

내용기반 음악장르 검색에서 시계열 패턴 인덱스 화일의 성능 분석

(Performance Analysis of the Time-series Pattern Index
File for Content-based Music Genre Retrieval)

김 영 인*, 김 선 종*
(Young In Kim, Seon-Jong Kim)

요 약 음악 데이터의 양이 급속히 증가함에 따라 음악 데이터베이스의 오디오 특징을 이용한 내용기반 음악 장르의 효율적인 유사도 검색 방법이 요구되고 있다. 이러한 시스템을 구현하기 위해서는 시계열 패턴인 오디오 특징을 인덱싱 할 수 있는 인덱싱 기법과 데이터마이닝 기술이 필요하다. 본 논문에서는 인덱싱 기법을 기반으로 하는 유사 장르 음악 검색 시스템의 개발에 대하여 논의한다. 먼저, 시계열 패턴 인덱싱 기법과 데이터마이닝을 이용한 내용기반 음악장르 검색 시스템의 구조를 제안한다. 또한, 오디오 특징을 이용한 유사 장르 검색의 성능을 보이기 위하여 시계열 패턴 인덱스 화일을 구축하고 성능 분석을 제시한다. 실제 데이터의 특징값을 이용한 실험을 통하여 제안한 기법의 성능을 확인하였다.

핵심주제어 : 인덱싱 기법, 음악장르, 내용기반 정보검색

Abstract Rapid increase of the amount of music data demands for a new method that allows efficient similarity retrieval of music genre using audio features in music databases. To build this similarity retrieval, an indexing techniques that support audio features as a time-series pattern and data mining technologies are needed. In this paper, we address the development of a system that retrieves similar genre music based on the indexing techniques. We first propose the structure of content-based music genre retrieval system based on the time-series pattern index file and data mining technologies. In addition, we implement the time-series pattern index file using audio features and present performance analysis of the time-series pattern index file for similar genre retrieval. The experiments are performed on real data to verify the performance of the proposed method.

Key Words : indexing technique, music genre, content-based information retrieval.

1. 서 론

컴퓨터 기술의 발달로 인하여 디지털 형태로 저장되는 음악 데이터의 양이 기하급수적으로 증가하고 있다. 이러한 음악 데이터의 대부분은 오디오 데이터로서, 이를 효율적으로 저장하고 검색할 수

있는 기술에 대한 연구로 내용기반 음악 장르 분류 및 검색 시스템이 활발히 연구되고 있다[1,2,3]. 내용기반 음악장르 분류 및 검색 시스템이란 기존의 수작업 중심 분류 방식의 비효율성과 부정확한 부분을 개선하기 위하여, 음악의 내용을 기반으로 분석하여 대표적인 특징을 추출하고 데이터베이스의 인덱싱 기술과 패턴인식 기술을 이용하여 장르

* 부산대학교 바이오시스템공학부

를 분류하고 검색하는 시스템을 말한다. 이러한 연구는 최근까지 주로 패턴인식 기술과 데이터베이스 저장된 데이터를 활용한 데이터마이닝 기술을 사용하여 연구되어 왔으며, 데이터베이스의 인덱싱 기술을 이용한 연구도 진행되고 있다[1,2,3,4]. 이러한 내용기반의 분류와 검색 기술은 사용자가 원하는 곡을 신속하고 정확하게 제공할 수 있기 때문에 음악정보검색과 관련한 연구에 필수적인 기술이며, 음악정보관리자의 시간과 노력을 줄여주며 관리자의 주관적인 판단에 따른 오류를 줄일 수 있다.

내용기반 음악장르 분류 및 검색에 대한 기존 연구는 오디오 데이터와 관련 문자기술 그리고 오디오 특징 계수를 이용하고 있다[1,5,6]. 여기서 오디오 데이터를 이용한 방법은 적은 양일 경우에는 가능하나, 대량의 경우 유사도 검색과 사용자가 원하는 성능으로 정보를 제공하는데 한계가 있는 실정이며[5], 문자정보를 이용한 방법은 정보제공자의 주관적 견해와 모든 오디오 데이터에서 제공하기에는 어려움이 있다[1,2]. 오디오 특징 계수를 이용한 방법은 이러한 문제점을 개선하기 위한 것으로, 오디오 데이터가 내포한 특징을 추출하여 사용한다. 이 방법은 오디오 데이터에 비하여 소량이며 정형화된 형태를 갖고 있는 점과 보다 빠르고 효율적으로 분류 및 검색이 가능하여 현재 활발히 연구되고 있다[2,3].

그러나 최근에는 대량의 음악 데이터베이스에 저장된 오디오 데이터에서 특징을 추출하여 패턴 인식과 이를 데이터베이스에 이용한 데이터마이닝 기술을 사용한 음악장르 분석 방법을 연구하고 있으나, 학습 방법과 인식률에 있어서 실용화하기에는 부족한 실정이며, 이를 보완하여 줄 수 있는 데이터베이스의 인덱싱 기술에 대한 연구는 미미한 실정이다[7,8,9,10]. 본 논문에서는 기존의 내용기반 음악검색 시스템에서 사용하는 단편적인 방법과는 달리 인덱싱 기술과 데이터마이닝 기술을 함께 이용할 수 있는 시스템의 구조를 제안하고, 실제 오디오 특징 계수를 이용한 인덱싱 기술로 시계열 패턴 인덱스 화일을 구축하여 음악장르 검색을 실험하고 그 결과를 분석한다. 시계열 패턴 인덱스 화일은 특징 계수를 다차원 공간상의 점으로 다룰 수 있는 다차원 공간 인덱싱 기법이며 연속된 특

