한국음향학회지 제24권 제3호 pp. 178-185 (2005)

피치 히스토그램과 MFCC-VQ 동적 패턴을 사용한 음악 검색

Music Identification Using Pitch Histogram and MFCC-VQ Dynamic Pattern

박 철 의*, 박 만 수*, 김 성 탁*, 김 희 린*
(Chuleui Park*, Mansoo Park*, Sungtak Kim*, Hoirin Kim*)

*한국정보통신대학교, 공학부

(접수일자: 2005년 1월 7일; 수정일자: 2005년 3월 2일; 채택일자: 2005년 3월 25일)

본 논문에서는 내용기반 음악 정보 검색 방법으로써 멜로디의 시간 변화 특성과 통계적 특성을 모두 이용할 수 있는 hybrid 방법에 대해 제안하였다. 실제 방송 환경에의 적용을 위해 드라마 OST의 좁은 검색 범위뿐만 아니라 가요 1,005곡의 넓은 검색 범위에서도 제안한 방법을 이용하여 실험하였다. 제안된 방법은 특징 벡터로써 pitch와 MFCC (Mel Frequency Cepstral Coefficient)를 사용하여 음의 특성을 나타내었으며 멜로디를 표현하기 위해 피치 히스토그램과 VQ (Vector Quantization) 코드화한 MFCC의 템포럴 시퀀스를 이용함으로써 음악 검색 방법에 멜로디의 시간 변화 특성과 통계적 특성을 함께 적용할 수 있었다. 또한 pitch 히스토그램과 MFCC-VQ 템포럴 방법을 모두 사용한 hybrid 방식에 적절한 패턴 매칭 방법을 제안함으로써 기존의 각 단일 방식을 이용한 성능 결과 (MFCC-VQ 템포럴)와 비교하여 볼 때 드라마 OST 검색 범위에서는 평균 9.9%, 가요 1,005곡의 검색 범위에서는 10.2%의 오류 감소율을 나타내었다.

핵심용어: 음악 정보 검색, 히스토그램, 피치, QBE, VQ, MFCC

투고분야: 음악음향 및 음향심리 분야 (8.6)

This paper presents a new music identification method using probabilistic and dynamic characteristics of melody. The proposed method uses pitch and MFCC parameters as feature vectors for the characteristics of music notes and represents melody pattern by pitch histogram and temporal sequence of codeword indices. We also propose a new pattern matching method for the hybrid method. We have tested the proposed algorithm in small (drama OST) and broad (1,005 popular songs) search spaces. The experimental results on search areas of OST and 1,005 popular songs showed better performance of the proposed method over conventional methods. We achieved the performance improvement of average 9.9% and 10.2% in error reduction rate on each search area.

Keywords: MIR (Music Information Retrieval), Histogram, QBE, Pitch, VQ, MFCC

ASK subject classification: Musical Acoustics and Psychoacoustics (8.6)

I. 서 론

최근 기술과 산업의 발달로 문화에 대한 관심이 증가하고 그에 따른 많은 문화 컨텐츠가 공급되고 있다. 우리는 인터넷 기술의 발달로 이러한 많은 멀티미디어 컨텐츠들을 쉽게 접하고 있다. 특히 음악 데이터들은 MP3의 대중화로 인하여 엄청난 수요와 공급이 이루어지고 있다. 그러나 아직까지 이러한 엄청난 양의 데이터를 관리하는 일 (음악의 제목이나 가수별로 인덱싱 하는 일, 음악의 장르를 구분하는 일, 악기 별 구분하는 일 등)들은 많은 과제를 남기고 있다. 음악 컨텐츠 공급자들은 수많은 데이터들을 관리하고 운영하는 일 대부분을 아직까지 텍스트를 기반으로 한 수작업으로 이루어지고 있다. 따라서 이러한 단점을 보완하기 위하여 내용 기반 특징을 이용하여 음악을 검색하고 관리하는 기술에 대해

책임저자: 김 희 린 (hrkim@icu.ac.kr)
305-732 대전광역시 유성구 문지로 119번지
한국정보통신대학교 공학부 음성인식기술연구실
(전화: 042-866-6139; 팩스: 042-866-6245)

다양한 연구가 진행되고 있다. 디지털 신호 처리 기술의 발달과 컴퓨터 계산 능력이 증가됨에 따라 음악 구조의 분석, 악기 별 음향학적 특성의 모델링, 음악 패턴의 비교와 인식 등, 음악의 특성 파악이 쉬워졌다. 이러한 환경적 이점으로 우리는 내용 기반 음악 정보 검색 방법을 위한 연구에 보다 쉽게 접근할 수 있다. 내용 기반 음악 정보 검색 방법을 이용함으로써 보다 빠르게, 효율적으로 멀티미디어 데이터를 처리하고 관리할 수 있을 뿐만 아니라 여러 응용 분야에 적용 할 수 있다.

Liu et al[1]은 오디오 정보를 이용하여 광고 장면, 농구 장면, 축구 장면, 야구 장면, 뉴스와 날씨 장면과 같은 각각의 장면을 구분하였다. 각 장면을 구분하기 위해 특징 벡터로는 volume distribution, pitch contour, bandwidth, frequency centroid and energy로 구성하였다. ANN (Artificial Neural Network) classifier를 사용하여 각 클래스를 구분하여 88%의 결과를 얻었다. 그러나 ANN을 사용하는 것은 클래스 간의 관계가 복잡한 비선형적인 성질을 가질 경우 효과적으로 구분할 수 있지만 계산량이 많고 클래스 구분 관계를 정확히 알 수 없다.

