

선율의 음높이와 리듬 정보를 이용한 음악의 유사도 계산 알고리즘

모 종 식[†] · 김 소 영^{††} · 구 경 이^{††} · 한 창 호^{†††} · 김 유 성^{†††}

요 약

컴퓨터 하드웨어 및 멀티미디어 정보 처리 기술의 발달로 인하여 멀티미디어 정보 검색 시스템에 대한 요구가 증대되고 있다. 멀티미디어 정보 검색 시스템은 텍스트와 이미지 위주로 발전하였으나 비디오와 오디오 정보, 특히 음악 정보를 위한 멀티미디어 정보 검색 시스템에 대한 요구가 점차 증대되고 있다. 최근의 음악 정보 검색 시스템에서는 해당 음악 정보의 작곡가, 곡명 등과 같은 메타 정보만을 이용한 메타 정보 기반의 검색뿐만 아니라 음악의 내용을 기반으로 하는 내용 기반 검색까지도 가능하다. 음악의 내용 기반 검색에서는 메타 정보 기반 검색에서 흔히 볼 수 있는 문자열에 대한 패턴 매칭보다는 멀티미디어 데이터간의 유사도를 기반으로 검색한다. 본 논문에서는 음악을 일정한 선율에 따른 음들의 배합으로 정의하고, 선율을 구성하는 음의 높낮이와 음의 길이 정보를 복합적으로 이용하여 두 음악 정보간의 유사도를 계산하는 알고리즘을 제안한다. 또한, 본 논문에서는 제안된 유사도 계산 알고리즘의 검증에 대해 동요를 대상으로 실험을 하였다. 실험 결과에 따르면 제안된 유사도 계산 알고리즘이 기존에 제안된 내용기반 음악 정보 검색 시스템과 비교하여 음악의 선율 정보를 기준으로 음악들간의 유사도를 보다 정확하게 효과적으로 계산하였다.

A Similarity Computation Algorithm Based on the Pitch and Rhythm of Music Melody

Jong-Sik Mo[†] · So-Young Kim^{††} · Kyong-I Ku^{††}
Chang-Ho Han^{†††} · Yoo-Sung Kim^{†††}

ABSTRACT

The advances of computer hardware and information processing technologies raise the needs of multimedia information retrieval systems. Up to date, multimedia information systems have been developed for text information and image information. Nowadays, the multimedia information systems for video and audio information, especially for musical information have been grown up more and more. In recent music information retrieval systems, not only the information retrieval based on meta-information such like composer and title but also the content-based information retrieval is supported. The content-based information retrieval in music information retrieval systems utilize the similarity value between the user query and the music information stored in music database. In this paper, hence, we developed a similarity computation algorithm in which the pitches and lengths of each corresponding pair of notes are used as the fundamental factors for similarity computation between musical information. We also make an experiment of the proposed algorithm to validate its appropriateness. From the experimental results, the proposed similarity computation algorithm is shown to be able to correctly check whether two music files are analogous to each other or not based on melodies.

※ 본 연구는 1998년도 정보통신부 초고속정보통신 응용기술개발사업의 지원으로 시작되었으며, 현재 본 연구는 한국과학재단 목격기초연구(2000-1-51200-009-2)지원으로 수행중
† 준 회 원 : ㈜ 이지시스템

†† 준 회 원 : 인하대학교 대학원 전자계산공학과
††† 종신회원 : 인하대학교 전자계산공학과 교수
논문접수 : 2000년 5월 2일, 심사완료 : 2000년 11월 25일