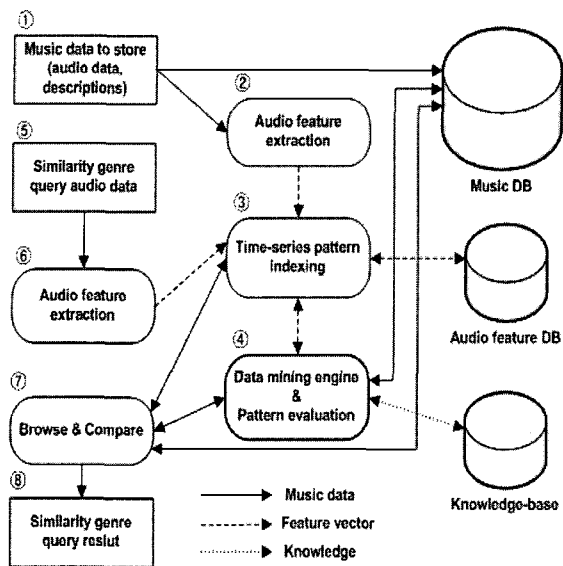
징계수의 순서를 감안한 저장과 탐색 기법을 가지고 있고, 기존의 EMDF[11]와 K-D-B-트리[12]보다 성능이 우수한 것으로 알려져 있다[11,13]. 그러나 현재까지 시계열 패턴 인덱스 화일을 이용한 음악 장르 검색과 같은 실제 응용에 대한 적용은 부족한 실정이다[8,9,10]. 따라서 본 논문에서는 오디오 특징계수를 효율적으로 저장하고 검색하기 위하여 시계열 패턴 인덱스 화일을 사용하며, 여기에 음악 장르 검색을 위하여 오디오 데이터에서 특징 계수를 추출하여 저장한다. 실제 응용에서 적용가능한가를 판단하기 위한 성능 분석을 위하여, 다양한 블록킹 인수와 유사도 그리고 공간분할방법을 사용하여 실험한다. 실험결과를 통하여 내용기반 음악장르 검색에 있어서 시계열 패턴 인덱스 화일이 적용가능한가를 평가한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 제안한 시스템의 전체 구조를 설명한다. 주요 구성 요소로는 시계열 패턴 인덱스 화일, 데이터마이닝 엔진과 그 관계를 제시한다. 제 3장에서는 본 논문에서 사용한 특징 계수와 음악 장르 분류에 사용된 시계열 패턴 인덱스 화일에 대하여 설명한다. 제 4장은 실제 데이터의 특징 계수를 사용한 실험 결과를 제시한다. 마지막으로 제 5장에서는 본 문을 요약하고 결론을 맺는다.

2. 시스템 구조

본 장에서는 본 논문에서 제안하는 내용기반 음악장르 검색 시스템에 대하여 설명한다. 본 시스템의 처리과정은 데이터베이스에 저장하는 과정과 미지의 입력 오디오 데이터와 유사한 장르의 데이터를 검색하는 2개의 과정으로 이루어지며, 그림-1은 전체적인 시스템 구조를 보여준다.

본 논문에서는 데이터마이닝과 같은 패턴인식 기술을 이용하는 기존 시스템과 달리 인덱싱 기술을 함께 사용하여 유사한 장르의 곡을 검색하는 기법을 제안하고 있으며, 이를 위하여 시스템은 2단계 과정으로 구성된다. 첫째로 음악 데이터의 저장 과정을 설명하면 다음과 같다. 먼저 음악 데이터(①)인 오디오 데이터와 관련된 문자정보를 음악 데이터베이스에 저장한다. 오디오 데이터에서는



<그림 1> 시스템 구조

적절한 클립 단위로 세그멘테이션과 해밍윈도우를 거쳐 오디오 신호처리 기법의 다양한 특징 계수를 추출(2)한 후, 시계열 패턴 인덱싱 기법을 이용하여 인덱스 구조(3)를 구성하고 오디오 특징 데이터베이스에 저장한다. 오디오 특징 데이터베이스에 저장된 특징 계수를 이용한 분석은 데이터마이닝 엔진(4)을 사용하여 수행하며 지식베이스를 구축한다. 본 논문에서는 다양한 오디오 특징계수 중 최근에 주로 사용되는 MFCC(Mel Frequency Cepstral Coefficient)를 추출하여 사용한다[2,5,14].

둘째로 미지의 곡에 대한 내용기반 장르 검색 과정은 먼저 해당 곡의 오디오 데이터를 시스템에 입력(5)하여 특징 계수를 추출(6)한다. 추출된 특징 계수는 구축되어 있는 시계열 패턴 인덱스 구조(3)를 통해 검색하며, 찾아진 후보곡과 유사 장르를 사용자에게 제공(7)하여 비교하여 찾아볼 수 있도록 한다. 이때 사용자가 데이터마이닝 엔진을 사용한 분류 결과도 원하면, 이에 대한 결과도 보여준다(4,7). 본 논문에서 언급한 데이터마이닝 엔진은 기존 연구에서 사용하는 시스템[2]을 이용하여 확장하는 것으로 가정한다. 본 시스템은 위와 같은 과정(3,4,7)을 반복할 수 있으며, 이를 통해 사용자에게 결과로서 찾아진 장르와 음악 데이터베이스에 저장되어 있는 유사곡을 보여주며 검색이 종료된다.