또한 다른 기술을 이용해 Lu et al[2]는 silence, music, background sound, pure speech and non-pure speech의 5개의 클래스를 구분하기 위해 support vector machine (SVM) 방법을 이용하였다. 특징 벡터로는 MFCCs, zero-crossing rate (ZCR), short time energy (STE), sub-band powers, brightness, and band periodicity (BP), noise-frame-ratio (NFR)을 이용하였다. 커널 기반의 SVM은 클래스 간의 비선형 관계에 대해 커널 함수를 이용해 구분하고자 하는 데이터를 고차원의 선형 관계로 매핑하여 클래스 구분을 용이하게 해주는 장점이 있다. 이 방법은 평균 90%의 성능을 나타내었다.

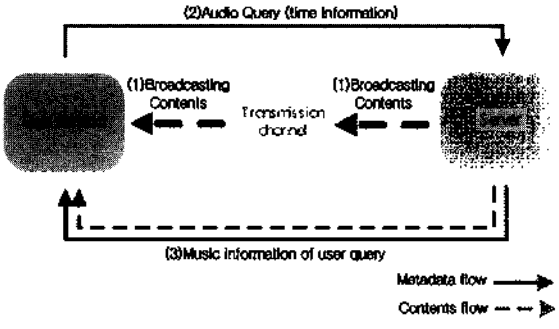

그림 1. 오디오 쿼리를 이용한 음악 정보 제공 검색 방법의 예
Fig. 1. The example of MIR based on QBE.

한편, 최근 Esmaili et al[3]은 이전의 연구를 바탕으로 rock, country, classic, folk, jazz 그리고 pop의 6개의 음악 장르를 5초의 클립을 이용해 구분하였다. 특징 벡터로는 entropy, centroid, centroid ratio, energy ratio, 그리고 location of minimum, maximum energy등 10개의 특징 벡터를 사용하였고, 패턴 매칭 방법은 LDA (Linear Discriminant Analysis)를 이용해 93%에 가까운 성능을 보여주었다. 그러나 구분하고자 하는 특징이 비선형 관계일 경우 LDA 방법은 효과적이지 않다.

그림 1은 본 논문에서 고려한 오디오 쿼리를 이용한 음악 정보 제공 서비스의 예를 나타내고 있다. 사용자가 서버 측에서 제공하는 컨텐츠를 시청하다가 배경 음악이 존재하는 부분에서 그 음악 정보에 대해 검색을 요청할 경우 서버 측에 사용자가 검색을 요청한 시점의 시간 정보를 메타 데이터 형태로 전송하게 된다. 서버 측에서는 컨텐츠에서 배경 음악에 해당하는 오디오 신호 일부를 추출하여 오디오 검색을 수행하고 그 결과에 해당하는 음악 정보를 시청자에게 제공한다. 메타 데이터에 의해 양방향 전송이 가능하기 때문에 사용자 단말에서 검색요청을 위한 오디오 쿼리 추출을 위한 시간 정보와 검색 결과에 해당하는 배경음악 정보는 메타 데이터 형태로 전송된다.

II. 음악정보 검색

현재의 QBE (Query By Example)의 내용 기반 음악 검색 방법에서 좋은 결과를 얻기 위해서는 아래의 세 부분을 고려해야 한다.

먼저, 특징 벡터를 결정하는 일이 중요하다. 음악은 음의 높이, 길이, 빠르기와 같은 특성을 가지고 있다. 이러한 멜로디를 구성하고 있는 음의 특성을 잘 표현하기 위해서는 특징 벡터를 결정하는 것이 중요한 부분이다. 보통 단음으로 이루어진 곡의 경우는 이러한 특성을 쉽게 파악할 수 있지만 실제 대부분의 경우 여러 악기와 음성으로 구성된 다중 음으로 표현되기 때문에 음의 특성을 표현하기가 어렵다. 따라서 이러한 다중 음의 특성을 나타내기 위해서 여러 기술들이 이용되고 있다. 예를 들면, 음의 비트 정보, 피치 정보나 MFCC, FFT, low-level audio feature, MPEG-7 descriptor[4,5]등

을 특징 벡터로 이용하여 음악의 멜로디 패턴을 나타낸다. 특히 MPEG-7 audio descriptor[4,5]는 음악 데이터에 대한 다양한 특징을 나타낼 수 있는 그룹별 특징 파라메터들을 제공한다. 그룹은 그 특성에 따라 timbral temporal, basic spectral, basic, timbral spectral, spectral basis, signal로 분류되어 총 17개의 특징 파라메터들을 제공한다. 그러나 잡음이 포함된 실제 테스트 음악의 멜로디 특성을 표현하기 위해서는 추가적으로 잡음 처리를 위한 과정을 거치거나 잡음에 영향을 받지 않으면서 음악의 고유한 특성을 잘 나타낼 수 있는 특징 벡터를 선정해야 할 필요가 있다.