1. 서론

멀티미디어 정보란 기존의 정형화된 정보뿐만 아니라 소리, 사진, 동영상 등이 한데 어우러져 있는 정보의 형태이다. 우리가 보고, 듣고, 말하는 모든 행동을 가장 잘 모델링하기 위해서는 멀티미디어 정보 형태의 접근은 필연적이며, 최근에 많이 개발되고 이용되는 모든 정보 시스템의 멀티미디어화가 멀티미디어 정보 시스템의 필요성을 잘 설명해 주고 있다. 멀티미디어 정보 검색에는 메타 정보 기반 검색과 내용 기반 검색, 두 종류의 검색 방법이 존재한다. 최근 하드웨어의 발달 및 멀티미디어 정보 처리 기술의 발달에 따라 관점의 차이로 인한 검색 오류 가능성이 큰 메타 정보 기반 검색보다는 내용 기반 검색에 대해 활발히 연구되고 있다. 특히, 이미지 프로세싱 기술과 이미지 데이터 저장 방식의 급속한 발달에 힘입어 내용 기반 이미지 검색에 대한 연구가 두드러진다고 할 수 있다[1, 2].

이에 반해, 오디오, 특히 음악에 대한 내용 기반 검색에 대한 연구는 미비하다. 기존의 연구 사례들이 있지만 단순히 선율 정보를 음의 높낮이 변화를 의미하는 문자열 형태로 표현하여 패턴 매칭의 불일치 수로서 유사도를 계산하기 때문에 정확한 유사도를 계산하지는 못하는 것으로 판명되었다. 음악 정보의 유사도 계산에서 단순히 불일치 수를 이용하여 유사도를 계산하는 것은 불일치율에 대한 정보를 전체 유사도 계산에 반영시키지 않고, 음의 길이를 유사도 계산 단계에 포함시키지 않음으로써 매우 부정확한 값으로 계산되게 된다[3]. 이러한 유사도 계산 알고리즘은 원음 자체에 오차를 포함하고 있는 두 음악 파일간의 유사도를 계산하기 위한 목적으로는 사용할 수 있으나 원음에 오차를 포함하고 있지 않는 두 음악 파일간의 정확한 유사도를 계산하는 목적으로는 사용할 수 없다.

본 논문에서는 음악에 대한 유사도를 정확하게 계산하기 위해 음악의 선율을 구성하는 음의 높낮이 정보와 음의 길이 정보를 복합적으로 고려하여 음악 정보간의 유사도를 계산하는 알고리즘을 제안하였다.

음악에 대한 유사도 계산 알고리즘의 응용 분야로는 음악 정보 검색 분야 외에 여러 분야에 활용될 수 있는데, 한 활용 방안으로는 고질적인 사회적 문제로 대두되고 있는 음악의 표절 여부를 정확하게 판정하기 위해 음악의 내용 기반 유사도를 계산하는 목적으로 사용될 수도 있다. 또한, 특별한 교사 없이 자기 스

로 교육할 수 있게 하는 음악 교육 시스템에서의 노래 및 연주 결과의 정확한 평가 알고리즘으로 활용될 수 있을 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 음악의 선율에 대한 기본적인 개념과 기존에 연구된 음악 정보 검색 시스템에 대한 소개 및 개선점에 대해 기술한다. 3장에서는 본 논문에서 제안된 선율 정보를 이용한 유사도 계산 알고리즘에 대해 자세히 기술하고, 4장에서는 제안된 유사도 계산 알고리즘을 이용하여 동요 3곡을 실험 데이터로 이용하여 실험한 결과 값의 분석을 통해 제안된 알고리즘의 타당성을 검토한다. 마지막으로 5장에서 결론 및 향후 연구 방향을 기술한다.

2. 관련 연구

본 장에서는 음악의 선율에 대한 기본적인 개념과 기존에 연구된 음악 정보 검색 시스템을 분석하고 음악 정보 검색 시스템의 개선점 등을 설명한다.