3. 특징계수 추출과 시계열 패턴 인덱스 화일

본 장에서는 실험 분석의 대상이 되는 오디오 특징계수인 MFCC[5,15]와 시계열 패턴 인덱스 화일[11]에 대하여 설명한다.

3.1 오디오 특징 계수 추출

오디오 데이터의 특징 추출은 오디오 신호의 크기(volume), 피치(pitch)와 에너지(energy) 등의 특성을 추출하여 구한다. 특히 A/D 변환을 통해 디지털화된 음악신호는 프리엠퍼시스 필터를 거쳐 원래의 성도 특성을 이끌어내고, 또한 청각시스템의 특징도 반영시켜 인간이 들을 수 있는 가장 좋은 환경을 가지도록 한다. 이러한 음악신호는 프레임 단위로 나누고, 나누어진 하나의 프레임을 갖는 음악 신호는 인간의 청각 시스템을 통하여 분류 또는 인식하기 때문에 이 청각시스템과 동일한 특성을 가지도록 모델링한 것이 MFCC이며, 음악장르 검색분야에서도 대표적인 특징값으로 사용되고 있다[2,5,14]. 따라서 본 논문에서는 오디오 특징값으로 MFCC를 사용한다.

그림-2는 MFCC의 추출과정[5,15]을 나타낸 것이다. 먼저, 주어진 오디오 신호는 일정한 시간범위 내의 신호인 프레임으로 나누고, 각 프레임단위로 MFCC를 추출한다. 그리고 다음 프레임은 시간을 중첩시켜 추출함으로써 정확한 특징을 추출할 수 있도록 하였다. 각 프레임은 DFT, Log, Mel, IDFT의 순서로 MFCC 계수를 만든다. 처음 DFT는 프레임 신호를 주파수 영역으로 변환하는 과정으로, N개의 샘플 데이터를 주파수 영역으로 변환하기 위한 X_n 은

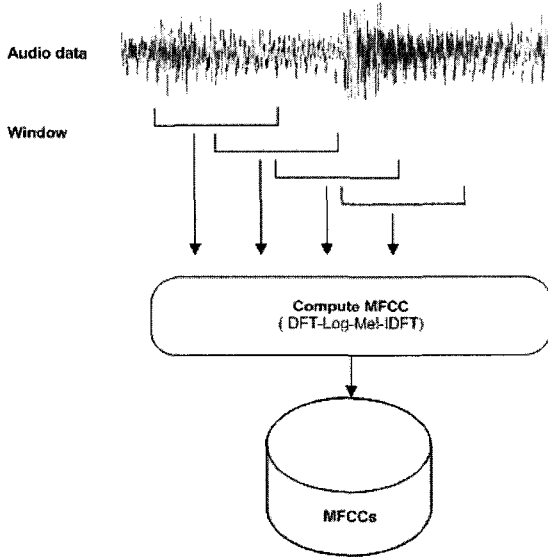
$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N}, \quad n = 0, 1, 2, \dots, N-1 \quad (1)$$

이다. 여기서 $j = \sqrt{-1}$ 를 나타내고, k는 N개의 샘플링 데이터를 나타내는 정수이다. 주파수 영역으로 변환된 신호 X_n 은 다시 여러 개의 필터뱅크에 의해 필터링 되고, 필터링된 에너지를 다시 로그를 취하는 Log 과정을 거친다. 그런 후, DCT(discrete cosine transform)를 한다. 이 때,

필터뱅크의 모양 및 중심주파수의 설정 방법은 사람의 달팽이관에서의 주파수 특성을 고려하여 결정되도록 한다. 일반적으로 삼각형 모양의 필터를 사용하며 중심주파수 1kHz까지는 선형적으로 위치하고, 그 이상에서는 mel 단위로 분포하는 20개의 बैं크로 만든다. 주파수에 따른 mel 함수는 다음과 같다.

$$mel(f) = 2595 \times \log_{10}(1 + f/700) \quad (2)$$

이렇게 필터링된 신호는 다시 IDFT 과정을 거쳐 MFCC 계수가 만들어진다.



<그림 2> MFCC 추출 과정

3.2 시계열 패턴 인덱스 화일

일반적으로 시계열 패턴 인덱스 화일[13]은 영역 벡터, 동적 영역 테이블, 계층 디렉토리 그리고 공간분할방법으로 구성된다. 각 구성 요소를 중심으로 설명하면 다음과 같다[8,10,13,16].

첫째로 영역 벡터는 Q-차원으로 구성된 데이터 공간내의 영역을 전체 분할 공간 영역에서 유일한 공간 영역을 나타내는 한 개의 해싱 값을 부여한다. 즉, 해싱 값은 영역 벡터로서, Q개로 구성된 영역 값에 대하여 영역을 2진수로 표현한 한 개의 원소를 갖는다. 그리고 영역 벡터는 계층 디렉토리

에서 전위 2진수 매칭을 하여 탐색한다.

둘째로 동적 영역 테이블은 영역 벡터와 영역 벡터의 공간 영역을 나타내는 각 차원의 최소값과 최대값을 갖고 있다. 유일한 영역 벡터를 지원하기 위하여 테이블의 엔트리가 Q-차원이라고 할 때 <영역 벡터, 1차원 영역 값, 2차원 영역 값,..., Q차원 영역 값>으로 구성되며, 동적 영역 테이블은 오직 한 개만 존재한다. 영역의 분열과 병합은 동적으로 변화한다. 동적 영역테이블의 예는 표-1과 같다.