두번째, 음악의 멜로디 패턴을 표현하는 방법의 선택은 중요한 부분 중 하나이다. 이전의 연구에서는 위의 특징 벡터들을 이용한 히스토그램을 이용하여 멜로디의 정적 패턴을 수치적으로 표현하거나 분석 프레임 별로 특징 벡터의 순차적 패턴으로 표현하였다. 또한 음악 악보를 MIDI 코드로 변환하여 텍스트를 기반으로 멜로디를 표현하기도 한다. 이러한 음악의 멜로디 패턴을 표현하는 방법은 음악의 중요 특성 중 하나인 음의 빠르기와 길이 같은 시간 정보의 사용 유무를 결정하기 때문에 검색 성능에 영향을 줄 수 있다. 그러므로 우수한 성능의 검색 방법을 설계하기 위해서는 적절한 멜로디 표현 방법의 선택이 필요하다.

마지막으로 비교할 두 멜로디 패턴 매칭 방법의 선택이 중요하다. 이는 음악 검색의 성능에 직접적인 영향을 끼친다. 그러므로 선택한 특징 벡터의 종류나 특성, 그리고 멜로디 표현 방법에 따라 적절한 패턴 매칭 방법의 선택이 필요하다. 그리고 음악 검색의 계산량에 대부분이 패턴 매칭 부분에서 발생하기 때문에 이 부분에서 검색 속도가 결정되어 진다. 패턴 매칭 방법으로는 비교할 멜로디의 패턴끼리 ED (Euclidean Distance), KNN (K-Nearest Neighbor), DP (Dynamic Programming) 등 과 같은 거리측정 방법들을 이용해 거리를 측정하여 검색하거나 음의 특성을 표현하는 특징벡터의 조합[5]을 이용하여 ANN[6], SVM (Support Vector Machine)[2], LDA (Linear Discriminant Analysis)[3]와 같은 classifier의 특성을 이용해 검색 결과를 찾아낸다.

본 논문에서는 오디오 쿼리를 기반으로 하고 있으며 음악의 다중 음의 특성을 표현하기 위한 특징 벡터로써 MFCC와 피치를 함께 사용하였다. MFCC와 피치는 각각 가지고 있는 특성이 다르기 때문에 음을 표현할 때 두 특징 파라메터가 나타낼 수 있는 특징을 모두 이용할 수 있는 장점이 있다. 또한 멜로디를 표현하는 방법은 MFCC의 VQ 코드화한 codeword sequence와 피치 히스토그램으로 멜로디를 표현함으로써 멜로디의 시변 특성과 통계적 특성을 제안한 음악 검색 방법에 반영하여 더 높은 성능의 검색 방법을 추구하였다.

본 논문의 구성은 다음과 같다. 제 3절에서는 논문의 제안한 검색 방법과 비교하게 될 피치 히스토그램 그리고 MFCC 템포럴 검색 방법에 대해 설명하고 본 논문에서 제안한 피치 히스토그램과 MFCC-동적 패턴을 결합한 검색 방법에 대해 설명하겠다. 그리고 제 4절에서 위의 각 검색 방법을 이용한 음악 검색 실험 결과를 비교하고 분석할 것이다.

III. 음악 정보 검색 방법

3.1. 피치 히스토그램

그림 2는 피치 히스토그램 방법[6]이다. 오디오 신호의 기본 주파수를 최소 62.5Hz에서 최대 1.5kHz 범위 내에서 bilinear scale로 등분하여 총 72차 frequency bin으로 구성된 피치 히스토그램 방법을 사용하였다. 히스토그램 방법은 멜로디에 확률적으로 유사한 음표에 해당하는 피치가 반복해서 나오는 특성이 존재하기 때문에 이러한 피치의 정적 패턴을 이용한 방법이다. 피치를 검출하는 방법은 여러 가지가 있지만 여기서는 MPEG-7에서 정의된 Audio FundamentalFrequency 서술자를 이용하여 피치를 검출하였다. 그리고 프레임 별 주기성

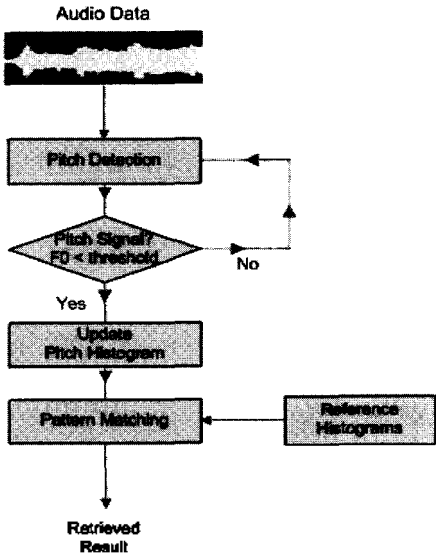

그림 2. 피치 히스토그램 검색 방법
Fig. 2. Pitch histogram retrieval method.

을 판단하여 하모닉 특성이 존재할 경우만 피치 히스토그램에 반영하였다. 즉, 기본 주파수가 1.5kHz 이상인 경우는 비 주기성 프레임으로 판단하여 히스토그램에 반영하지 않는다. pitch frequency bin에 대해서 각 클립의 히스토그램을 구한 다음 레퍼런스와 테스트 클립에 대해 패턴 매칭을 하여 결과를 얻게 된다.

피치 히스토그램 검색 방법은 멜로디의 시간 변화 특성을 검색에 사용하지 않고 멜로디 전체 구간의 확률적 특징만을 이용하여 검색하는 방법이기 때문에 음악의 중요한 시간적 정보인 템포, 박자를 이용하지 않는다. 그러므로 다음 절에 설명하게 될 검색 방법은 이러한 멜로디의 확률적 특성만 이용하는 특성을 보완하여 음악의 시간 정보를 이용함으로써 좀 더 나은 성능의 음악 검색을 수행하였다.