2.1 음악의 선율 정보

19세기 벨기에의 음악학자 Francais Joseph Fetis (1784-1871)는 “음악이란 음의 배합에 의해 사람의 감정을 감동시키는 예술이다”라고 말했다[4]. 즉, 음악은 음의 배합이라 할 수 있으며, 어떤 음이 어디에 배치되느냐에 따라 음악은 다른 의미로 해석된다. 결국, 음악의 고유성은 음악을 구성하는 음들의 배열에 따라 달라진다. 즉, 음악에서 가장 중요한 것은 음악의 선율이라고 할 수 있다. 음악의 선율이란, 여러 개의 음을 일정한 규칙에 따라 미적, 시간적으로 리듬을 갖고 배열한 것으로 음악이 청취자에게 음악이 표현하고자 하는 바를 전달하는 중요한 음악 구성 요소이다. 따라서 음악의 선율이란 리듬을 갖고 시간적으로 나열된 연속적인 음표들의 집합이라고 정의할 수 있다.

음악의 선율을 구성하는 음의 주요 성질에는 다음과 같이 높이(pitch), 길이(duration), 세기(intensity), 음색(timbre, tone color)이 있다[5, 6].

- 음의 높이 : 음높이는 본질적으로 1초 동안의 진동수에 따라 달라지며 헤르츠(Hertz : Hz)로 단위를 나타낸다. 높은 진동수는 높은 음, 낮은 진동수는 낮은 음에 해당된다.

- 음의 길이 : 음의 길이는 진동 시간에 비례한다. 즉, 진동이 계속되는 시간에 따라 음의 길이가 길고 짧음이 결정되며, 진동이 그치면 음도 따라서 소멸된다.
- 음의 세기 : 음의 세기는 주로 진동의 폭에 따라 달라진다. 진폭이 클수록 그 음이 세어지고, 진폭이 작을수록 그 음도 여려진다. 즉, 진동하는 에너지의 크기, 진동의 방향과 거리 및 주위의 상황에 따라서 음의 세기가 결정된다.
- 음색 : 기초음과 더불어 진동하는 배음들의 수와 세기는 각 악기마다, 노래하거나 말하는 음성마다 다르며 각각 고유한 배음렬을 보여준다. 따라서 소리를 내는 발음체의 구조와 재료의 차이, 또는 발음 방법에 따라 음색이 달라진다. 진동에 따라 배음렬의 어떤 부분음(윗배음)이 어느 정도의 강도로 발생하느냐에 의해서 음색이 결정된다. 따라서 종합적으로 진동하는 음파의 모양에 따라 음이 갖는 감각적 성질이 다르다.

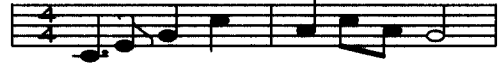
음의 4가지 특성, 즉 음의 높이, 길이, 세기 및 음색 중에서 음의 높이와 길이를 음의 구조적 특성, 음의 세기와 음색을 음의 장식적 특성으로 구분한다[4-6]. 본 논문에서 하나의 음악 작품을 결정하는 요소를 음악의 선율을 구성하는 여러 요소들 중에서 음의 구조적 특성인 음의 높이와 음의 길이로 규정하고, 이를 이용하여 두 음악들간의 유사도를 계산한다.

2.2 기존의 음악 정보 검색 시스템

음악 정보를 내용 기반으로 검색하는 시스템에 대한 연구로는 미국 코넬대학의 Query by Humming[7], 뉴질랜드 와이카튜대학의 Melody Index[8], 그리고 국내의 숭실대학교에서 발표한 음표 열 탐색 알고리즘을 사용하는 음악 정보 검색 시스템[3]이 있다. 본 절에서는 이들 연구 중에서 가장 최근에 발표되었고 다른 2개의 기존 연구 결과를 발전시킨 [3]의 내용을 기본적으로 설명하고자 한다.

음악 정보 검색 시스템에서는 검색을 위해서 음악의 구성 음들의 높낮이 변화를 표현한 문자열 데이터를 이용한다. 즉, 음정 곡선은 인식된 음표들을 차례로 한 다음, 두 음표들간의 차를 음계에 의해 구한다. 구해진 음정의 값이 0보다 크면 그 수만큼은 'A'에서부터 'M'까지 차례로 할당하고, 0보다 작으면 구해진 값만큼은

'O'에서부터 'Z'까지 할당하며, 음정의 차가 0, 즉 1도이면 'N'을 할당하여 문자열로 나타낸다. (그림 1)은 동요 "봄바람"의 첫 동기인데, 이를 음표(도미솔도 라도라솔)들에 대한 음정을 계산하여 음의 높낮이 변화를 표현한 문자열로 나타내면 "BBCPBPO"로 표현된다.