<표 1> 동적 영역 테이블의 예

영역 벡터	X축 영역값 (≤ , <)	Y축 영역값 (≤ , <)
0	0 , 10	0 , 20
1	10 , 20	0 , 20
00	0 , 10	0 , 13
01	0 , 10	13 , 20
10	10 , 20	0 , 16
11	10 , 20	16 , 20
000	0 , 7	0 , 13
001	7 , 10	0 , 13
100	10 , 16	0 , 16
101	16 , 20	0 , 16
1000	10 , 16	0 , 11
1001	10 , 16	11 , 16

셋째로 계층 디렉토리 블록은 유일한 영역 벡터를 이용하여 구성한다. 하위 레벨의 디렉토리가 블록킹 인수를 초과하면 공통의 전위 2진수를 기준으로 상위 레벨의 디렉토리를 구성한다. Dir은 계층 디렉토리로서 한개의 x는 공간 벡터로서 디렉토리 엔트리의 데이터 영역을 나타내는 영역 벡터이며, 다른 한개의 애트리뷰트인 y는 이 데이터 영역에 대한 보다 구체적으로 저장된 하위 단계 페이지의 주소이다. 영역 벡터는 각 키 애트리뷰트에 대응되는 N개의 해싱 값으로 구성된다. i번째 해싱값은 디렉토리 엔트리의 데이터 영역내에 속하는 모든 데이터 레코드의 i번째 키 애트리뷰트를 해싱하였을 때 나타나는 해싱값들의 공통 전위(prefix)가 된다. D1은 최하위 단계 디렉토리로서, 한개의 x는 공간 벡터로서 디렉토리 엔트리의 데이터 영역을 나타내는 영역 벡터이며, 다른 한개의

에트리뷰트인 y는 이 데이터 영역에 속한 프레임 식별자 블록(FID : Frame Identifier) 저장된 하위 프레임 식별자 블록의 주소이다. 계층 디렉토리는 항상 균형 트리를 유지한다.

마지막으로 시계열 패턴 인덱스 화일의 공간 분할 방법으로는 기본인 최대거리기반분할방법과 윤변분할방법이 있다. 먼저 최대거리기반분할방법은 분할되는 공간의 분할 차원을 저장된 특징 계수의 각 차원 단위로 최대 값과 최소 값을 구하여, 그 중 차이 값이 가장 큰 차원을 선택한다. 윤변분할 방법은 한 저장 공간에서 오버플로우가 발생하면 분할되는 공간의 분할 차원을 미리 지정된 순서에 따라서 윤변(rotation)으로 선택하여 저장된 특징 계수의 해당 차원 값의 중간 값을 구하여 공간 분할 기준으로 삼는다.

4. 실험과 분석

본 장에서는 내용기반 음악의 장르검색에서 시계열 패턴 인덱스 화일의 적용과 성능을 평가한다. 시계열 패턴 인덱스 화일의 성능은 일반적으로 저장과 질의의 2가지 과정에서 성능을 구하여 분석한다[13,16]. 먼저 데이터 저장 과정은 계층 디렉토리 블록에서 저장할 프레임 식별자 블록을 찾는 과정과 찾아진 블록에 입력된 데이터를 저장하는 과정으로 이루어지며, 해당 저장 공간이 부족한 경우가 발생하면 공간분할 과정이 필요하다. 시계열 패턴 인덱스의 성능을 평가하기 위해서는 이러한 과정을 측정할 수 있는 소요 시간의 비교가 필요하다. 데이터가 저장되는 블록의 블록킹 인수는 시계열 패턴 인덱스의 성능에 영향을 주므로 다양한 블록킹 인수를 이용한 실험도 하여야 한다. 다음으로 질의 처리 과정의 검색 연산에 대한 성능은 후보 곡을 찾는데 소요되는 전체 시간과 검색 연산 결과의 정확도이다. 검색 연산에 소요되는 전체 소요 시간에는 후보 곡을 선정하는데 소요되는 시간도 포함한다. 그러므로 소요 시간을 측정함이 중요하다. 검색 결과의 정확도 측정을 위해서는 미리 저장한 곡에서 저장하지 않은 임의의 곡을 검색하는 과정을 수행하여야 한다. 유사도를 이용한 질의의 성능을 구하기 위하여 최소한 찾아진 데이터

블록의 모든 데이터를 분석하여 정확도를 계산하여야 한다. 따라서 본 장에서는 이와 같은 성능 분석 기준으로 실험하여 결과를 보이기 위하여, 제 1절에서 실험 환경을 설명하고, 제 2절에서는 실험 결과를 제시한다.

4.1. 실험 환경

시계열 패턴을 위한 인덱스에서 가장 영향을 많이 주는 파라미터는 프레임 식별자 블록의 블록킹 인수(blocking factor)로 알려져 있다[13,16]. 본 실험에서는 다양한 블록킹 인수를 사용하며 질의어 유형으로는 많은 사용이 예상되는 유사도를 감안한 질의 형태를 사용한다[17].

