal Processing to Audio and Acustics*, pp. 103-106, 1999.
- [11] G. Tzanetakis and P. Cook, "Musical Genre Classification of Audio Signal," *IEEE Transactions on Speech and Audio Processing*, Vol.10, No.5, pp. 293-302, 2002.
- [12] 정명범, 정보경, 고일주, "내용 기반 음악 검색의 문제점 해결을 위한 전처리," *한국 컴퓨터 정보 학회지*, Vol.12, pp. 97-104, 2007.
- [13] F. Idris and S. Panchanathan, "Review of Image and Video Indexing Techniques," *Journal of Visual Communication and Image Presentation*, Vol.8, No.2, pp. 146-166, 1997.
- [14] G. Lupatini, C. Saraceno, and R. Leonardi, "Scene Break Detection: a Comparison," *International Workshop on Continuous-Media Databases and Application*, pp. 34-41, 1998.
- [15] Y. Yusoff, W. Christmas, and J. Kitter, "A Study on Automatic Shot Change Detection," *European Conference on Multimedia Applications*, No.1425, pp. 177-189, 1998.
- [16] R. Zabih, J. Miller, and K. Mai, "A Feature-based Algorithm for Detecting and Classifying Scene Breaks," *ACM Conference on Multimedia*, pp. 189-200, 1995.
- [17] M.A. Stricker and M. Orengo, "Similarity of Color Images," *International Society for Optical Engineering*, Vol.2420, pp. 381-392, 1995.
- [18] Y. Yusoff, K. Kitter, and W. Christmas, "Combining Multiple Experts for Classifying Shot Changes in Video Sequences," *IEEE International Conference on Multimedia Computing and Systems*, Vol.2, pp. 700-704, 1999.
- [19] 이훈철, 고윤희, 윤병주, 김성대, 유상조, "클러스터링을 이용한 급격한 장면 전환 검출 기법," *대한전자공학회, 전자공학회논문지*, Vol.38, pp. 114-122, 2001.
- [20] M.R. Naphade, R. Mehrotra, A.M. Ferman, J. Warnick, T.S. Huang, and A.M. Tekalp, "A High-Performance Shot Boundary Detection Algorithm Using Multiple Cues," *International Conference on Image Processing*, Vol.1, pp. 4-7, 1998.
- [21] R. Dugad, K. Ratakonda, and N. Ahuja,



“Robust Video Shot Change Detection,” *IEEE Transactions on Multimedia Signal Processing*, pp. 376-381, 1998.

[22] 김정수, 성보경, 구광효, 고일주, “최소파형모형을 이용한 음악 시작점 검출 방법,” *한국 컴퓨터 정보학회 하계 학술발표 논문집*, Vol.16, pp. 137-141, 2008.



**성 보 경**

2006년 숭실대학교 미디어학부 졸업 (공학사)  
2008년 숭실대학교 미디어학과 졸업 (공학석사)  
2008년~현재 숭실대학교 미디어학과 박사과정  
관심분야 : 디지털 음악 분석, 음악 검색, 감성공학



**정 명 범**

2004년 숭실대학교 미디어학부 졸업 (공학사)  
2006년 숭실대학교 미디어학과 졸업 (공학석사)  
2006년~현재 숭실대학교 미디어학과 박사과정  
관심분야 : 디지털 신호처리, 감성 인식, 콘텐츠 공학



**고 일 주**

1992년 숭실대학교 전산학과 (공학사)  
1994년 숭실대학교 전산학과 (공학석사)  
1997년 숭실대학교 전산학과 (공학박사)  
2003년~현재 숭실대학교 미디어학부 조교수

관심분야 : 콘텐츠, 정보검색, 감성공학