(그림 1) 동요 "봄바람"의 첫 동기

사용자가 검색을 원하는 노래 중 알고 있는 부분을 마이크를 통해 입력하면 음표 인식을 통해 음표 정보가 인식되고, 이를 이용하여 음의 높낮이 변화를 표현하는 문자열을 만들어 탐색 패턴을 구성한다. 구성된 탐색 패턴을 가지고 음악 데이터베이스내의 각 음악의 음의 높낮이 변화를 표현하는 음표 문자열 즉, 음정 곡선과 비교하여 [3]에서 제안된 음표 열 탐색 알고리즘을 이용하여 탐색을 시도한다. [3]은 음표 열 탐색 알고리즘의 빠른 탐색을 위해 데이터베이스에 저장된 음정 곡선에서 나타나는 각 문자 종류별로 발생 위치를 조사하여 저장하고 있다. 따라서, 탐색 패턴의 첫 문자가 음정 곡선상에서 나타나는 위치를 미리 알고 있으므로 빠른 탐색을 할 수 있으며, 또한 병렬 탐색이 가능하므로 효율적이라 할 수 있다. 즉, [3]은 사용자가 악보 없이 정확한 음을 표현하기 힘들다고 판단하여 단지 음들간의 높이차 정보를 가지고 검색을 실시한다. 즉, 음들간의 높이차 정보를 이용하여 음정 곡선을 구하고, 이를 이용하여 문자열 패턴 매칭 방법을 사용한다. 따라서, 비교시에 미리 정해진 불일치 허용수보다 적으면 후보곡으로 검색되어 최종적으로 사용자가 판단하게 하는 시스템이다. 이러한 기존의 음악 정보 검색 시스템은 몇 가지 문제점을 드러낸다.

먼저, 각 음표들간의 차 값을 이용하여 대략적인 흐름 정보를 담은 음정 곡선은 음표들간의 차 값은 동일하지 않아 부정확한 값을 지니게 된다. 예를 들면, 음표가 "도미솔"이면 음악 정보 검색 시스템의 음정 곡선은 "BB"로 표현된다. 그러나, '도'와 '레', '레'와 '미' 사이는 온음차이지만, '미'와 '파' 사이는 반음차가 된다[5]. 따라서, 실질적으로 "도미"와 "미솔" 음표들간의 차 값이 동일하지 않음에도 불구하고 동일한 값으로 표현된다. 비록 음정 곡선이 대략적인 정보만을 나타낸다고 하더라도 그 대략적인 정보는 올바른 정보에 대한 대략적

인 정보이어야 할 것이다. 따라서, 음표간의 차를 음계에 의해 구할 것이 아니라 반음 단위로 할당해야 한다. [7,8]시스템은 [3]의 시스템보다 더욱 간단하게 연속되는 두 음의 차 정보를 나타내는데, 비교되는 두 음 중 음이 다음 음보다 높을 경우 'D(Down)'로, 음이 다음 음보다 낮을 경우 'U(Up)'로, 서로 같을 경우 'R(Repetition)'로 표시하고 있다. 따라서, 음악 검색의 정확성이 [3]에 비해 낮아지게 된다. 두 번째로 기존의 음악 검색 시스템들은 음의 선율 정보 중 음의 길이는 전혀 고려하지 않는다[3,7,8]. 마지막으로, 기존의 음악 정보 시스템에서는 문자열에 대한 패턴 매칭 방법을 통해 탐색하고, 탐색의 정확률을 단지 음정 곡선의 불일치 문자수로 계산함으로써, 검색의 정확성이 낮아지게 된다.