3.2. MFCC-VQ 동적패턴을 이용한 검색

그림 3에서 보듯이 템포럴 특성을 이용한 MFCC-VQ 검색 방법[7]은 크게 3가지의 과정으로 이루어져 있다. 즉, 특징벡터 추출 부분과 VQ 코드화를 통한 인덱싱 부분, 그리고 패턴 매칭을 통한 검색과정으로 이루어져 있다.

특징벡터 추출은 오디오 신호에 대해 40ms의 프레임 크기로 오버래핑 없이 데이터의 샘플링 주파수와 그에 따른 주파수의 해상도를 고려하여 64차의 MFCC를 추출하였다. 훈련 데이터에 대해 위의 특징벡터 추출과정을 거친 후 VQ를 통해 전체 곡 수와 음의 변화 정도를 고려하여 N개의 codeword로 이루어진 하나의 codebook을 얻는다.

생성된 codebook을 이용하여 훈련 데이터의 각 클립에 대해 프레임마다 해당되는 codeword index로 코드화한다. 테스트 클립에 대해서도 위의 과정을 거치면 비교할 두 클립의 indexed codeword sequence를 생성할 수 있다.

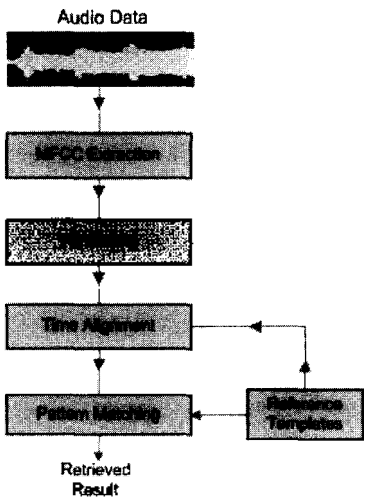

그림 3. 동적 특성을 이용한 MFCC-VQ 검색 방법
Fig. 3. MFCC-VQ retrieval method using dynamic patterns.

패턴 매칭은 위에서 추출한 길이가 같은 두 클립의 codeword sequence 간의 거리를 측정한다. 두 클립은 OST내의 각 곡들에 대해 1초의 간격으로 추출된 8초 길이의 클립들의 집합으로 이루어진 CD 음질의 레퍼런스 템플릿과 원하지 않는 신호가 섞인 테스트 클립 간의 패턴을 비교해야 하기 때문에 잡음에 의해 멜로디가 왜곡되는 문제가 발생한다. 또한, 두 클립의 동기 차이로 인해 거리가 왜곡되는 문제를 고려해야 한다.

3.2.1. Time alignment

비교하고자 하는 두 클립의 패턴 매칭에서 비슷한 멜로디를 포함하고 있다 할지라도 거리를 측정할 때 시작 부분에서 동기가 맞지 않으면 위의 그림 4에서 보듯이 동기가 맞지 않는 프레임 사이에 거리를 측정하게 된다. 그러므로 그에 따른 거리 값의 왜곡으로 인하여 잘못된 결과가 발생 할 수 있다. 따라서 위의 그림 4의 경우와 같이 테스트 쿼리의 처음 N개의 프레임과 레퍼런스 클립의 처음 N개의 프레임들 사이의 유사도를 측정하여 최대의 유사도 값을 갖는 프레임까지 레퍼런스 클립을 쉬프트 시킨다. 이렇게 함으로써 테스트 클립과 레퍼런스 클립간의 동기 차이를 어느 정도 보상 할 수 있다. (본 실험에서는 N을 15로 정의하였다.)

3.2.2. Smoothing을 이용한 거리 측정 (Modified ED)

$$\frac{1}{w+1}\sum_{m=1}^{M}\left|q_m - r_m\right| \tag{1}$$

$$S = \arg\min_{r\in R}\{WD(q,r)\} \tag{2}$$

WD : 가중된 거리값
N : 한 클립의 전체 프레임수
w : 같은 codeword 인덱스를 가지는 프레임수
q_n : 테스트 클립의 n번째 프레임의 codeword 벡터
r_n : 레퍼런스 클립의 n 번째 프레임의 codeword 벡터
R : 레퍼런스 클립 전체의 집합
S : 검색된 결과

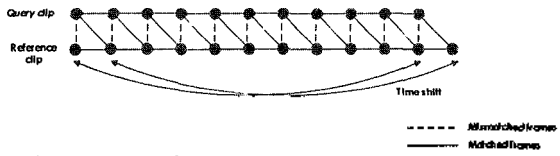

그림 4. 쉬프트를 통한 time alignment
Fig. 4. Time alignment by shifting.

위의 time alignment 과정에서 동기를 맞춘 두 클립에 대해 프레임 별 거리를 측정하는데 기존의 ED (Euclidean Distance) 방식을 적용하는 대신에 식 (1)에서처럼 기존의 ED로 프레임 별 측정한 거리에 같은 codeword index를 가지는 프레임 수로 나누어주는 modified ED식을 적용한다. 따라서 잡음을 포함하고 있는 테스트 패턴와 레퍼런스 패턴을 비교할 때 모든 프레임에 대해 정확히 일치하지 않더라도 전체 프레임 중 구간구간 일치하는 부분, 즉 노이즈가 없거나 약간 존재하는 부분에서는 비교하는 두 프레임의 codeword index가 일치한다. 이러한 일치하는 프레임이 발생하면 그 프레임 수만큼의 가중치로 나누어 줌에 따라 잡음에 따른 성능 저하를 어느 정도 보상할 수 있었다.