먼저 시계열 패턴 인덱스의 영역 벡터는 320비트를 사용한다. 사용하는 데이터는 곡은 MIREX [2]에서 구한 서로 다른 6가지 장르의 72곡을 사용하였다. 음악 장르의 분류는 이 분야를 연구하기 위하여 관련 연구자들이 정한 MIREX[2]의 분류 기준을 사용하였으며, 그 내용은 다음과 같다.

- ① 장르 1: 클래식 음악
- ② 장르 2: 전자 음악
- ③ 장르 3: 재즈와 블루스 음악
- ④ 장르 4: 메탈과 펑크 음악
- ⑤ 장르 5: 락과 팝 음악
- ⑥ 장르 6: 월드 음악

각 오디오 화일은 22050Hz, 16비트 모노로 변환한 후, 각 오디오 화일에서 시작부분의 60초를 지난 13.9초 분량의 오디오 클립을 음악정보검색 도구인 MARSYAS(Music Analysis, Retrieval and Synthesis for Audio Signals)[18]를 사용하여 구하였다. 구해진 오디오 클립에서 13차 MFCC를 특징계수로 추출하여 사용하였다.

본 논문에서는 장르를 기준으로 한 유사 탐색의 성능이 중요하다. 따라서 시계열 패턴 인덱스 화일에서 유사도 임계값 ϵ 의 변화에 따른 탐색 성능의 차이를 구하기 위하여 100, 200, 400 그리고 800을 사용하였다. 또 저장하지 않은 미지의 곡에 대한 장르 검색의 성능을 알아내기 위하여 각 장르의 곡중 2/3만 저장하고, 1/3은 탐색에만 사용하였다. 다양한 실험을 위하여 블록킹 인수는 200, 300, 400, 500 그리고 600으로 하였다. 블록킹 인수를

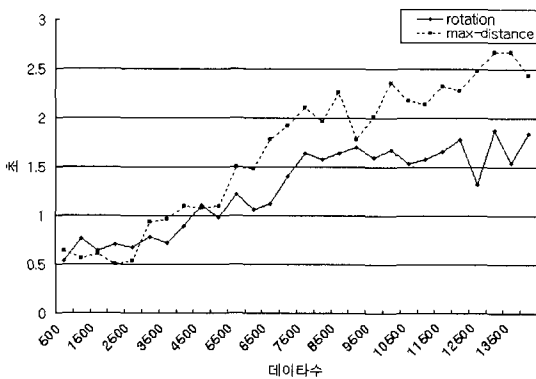
다양하게 한 이유는 블록킹 인수가 시계열 패턴 인덱스의 성능에 많은 영향을 미치기 때문이다. 실험에 사용한 컴퓨터는 IBM-PC 펜티엄4 (2.6GHz, 512MB)이며, 프로그래밍 언어는 C 언어를 사용하였다.

4.2. 실험 결과

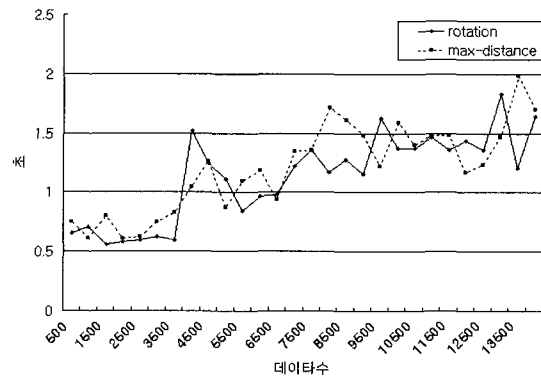
4.2.1 오디오 특징계수의 저장에 따른 소요 시간

6가지 장르의 오디오 특징 계수는 시계열 패턴 인덱스 화일에 블록킹 인수를 각각 200, 300, 400, 500 그리고 600으로 삽입하면서 발생하는 소요 시간을 관찰하였다. 이 실험의 목적은 시계열 패턴 인덱스의 분할 방법에 따른 소요 시간을 보이기 위함이다. 그림-3에서 그림-7은 5가지 블록킹 인수에 대한 실험 결과를 나타낸 것이다.

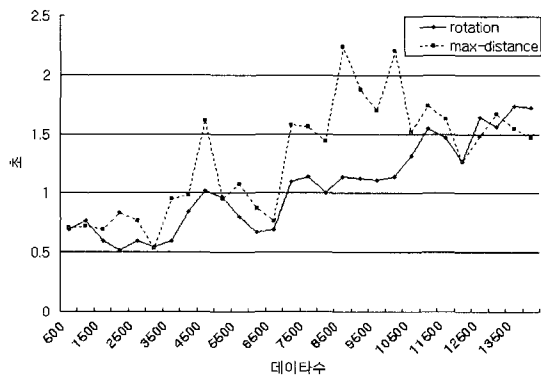
실험 결과, 최대거리기반 공간분할방법이 운반 분할방법보다 블록킹 인수가 200일 때 31.6%, 300일 때 6.2%, 400일 때 22.2%, 500일 때 25.6% 그리고 600일 때 34.1% 정도 증가하였다. 최대거리기반 분할방법의 소요 시간이 큰 이유는 분할 축을 구하는데 소요되는 시간이 많기 때문이다. 본 실험 결과에서도 시계열 패턴 인덱스 구축에 있어서 적절한 블록킹 인수의 선택이 성능에 많은 영향을 미친다는 것을 보여 주고 있으며, 블록킹 인수가 커지면 최대거리 기반 분할 방법으로 인한 소요 시간이 더욱 많아지고, 성능이 저하됨을 알 수 있다.