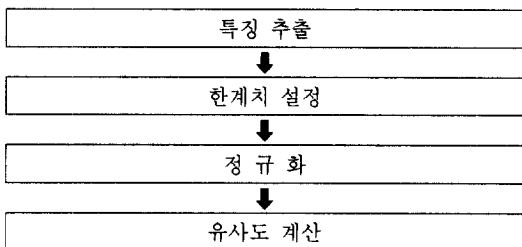
따라서, 본 논문에서는 검색시 단순한 패턴 매칭 방법을 사용하지 않고, 좀더 정확성을 높이기 위해 음악 선율을 구성하는 음의 높이와 길이 정보를 복합적으로 고려하여 유사도를 계산하는 방법을 제시하고자 한다.

3. 유사도 계산 알고리즘

본 장에서는 음악에 있어 선율을 구성하는 음의 높이와 길이를 이용하여 두 음악간의 유사도를 계산하는 알고리즘을 설명한다. 본 논문에서 제안하는 유사도 계산 알고리즘은 음악 파일로서 국제적 표준인 미디(MIDI : Musical Instrument Digital Interface) 파일[9]을 대상으로 고려한다.

3.1 유사도 계산 알고리즘의 전체적 흐름

본 논문에서 제안된 선율 정보를 이용한 유사도 계산 알고리즘의 전체적 흐름도는 (그림 2)와 같다.



(그림 2) 제안된 유사도 계산 알고리즘의 흐름도

력된 미디 파일로부터 음악의 특징 요소 및 유사도 계산에 필요로 하는 정보를 추출하고, 특징 요소의 추출된 값을 통해 상대적 음높이와 음의 길이를 반영하는 비교 선분의 기울기 차에 대한 한계치를 설정한다. 한계치 설정 후, 두 음악의 비교 단위 및 비교 단위내의 음들의 발생 시간의 정규화를 통해 유사도를 계산한다.

3.2 음악의 특징 요소 추출

동요 "달"의 미디 파일을 Microsoft Visual C++에서 컴파일한 그림이 (그림 3)과 같다. 입력된 미디 파일로부터 음악의 특징 요소 및 유사도 계산을 위해 필요로 하는 정보를 추출해야 하는데, 필요로 하는 정보는 박자(Time-meter), 키(Key), 음의 높이(pitch)와 음의 길이(length) 집합 등이다.