3.3. 피치 히스토그램과 MFCC-VQ 동적패턴의 결합

이전의 설명에서 알 수 있듯이 피치 히스토그램 검색 방법은 클립을 구성하고 있는 멜로디의 정적 패턴인 통계적 특성을 이용한 검색 방식이며, 반면 MFCC-VQ 템포럴 검색방법은 멜로디의 프레임 별 시간 변화 특성을 반영하여 검색을 하는 방식이다. 음의 높낮이를 나타내는 피치와 음성의 특성을 잘 표현할 수 있고 음성 신호처리에서 널리 쓰이는 MFCC는 서로 다른 특성을 가지고 있다. 즉, 주파수 축에서 음의 특성을 나타내는 피치와 캡스트럼 영역에서 음의 특성을 나타내는 MFCC는 멜로디를 구성하는 음의 서로 다른 특징을 각각 추출하여 나타낼 수 있다. 그러므로 2가지의 서로 다른 특성을 가진 검색 방법을 함께 사용함으로써 상호 보완적인 구조로써 좀 더 나은 성능의 음악 검색 방법을 제안하였다. 위의 그림 5는 본 논문에서 제안한 피치 히스토그램과 MFCC-동적 패턴을 결합한 검색 방법의 블록도 이다. 특징 백터의 추출은 앞 절에서 설명하였던 피치 히스토그램 시스템의 과정과 MFCC-VQ 템포럴 방법과 동일한 과정으로 특징 벡터를 추출한다. 마찬가지로 레퍼런스 템플릿의 생성과정도 피치 히스토그램과 MFCC-VQ 템포럴 방법과 같은 방식으로 레퍼런스 패턴을 형성한다.

패턴 매칭 과정에서는 서로 다른 특성과 구조를 가지고 있는 검색 방법을 함께 사용하기 때문에 적절한 패턴 매칭 방법을 사용하지 않으면 어느 한 검색 방법의 성능에 과도한 영향을 받아 그 검색 방법의 성능 이상의 결과를 얻기가 힘들다. 그러므로 본 논문에서는 이러한 두 검색 방법의 특성이 서로 잘 결합되어 좀 더 나은 성능을 나타내는 적절한 패턴 매칭 방법인 TSO (Two Step Ordering)방법을 제안한다.

먼저 MFCC-VQ 템포럴 방식을 이용하여 N-best의 결과를 추출하고 추출된 N-best의 결과에 해당하는 time alignment된 레퍼런스 패턴들을 가지고 다시 피치 히스토그램 방식을 이용하여 패턴 매칭을 수행한다. 그 후 패턴 매칭 결과를 측정된 거리에 대해 내림차순으로 정렬하여 그 순서에 따라 인덱스 붙인다. 그리고 마찬가지로 이전의 MFCC-VQ 템포럴 방식을 이용하여 나온 N-best의 결과에 대해서도 마찬가지로 측정된 거리에 대해 내림차순으로 정렬한 후 그 순서에 따라 인덱스를 붙인다. 위의 두 거리 인덱스의 합을 가지고 거리 측정을 하여 최소 거리 (즉 인덱스가 작을수록 레퍼런스 패턴과 유사도가 높다는 것을 나타냄)를 나타내는 곡을 검

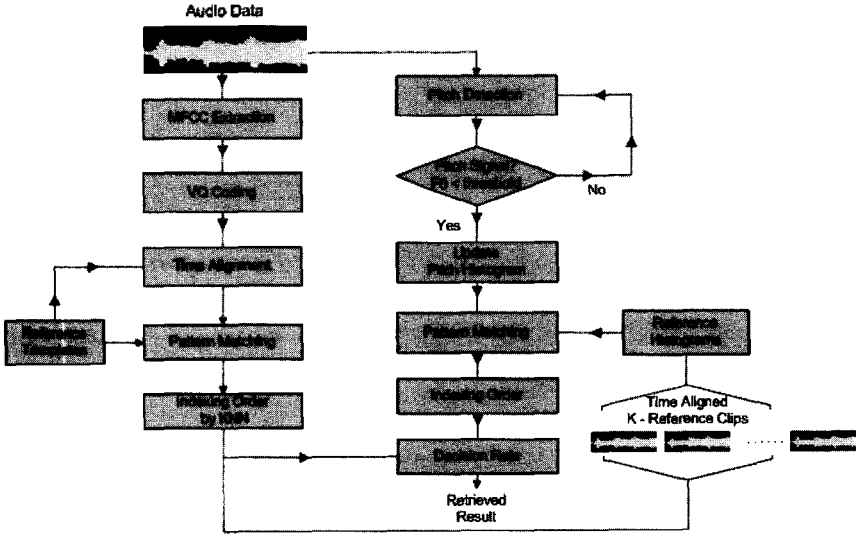

그림 5. 피치 히스토그램과 MFCC-VQ 동적 패턴의 결합 방법
Fig. 5. The hybrid method of pitch histogram and MFCC-VQ dynamic patterns.

색 결과로 나타낸다. 이는 피치 히스토그램과 MFCC-VQ 템포럴 방식이 제안한 검색 방법의 성능에 미치는 기여도를 특별한 정규화 방식을 사용하지 않아도 일정하게 할 수 있어 제안한 검색 방법의 성능이 어느 하나의 특정한 단일 방식의 성능을 따라가지 않고 효과적인 패턴 매칭을 수행할 수 있었다.