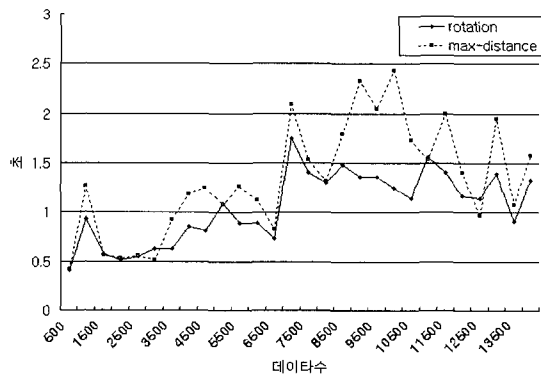
<그림 3> 데이터 삽입에 따른 소요 시간 (블록킹 인수 = 200)



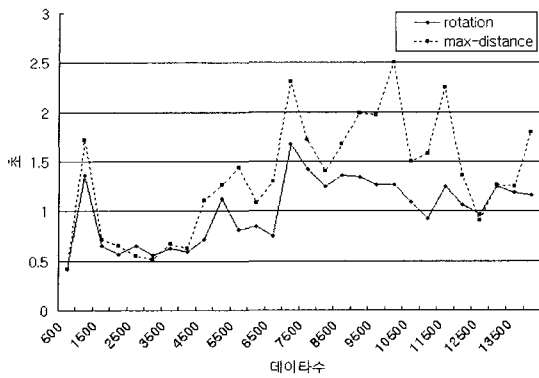
<그림 4> 데이터 삽입에 따른 소요 시간 (블록킹 인수 = 300)



<그림 5> 데이터 삽입에 따른 소요 시간 (블록킹 인수 = 400)



<그림 6> 데이터 삽입에 따른 소요 시간 (블록킹 인수 = 500)



<그림 7> 데이터 삽입에 따른 소요 시간
(블록킹 인수 = 600)

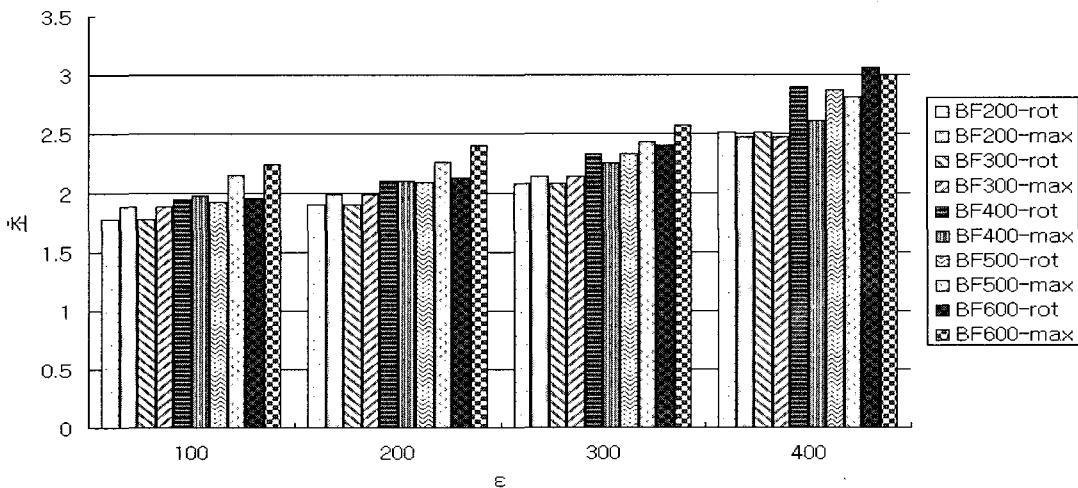
4.2.2 유사도 절의에 따른 소요시간 및 탐색결과

시계열 패턴 인덱스 화일은 탐색되어진 프레임 식별자를 저장하고 비교하여, 유사 장르의 곡을 선정한다. 탐색 성능의 측정에 있어서 소요 시간은 중요하다. 시계열 패턴의 탐색 성능을 측정하기 위하여, 4.2.1절에서 만든 시계열 패턴 인덱스 화일에 저장하지 않은 각 장르의 3곡을 4가지 ϵ 값을 사용하여 검색하여, 분할 방식에 따라 찾아진 곡과 장르를 비교하는 실험을 수행하였다. 이때 소요되는 시간도 구하였으며, 그 결과는 그림-8과 같이 ϵ 값이 클수록 탐색되는 공간이 넓어져 소요시간이 커짐을 알 수 있었다. 그러나 공간분할방법에 따라 탐색의 소요시간이 다르다고 간주할 수 있는 결과

는 발견되지 않았다.