추출되어야할 정보들은 미디 파일의 트랙 청크(MTrk chunk)안에 저장되어 있다. (그림 3)에서 첫 줄의 첫 4바이트의 16진수 값은 '4D 54 68 64'가 되는데, 이는 청크 ID를 나타낸다. 즉, 헤더 청크(MThd chunk)를 나타내는 ID이다. 그 뒤에 '00 00 00 06'이 오는데, 이는 헤더 청크의 길이를 나타내는데, 총 6바이트 길이를 가짐을 나타낸다. 그 뒤의 6바이트는 2바이트씩 묶어서 의미를 가지는데, 미디 형식, 트랙 청크 수 및 디비전(Division)을 각각 나타낸다. (그림 3)에서 나타난 값들이 실제로 의미하는 정보들은 다음과 같다. 미디 형식은 형식 1을 나타내고, 트랙 청크 수는 3개이고, 디비전은 '0x00', 즉 '192'를 나타낸다. 유사도 계산을 위해 필요로 하는 정보는 첫 트랙 청크에는 포함되지 않고, 둘째 트랙 청크((그림 3)의 음영으로 표시된 부분)에 포함되어 있다.

```

000000 4D 54 68 64 00 00 00 06 00 01 00 00 00 00 4D 54 MThd.....M
000001 72 58 00 00 00 00 00 FF 01 02 04 DE 00 FF 01 14 r4.....
000002 42 79 28 55 6E 72 65 67 69 73 74 65 72 65 44 28 By Inregistr...
000003 55 73 45 72 08 FF 02 40 40 6F 78 79 72 69 67 08 User...C
000004 74 28 89 28 31 39 39 38 28 62 79 28 4E 6F 74 65 t. 1998 by No
000005 57 6F 72 74 68 79 28 41 72 74 57 61 72 65 2C 28 Worthy ArtWare,
000006 40 6E 45 2E 00 41 6C 0C 28 52 69 67 68 74 78 28 Inc. All Rights
000007 52 65 79 45 72 74 65 04 00 FF 01 20 47 65 6E 65 Reserved... Gene
000008 72 61 74 45 64 28 62 79 28 4E 6F 74 65 57 6F 72 rated by NetWor
000009 74 68 79 28 A3 6F 6D 78 6F 78 65 72 00 FF 59 02 thy Composer..V.
00000A 01 00 00 FF 58 04 02 02 18 00 00 FF 2F 00 40 54 ..X.....M
00000B 72 68 00 00 01 14 00 FF 21 01 00 00 FF 01 00 55 r4.....
00000C 6E 6E 61 6D 45 64 20 38 30 30 00 FF 00 00 55 6E named-000...In
00000D 6E 61 6D 45 64 20 38 30 30 00 FF 04 00 55 6E named-000...Unn
00000E 61 6D 65 44 20 38 30 30 00 07 7F 00 00 00 00 and-000.....
00000F 00 ..V.....
000010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..G.....G.
000011 28 98 47 6E 5E 98 0A 02 98 46 6E 5E 98 0A 00 ..G.....G.
000012 02 98 47 6E 01 28 98 47 00 28 98 46 6E 5E 98 0A ..Gn...G...Na^H
000013 00 02 98 47 6E 5E 98 47 00 02 98 45 6E 5E 98 0A ..Gn...G...Ea^E
000014 00 02 98 46 6E 5E 98 43 00 02 98 45 6E 5E 98 0A ..Cn^...Ea^E
000015 00 02 98 45 6E 5E 98 45 00 02 98 46 6E 61 28 98 ..En^...Ea^>
000016 3E 00 28 98 43 6E 5E 98 43 00 02 98 43 6E 5E 98 >...G...G...Gn^
000017 43 00 02 98 47 6E 5E 98 47 00 02 98 47 6E 5E 98 C...Gn^G...Gn^
000018 47 00 02 98 0A 6E 01 28 98 0A 00 28 98 47 6E 01 G...Ja...J...Gn
000019 28 98 47 00 28 98 45 6E 5E 98 45 00 02 98 40 4E ..G...Ea^E..Mn
00001A 5E 98 46 00 02 98 47 6E 5E 98 47 00 02 98 45 6E ..H...Gn^G...En
00001B 5E 98 45 00 02 98 43 6E 01 28 98 43 00 28 98 43 ^...G...G...G
00001C 6E 01 28 98 43 00 00 FF 2F 00 40 54 72 48 00 00 n...G.../MTrk...
00001D 01 4E 00 FF 21 01 00 00 FF 01 00 55 6E 6E 01 6D ..M...F.....Unann
00001E 65 64 20 28 30 31 00 FF 00 00 55 6E 6E 01 6D 65 ed-001...Unname
00001F 64 20 30 30 31 00 FF 00 00 55 6E 6E 01 6D 65 G-001...Unname
000020 20 38 30 31 00 01 07 7F 00 01 00 40 00 FF 59 02 -001...G...V.
000021 01 00 00 FF 58 04 02 18 00 00 91 37 6E 5E 01 ..X.....G^?
    
```

(그림 3) 동요 "달"의 미디 파일

선율 정보를 이용한 유사도 계산 알고리즘은 먼저 임

실제 필요로 하는 각각의 정보들이 구해지는 방법은 다음과 같다.

- 키(Key) : 반음올림표(#)가 하나 붙어 있는 G장조를 나타낸다(<표 1> 참조).