IV. 실험 결과

본 논문에서 사용한 실험 데이터는 제안된 검색 방법의 실제 환경에서의 성능 평가를 위해 드라마 '다모' 14부작 과 '옥탑방고양이' 16부작의 비디오 파일, 그리고 TV 라이브 음악 프로그램의 비디오 파일로부터 테스트 오디오 쿼리를 추출하였고 14곡으로 구성된 '다모' OST 앨범 (CD)과 20곡으로 구성된 '옥탑방고양이' OST 앨범 (CD), 또한 보다 넓은 검색 범위에서 실험을 하기위해 1005곡의 가요에 대해 각각 레퍼런스 템플릿을 구성하여 성능을 평가하였다 레퍼런스 템플릿은 각 곡에 대해 1초 간격으로 8초길이의 클립들로 구성되어 있다. 이 경우 드라마 OST에 대한 테스트 오디오 쿼리와 레퍼런스 데이터는 모두 44.1kHz, 16bit, 모노, 웨이브 포맷의 데이터로 구성하였다. 가요 1005곡에 대한 테스트 오디오 쿼리와 레퍼런스 데이터는 22.05kHz, 16bit, 모노, 웨이브 포맷의 데이터로 구성되어있다. 레퍼런스 데이터는 본 연구실에서 수집한 1005여곡의 MP3파일로부터 웨이브 포맷으로 변환하였다. 10005곡의 데이터가 드라마 OST의 데이터와 포맷이 다른 이유는 수집된 MP3 파일의 포맷이 일정하지 않아서 고주파 부분에서 차이가 존재하기 때문에 포멧을 일정하게 하기위해 22.05kHz로 다운 샘플링을 하였다. 비디오로부터 추출된 테스트 오디오 쿼리의 경우 배우들의 대사와 관중의 함성 등의 배경 잡음이 포함되어 있어 깨끗한 음질의 레퍼런스 데이터와 다르다. 테스트 클립은 8초의 길이의 '다모' 3,613개와 '옥탑방고양이' 5,659개, 그리고 TV 프로그램에서 추출한 가요 3곡의 473개의 오디오 클립들로 구성되었다. 실험은 크게 4부분으로 구성되어 있다. 먼저 MFCC-VQ 템포럴 검색 방식에서의 거리척도 방법에 따른 검색 성능평가 결과, 그리고 두 번째로 본 논문에서 제안한 검색방법에서의 패턴 매칭 방식에 따른 성능 평가, 세 번째로는 검색 범위가 작은 OST 앨범에 대한 각

표 1. 거리측정 방법에 따른 성능 평가 (MFCC-VQ 템포럴 방법)
Table 1. Retrieval accuracy to the distance measures in MFCC-VQ temporal method.

Contents / 거리척도	'다모'	'옥탑방고양이'	Average
ED	66.7%	65.2%	65.9%
Time alignment+ Modified ED	87.4%	90.4%	88.9%

표 2. 패턴매칭 방법에 따른 성능 평가 (피치 히스토그램과 MFCC-동적 패턴을 결합한 검 색 방법)
Table 2. Retrieval accuracy to pattern matching methods in pitch histogram + MFCC-VQ temporal.

Contents / 패턴매칭	'다모'	'옥탑방고양이'	Average
KNN	85.7%	86.9%	86.3%
Proposed method	88.9%	91.1%	90.0%

검색 방법에 대해 실험을 실시하였다. 마지막으로 검색 범위가 넓은 가요 1,005곡에 대해 각 검색 방법에 따른 실험 결과에 대해 보여줄 것이다.

표 1에서는 거리척도 방법에 따른 드라마 OST인 '다모'와 '옥탑방고양이' DB에 대해 성능 평가한 결과를 나타내고 있다. 성능 측정은 MFCC-VQ 템포럴 검색 방법에 대해서 이루어 졌다. 결과에서 보듯 패턴 매칭 하기 전에 전처리로써 time alignment를 수행하고, 거리 측정 시 가중치를 주는 modified ED를 사용하는 방법이 평균 88.9%의 검색률을 보여줌으로써 전처리 없이 기존의 ED 방식만 사용한 방법에서 나타낸 평균 66.5%보다 약 23% 정도의 우수한 성능을 나타내었다. 즉, time alignment를 수행함으로써 비교할 두 대상의 동기 문제를 보상하였고 또한 modified ED를 사용하여 테스트 오디오 쿼리에 포함된 잡음에 의해 발생하는 문제도 어느 정도 해결해 주고 있다.

표 2는 OST '다모'의 검색 범위에서 피치 히스토그램과 MFCC-동적 패턴을 결합한 검색 방법을 사용하여 패턴 매칭 방식에 따른 검색 결과를 보여주고 있다. 피치 히스토그램과 MFCC-VQ 템포럴 검색 방법에 대해 각각의 KNN 결과의 합을 이용하여 패턴 매칭한 경우 성능이 표3에 나타난 기존의 MFCC-VQ 템포럴 검색 방법의 성능보다 오히려 2% 정도 떨어진 결과를 나타내고 있다. 반면 제안된 패턴매칭 방법을 가지고 패턴 매칭을 수행하였을 경우 기존의 성능 (표 3) 보다 약 2% 정도의 성능 향상을 보였다. 이와 같은 결과에서 알 수 있듯이 제안된 패턴매칭 방법을 사용하였을 경우 어느 한 검색 방

표 3. 검색 방법의 종류에 따른 성능 변화 (드라마 OST)
Table 3. Retrieval correct rate to each retrieval methods in drama OST domain.