다음으로 검색되어진 곡 중에서 같은 장르의 곡이 포함된 성공률을 구하기 위하여 5가지 블록킹 인수와 4가지 ϵ 값을 사용한 검색결과를 정리한 결과는 표-2와 같다. 표에 나타난 바와 같이 같은 장르의 곡이 포함된 곡 탐색률은 운번분할방법이 최대거리기반 분할방법보다 약 12.6%정도 높게 구해졌다. 장르에 따른 탐색률 차이가 많이 발생하였으며, 이는 같은 장르의 곡이라도 특징계수의 값 차이에 기인하였다. 그러나 본 실험에 사용한 곡의 클립이 13.9초 인점을 감안할 때, 약 40%이상의 탐색률을 보인 것은 우수한 탐색 성능으로 판단할 수 있다[2,3]. 블록킹 인수에 따른 성공률은 장르별로 차이가 있으나 400의 부근에서 보다 우수함을 알 수 있었다. 본 실험에서 사용한 특징 계수의 수가 약 15,000개이상임을 감안할 때, 저장하는 특징 계수의 갯수의 약 2.7%정도가 적절한 블록킹 인수로 판단된다. 다음으로 4가지 ϵ 값에 대한 성공률을 보면 많은 경우 400이상에서 우수한 결과가 나타났다. 따라서 본 실험에서 사용한 특징 계수의 저장 공간은 0에서 100,000범위이므로, 범위값의 약 0.4% 부근의 ϵ 값에서 우수한 탐색이 이루어질 수 있다고 판단된다.

탐색 과정에서 찾아진 곡이 많으면 탐색의 성공률이 높아지나, 탐색 소요시간 등의 성능이 저하되는 원인이 된다. 따라서 탐색 결과로 찾아진 곡



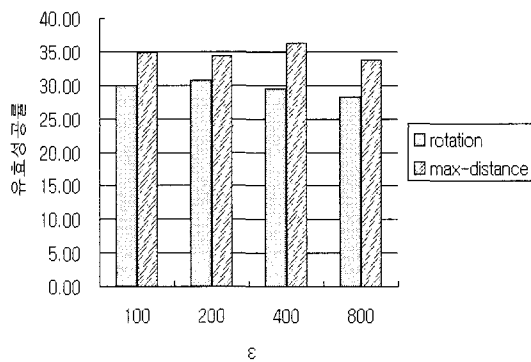
<그림 8> 유사곡 탐색에 따른 소요시간 (BF: Blocking Factor)

<표 3> 전체 장르에 대한 유사곡 탐색 성공률의 평균 비교표

ε 블록킹 인수	100		200		400		800	
	윤번분할	최대거리 기반분할	윤번분할	최대거리 기반분할	윤번분할	최대거리 기반분할	윤번분할	최대거리 기반분할
200	7.81	4.17	11.98	6.25	22.92	14.58	41.67	34.38
300	17.19	15.10	24.48	18.23	34.38	28.65	60.42	44.27
400	22.92	16.67	30.73	18.75	42.19	28.65	61.98	45.83
500	25.00	20.31	30.21	23.44	44.27	32.81	60.94	47.92
600	35.94	17.19	42.71	21.35	55.73	31.77	71.35	47.92

<표 4> 전체 장르에 대한 유사곡 탐색 유효성공률의 평균 비교표

ε 블록킹 인수	100		200		400		800	
	윤번분할	최대거리 기반분할	윤번분할	최대거리 기반분할	윤번분할	최대거리 기반분할	윤번분할	최대거리 기반분할
200	27.08	18.06	27.40	16.88	30.40	34.34	30.32	39.08
300	36.32	37.22	40.26	34.83	39.04	43.64	34.87	38.02
400	29.59	38.54	31.61	37.99	28.31	33.99	25.95	35.14
500	26.68	35.45	23.92	34.85	25.34	33.66	27.83	32.56
600	27.29	28.21	26.89	29.98	25.42	33.83	24.23	29.36



<그림 9> 평균 유사곡 탐색 유효성공률

중에서 해당 장르의 곡이 포함된 비율이 중요하며, 이를 유사곡 탐색 유효성공률로 본다. 이에 대한 결과는 표-3과 같으며, 그림-9는 유사곡 탐색 유효성공률을 4가지 ε값을 기준으로 평균하여 구한 결과이다. 그림-9와 같이, 검색된 곡 중 같은 장르인 경우가 최대거리기반 분할이 약 5.23% 정도 높음을 알 수 있었다. 따라서 저장한 곡의 수가 대량

인 경우에는 윤번분할방법을 사용하기 보다는 최대거리기반 분할방법을 사용하는 것이 유리함을 알 수 있다. 그러나 탐색의 성공률만 고려한다면 윤번분할방법을 사용하는 것이 효과적일 수 있음을 알 수 있다.

5. 결 론

본 논문에서는 시계열 패턴 인덱스 화일을 이용하여 기존의 내용기반 음악검색시스템에서 사용하는 방법과는 달리 데이터마이닝 기술과 인덱싱 기술을 함께 이용할 수 있는 시스템의 구조를 제안하고, 실제 오디오 특징 계수를 이용하여 시계열 패턴 인덱스 화일을 구축한 후, 음악장르 검색을 실험하고 그 결과를 제시하였다. 제안한 시스템은 데이터베이스의 인덱싱 기술과 데이터마이닝 기술을 함께 사용할 수 있는 특징을 갖고 있으며, 데이터베이스에 저장하는 과정과 미지의 입력 오디오

데이터와 유사한 장르의 데이터를 검색하는 과정을 처리할 수 있도록 구성하였다.