Contents / 검색 방법	'다모'	'옥탑방고양이'	Average
Pitch histogram	83.1%	81.3%	82.2%
MFCC-VQ temporal	87.4%	90.4%	88.9%
Hybrid	88.9%	91.1%	90.0%

표 4. 검색 방법의 종류에 따른 성능 변화(1005곡)
Table 4. Retrieval accuracy to each system in 1,005 popular songs.

Contents / 검색 방법	라이브 음악
Pitch histogram	23.4%
MFCC-VQ temporal	73.6%
Hybrid	76.3%

법의 특성에 과도한 영향을 받지 않고 두 검색 방법의 성능이 적절하게 결합되어 기존의 단일 검색 방법을 적용한 방식보다도 더 효과적인 성능을 나타내었다.

표 3에서는 검색 방법의 종류에 따라 검색 범위가 좁은 드라마 OST 실험 데이터에 대한 실험 결과이다. 기존의 검색 방법과 본 논문에서 제안한 검색 방법을 이용하여 검색을 수행하였을 경우 검색 결과를 나타내고 있다. 결과에서 보듯이 음악의 일정부분에서 반복되는 확률적 특성을 이용하는 피치 히스토그램 방식을 이용한 방식은 81~83%의 검색률을 보이며 가장 낮은 성능을 나타내고 있다. 이러한 방식은 음악에 있어서 중요한 시간적 변화 정보를 사용하지 않고 단지 멜로디의 정적 패턴만 이용하기 때문에 검색 결과의 정확성이 떨어지는 점을 알 수 있었다. 반면 음악의 시간적 변화 특성을 반영함으로써 멜로디 표현의 정확성을 높여 준 MFCC-VQ 템포럴 방법이 피치 히스토그램 방법보다 검색률이 7~8% 정도 우수한 성능을 나타내었다. 그리고 본 논문에서 제안한 피치 히스토그램과 MFCC-동적 패턴을 결합한 검색 방법이 기존의 두 검색 방법보다 더 나은 성능을 나타내고 있다. 이는 내용기반 음악 검색에 사용하는 특징 벡터로써 음의 높낮이 특성을 나타내는 피치와 음의 또 다른 특성을 나타내는 MFCC를 함께 이용하였고 또한 멜로디 표현 방법에 확률적 특성과 시간적 변화 특성을 모두 반영하고 적절한 거리척도 방식을 적용함으로써 우수한 검색 성능을 얻을 수 있었다.

표 4는 가요 1,005곡의 넓은 검색 범위에서 검색 방법에 따른 성능 결과를 보여주고 있다. 실험 데이터는 MP3에서 추출한 44.1kHz 웨이브 실험 데이터를 22.05kHz로 다운 샘플링 하였으며 테스트 데이터인 라이브 음악 데이터는 기존의 TV음악 프로그램에서 가수들이 자신의 노래를 라이브로 부르는 부분에서 추출하였다. 그러므로 실제의 레퍼런스의 노래와는 박자나 음정이 약간 틀리는 부분이 발생하였으며 관중들의 함성이 노이즈로 작용하였다. 전체적으로 검색 범위가 넓어짐에 따라 좁은 검색 범위인 드라마 OST에서의 실험 결과 보다 검색 방법에 따른 성능이 모두 떨어진다는 것을 알 수 있다. 이는 실험 데이터의 다운 샘플링에 따른 성능 저하와 검색 범위의 확장에서 오는 패턴 매칭의 부정확성, 테스트 데이터에 포함된 잡음에 따른 성능의 저하의 결과이다. 피치 히스토그램은 작은 범위에서 적용한 표 3에서 보다 성능이 매우 저하되는 것을 알 수 있었다. 이는 1,005곡의 넓은 범위에서는 피치 히스토그램 방법이 효과적이지 못하다는 것을 나타내고 있다. MFCC-VQ 템포럴 검색 방법은 검색 범위가 늘어남에도 불구하고 성능저하가 크지 않았으며 피치 히스토그램 방법보다 훨씬 더 나은 성능을 보였다. 그리고 제안된 hybrid 방법을 이용하여 실험하였을 경우 두 검색 방법보다 우수한 성능을 나타내고 있다. 이러한 결과를 볼 때 넓은 검색 범위에서도 제안된 검색 방법이 적용 가능하다는 것을 보여주었다고 할 수 있다.

그러나 보다 나은 검색 결과를 얻기 위해서는 검색 범위의 확장에 따른 성능 저하를 줄이기 위해서는 비교 할 쿼리의 길이를 10초 이상으로 길게 하거나 특징 벡터의 VQ 적용시 codebook의 사이즈를 검색 범위에 맞게 최적화 하는 작업이 필요하다고 생각한다. 또한 검색 방법의 검색 속도에 있어서도 OST의 검색 범위에서 보다 검색 범위가 늘어남에 따라 계산량이 많아져 그만큼 검색 속도가 느려지는 것을 알 수 있었다. 그러므로 실제 실시간 검색 방법을 구성하기 위해서는 방대한 레퍼런스 DB의 사이즈에 적합한 계산량이 적은 패턴 매칭 방법에 관한 연구가 필요하다.