내용기반 음악장르 검색에 대한 시계열 패턴 인덱스 화일의 성능 실험은 저장과 탐색의 2가지로 구분하여 실험하였다. 먼저 저장 시의 실험 결과, 최대거리기반 공간분할방법이 각 블록킹 인수에서 윤번분할방법 보다 소요시간이 크게 나타났으며 이는 분할 축을 구하는데 소요되는 시간이 많았기 때문으로 분석되었다. 다음으로 탐색 시의 실험 결과에서는 첫째로 소요 시간을 구하였으며, 그 결과는 ϵ 값이 클수록 탐색되는 공간이 넓어져 소요시간이 커짐을 알 수 있었다. 둘째로 검색되어진 곡 중에서 같은 장르의 곡이 포함된 탐색물을 구하기 위하여 실험한 결과, 탐색물은 윤번분할방법이 최대거리기반분할방법보다 약 12.6%정도 높게 구해졌다. 셋째로 탐색물 향상을 위한 블록킹 인수는 저장하는 특징 계수 갯수의 약 2.7%정도가 적절한 것으로 구해졌으며, ϵ 값은 사용한 공간의 면적의 약 0.4% 부근을 사용하는 것이 우수하다는 결과를 얻었다. 마지막으로 유사곡 탐색 유효성공률을 비교한 결과, 최대거리기반 분할이 약 5.23% 정도 높았다. 그러므로 대량의 곡을 사용할 때는 윤번분할방법을 사용하기 보다는 최대거리기반 분할방법을 사용하는 것이 유리함을 알 수 있었다. 본 실험에서 구한 블록킹 인수와 ϵ 값을 감안하여 제안한 시스템을 구축한다면, 음악장르 검색의 성능 향상과 실용화에 매우 도움이 될 것으로 판단된다.

참 고 문 헌

- [1] R. Typke, F. Wiering and R. Veltkamp, "A survey of music information retrieval systems," *In ISMIR Proceedings*, pages 153-160, 2005.
- [2] The 1st Music Information Retrieval Evaluation eXchange, <http://www.music-ir.org/mirex2005/>, 2006. (accessed 8/2006)
- [3] J.J. Aucouturier and F. Pachet, "Representing Musical Genre: A State of the Art," *Journal of New Music Research*, 32(1), 2003.
- [4] G. Guo, H. Zhang, and S. Z. Li, "Boosting for content-based audio classification and retrieval: an evaluation," *Microsoft Research Tech Rep.*, MSR-TR-2001-15, 2001.
- [5] J. T. Foote, "Content-based retrieval of music and audio," *Multimedia Storage and Archiving Systems II, Proc. of SPIE*, pages 138-147, 1997.
- [6] D. Datta, "Managing Metadata," *In ISMIR Proceedings*, 2002.
- [7] J. Reiss, Jean-Julien Aucouturier and M. Sandler, "Efficient multidimensional searching routines for music information retrieval," *In ISMIR Proceedings*, 2001.
- [8] 김영인, "TIP-인덱싱 기법과 곡단위 특징을 이용한 내용기반 음악 검색", 한국산업정보학회논문지, Vol. 11, No. 3, pp.10-14. 2006.
- [9] 김영인, 김선중, "시계열 패턴의 효율적인 검색을 위한 공간 분할 정책의 성능 평가", 한국정보과학회논문지, Vol. 4, No. 3, pp71-76, 2006.
- [10] 김영인, "오디오 특징계수를 이용한 시계열 패턴 인덱스 화일의 뮤지션 검색 기법", 한국컴퓨터정보학회논문지, Vol. 11, No. 5, 2006.
- [11] 박영배, 이석호, "다차원 동적 인덱스 화일을 이용한 시계열 패턴 클래스의 예비선택 기법", 한국정보과학회논문지, Vol. 20, No. 11, 1993.
- [12] J. T. Robinson, "The K-D-B-tree : A search structure for large multidimensional dynamic indexes," *Proc. of Intl. Conf. on Management of Data*, ACM SIGMOD, 1981.
- [13] Kim Y. I., Park Y. & Chun J., "A Dynamic indexing structure for time-series patterns," *The Twentieth Annual International Computer Software and Applications Conference (COMPSAC'96)*, 1996.
- [14] J.J. Aucouturier and F. Pachet, "Music Similarity Measures: What's the Use?," *In ISMIR Proceedings*, 2002.
- [15] Minh N. Do, *Digital Signal Processing*

Mini-Project, http://lcawww.epfl.ch/~minhdo/asr_project/, 2006. (accessed 6/2006)

- [16] 김영인, "다양한 분포의 데이터를 이용한 시계열 패턴 인덱스의 성능 비교", 한국산업정보학회논문지, Vol. 3, No. 2, pp.134-142. 1998.
- [17] Jin Ha Lee, J. Stephen Downie and Sally Jo Cunningham, "Challenges in Cross-Cultural/Multilingual Music Information Seeking," *In ISMIR Proceedings*, 2005.
- [18] George Tzanetakis, Marsyas software 0.2, <http://opihi.cs.uvic.ca/marsyas/>, 2004. (accessed 4/2006)



김 영 인 (Young In Kim)

- 정회원
- 1996년 명지대학교 컴퓨터공학과(공학박사)
- 1996년~2006년 밀양대학교 컴퓨터공학부 부교수.
- 2006년~현재 부산대학교 바이오시스템공학부 부교수
- 관심분야 : 바이오지능정보시스템, 멀티미디어정보검색, 멀티미디어영상처리 등



김 선 중 (Seon-Jong Kim)

- 1996년 경북대학교 전자공학과(공학박사)
- 1995년~1997년 순천제일대학 제어계측과 전임강사
- 1997년~2006년 밀양대학교 컴퓨터공학부 부교수
- 2006년~현재 부산대학교 바이오시스템공학부 부교수
- 관심분야 : 신호 및 영상처리, 머신비전, 바이오시스템 등