V. 결 론

본 논문은 QBE 기반으로 드라마 OST내의 검색 범위와 가요 1,005곡의 검색 범위에서 음악 정보를 검색하기

위해 기존의 피치 히스토그램 검색 방법과 MFCC-VQ 템포럴 검색 방법을 모두 이용한 hybrid 검색 방법을 제안하여 기존의 단일 검색 방법보다 높은 성능을 나타내었다. 드라마 OST와 같은 좁은 검색 범위뿐만 아니라 넓은 검색 범위에서 적용해 봄으로써 제안한 검색 방법을 적용할 수 있는 가능성을 보였다. 또한 본 논문에서는 멜로디의 동적 특성뿐만 아니라 통계적 특성을 모두 반영한 hybrid 검색 방법에서 발생할 수 있는 어느 한 검색 방법의 성능에 과도한 영향을 받는 문제점을 본 논문에서 제안한 패턴 매칭 방식을 적용함으로써 문제점을 보완할 수 있었고 각 단일 검색 방법을 적용한 경우보다 음악 검색의 정확성을 높였다. 그러나 아직도 심한 잡음이나 동기 차이가 클 때는 성능저하가 발생하고 1,005곡에 대한 실험과 같은 넓은 검색 범위에서 검색 방법을 적용 시에 성능이 저하되는 문제점과 계산량의 증가는 해결해야 하는 과제로 남아있다.

검색 방법을 음악 검색을 방송 환경에 성공적으로 적용하기 위해서는 검색 범위에 따라 적절한 멜로디를 효율적으로 표현할 수 있는 방법, 그리고 그에 따른 적절한 패턴 매칭 방법에 대한 연구와 대사나 배경 잡음에 대한 강인한 검색 방법 설계, 마지막으로 비교할 두 음악 패턴 간의 동기의 불일치에서 오는 문제를 고려해야 한다.

향후 계획으로 위의 문제점을 해결할 수 있도록 밴드패스 필터링하여 노이즈에 강인한 검색 방법 설계 방법과 패턴 매칭시 좀더 세밀한 DTW (Dynamic Time Warping)[8] 방법과 같은 거리측정 방법을 적용하기 위한 구체적인 방안에 대해 연구할 계획이다.

참 고 문 헌

1. Z.Liu, J.Huang, Y. Wang, and T. Chuan, "Audio feature extraction and analysis for scene classification," in Proc. IEEE 1st Multimedia Workshop, 1997.
2. L. Lu, H. Zhang, and S. Li, "Content-based audio classification and segmentation by using support vector machines," Multimedia Systems Journal, 8 (6), 482-492, March, 2003.
3. S. Esmaili, S. Krishnan and K. Raahemifar, "Content based audio classification and retrieval using joint time-frequency analysis," in Proc. ICASSP, May 2004.
4. Overview of the MPEG-7 Standard(version 6.0), ISO/IEC, TC1/SC29/WG11/N4509.
5. BOZENA KOSTEK, "Musical Instrument Classification and Duet Analysis Employing Music Information Retrieval Techniques," *Proceedings of the IEEE*, 92, 712-729, April, 2004.
6. 박만수, 박철의, 김희린, 강경옥, "Pitch 히스토그램을 이용한 내용기반 음악 정보 검색," 방송공학회논문지, 9 (1), 2-8, 3월, 2004.
7. 박철의, 박만수, 김성탁, 김희린, 강경옥, "Temporal 특성을 이용한 내용기반 음악 정보 검색," 음향학회 가을학술대회, 2004.
8. Aggelos Pikrakis, Sergios Theodoridis, Dimitris Kamarotos, "Recognition of Isolated Musical Patterns Using Context Dependent Dynamic Time Warping," IEEE trans. Speech & Audio Proc.,11 (3), 175-183, May, 2003.

저자 약력

• 박 철 의 (Chuleui Park)

2003년 2월: 전남대학교 전자공학과 (학사)
2005년 2월: 한국정보통신대학교 공학부 (석사)
2005년 2월 ~ 현재: 대우일렉트로닉스 IS 연구소
※주관심분야: 내용기반 음악정보 검색, 음향정보 인덱싱, 음성인식

• 박 만 수 (Mansoo Park)

2000년 2월: 인하대학교 전자공학과 (학사)
2002년 2월: 한국정보통신대학교 공학부 (석사)
2002년 3월 ~ 현재: 한국정보통신대학교 공학부 (박사과정)
※주관심분야: 내용기반 음악정보 검색, 음향정보 인덱싱, 음성인식

• 김 성 탁 (Sungtak Kim)

2000년 2월: 울산대학교 전자공학과 (학사)
2003년 8월: 한국정보통신대학교 공학부 (석사)
2003년 9월 ~ 현재: 한국정보통신대학교 공학부 (박사과정)
※주관심분야: 음성인식, 음향정보 인덱싱

• 김 희 린 (Hoirin Kim)

1984년 2월: 한양대학교 전자공학과 (학사)
1987년 2월: 한국과학기술연구원 전자공학과 (석사)
1992년 2월: 한국과학기술연구원 전자공학과 (박사)
1987년 10월~1999년 12월: ETRI 선임연구원
1994년 6월~1995년 5월: 일본 ATR-ITL 방문연구원
2001년 1월~현재: 한국정보통신대학교 공학부 조교수
※주관심분야: 음성인식, 화자인식, 음향코딩, 음향정보 인덱싱