<표 1> 미디에서 키 정보 해석

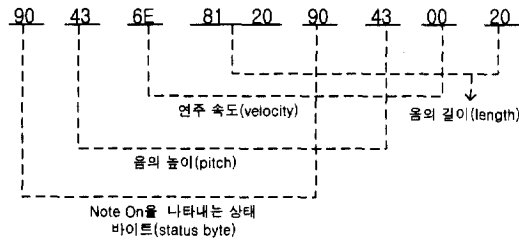
	FF 59 02	01	00
역 할	상태바이트	반음올림 : 양수 반음내림 : 음수	장조 : 00 단조 : 01
분 석	키를 나타내는 상태바이트	반음올림표가 하나 붙음(G)	장조
해 석	반음올림표(#)가 하나 붙어 있는 G장조를 나타냄		

- 박자(Time-meter) : 2/4박자를 나타낸다(<표 2>참조).

<표 2> 미디에서 박자 정보 해석

	FF 58 04	02	02
역 할	상태바이트	박자표의 분자	박자표의 분모 (2의 승수로 표시)
분 석	박자를 나타내는 상태바이트	2	2 ² = 4
해 석	2/4 박자를 나타냄		

- 음높이(pitch)와 길이(length) : 표시된 부분은 동요 "달"에서 첫 음의 높이와 길이를 나타낸 것으로, 높이는 '솔'이고, 길이는 한 박자를 나타낸다 ((그림 4) 참조).



(그림 4) 미디에서의 음의 높이와 길이 해석

이와 같이 구해진 정보들을 이용하여 음악의 특징 요소를 구하게 되는데, 본 논문에서는 음악을 음의 배합으로 정의하기 때문에 음악의 특징이 되는 것은 그 음악의 선율을 구성하는 음들의 집합과 음들의 배합이 될 것이다. 음을 나타내는 값은 앞서 언급한 것과 같은 음의 4가지 특성 중에 음의 구조적 특성인 음높이 값과 음의 길이를 이용한다. 즉, 미디 파일로부터 추출

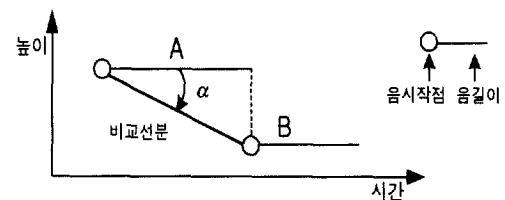
한 각 음표의 음 높이 정보와 음의 길이 정보를 복합적으로 이용한다. 음의 높이 정보와 길이 정보를 복합적으로 이용하기 위하여 연속적인 두 음의 시작점을 연결한 선분(본 논문에서는 비교 선분이라 명명)의 기울기를 유사도 계산에서 이용한다. 다시 말하면, 본 논문에서의 음악의 특징 요소는 음을 나타내는 정보와 음들간 관계를 나타내는 정보들의 집합이라 할 수 있는데, 음을 나타내는 정보는 음들의 상대적 음높이로 표현하고, 음들간 관계를 나타내는 정보는 연속적인 두 음의 시작점을 이은 선분의 기울기로 나타낸다.

본 연구에서 음의 절대적인 높이를 이용하지 않고 음의 상대적 높이를 이용함으로써 음악의 조바꿈이나 또는 음악의 높이 변화에 관계없이 동일한 음악으로 간주한다[10, 11]. 즉, 음악의 특징 요소는 오선지 상에 기보된 음들의 위치에 영향을 받지 않으며, 같은 곡에 대해 조바꿈으로 인해 발생하는 음높이의 변함에 관계없이 음악의 특징 요소는 동일한 값을 나타낸다. 따라서, 음을 나타내는 특징 요소는 실제 음의 높이 값으로 정하지 않고, 식 (1)을 이용하여 평균에 대한 상대적인 높이 값으로 정한다.

$$x = p - \bar{p} \tag{1}$$

- x : 상대적 음높이
- p : 절대 음높이 값
- \bar{p} : 음악의 모든 음표의 음높이 값의 평균

음들간의 관계 정보를 나타내는 비교 선분의 기울기는 (그림 5)와 같이 음 A와 다음 음 B사이에서 형성된 비교선분의 각 α 를 정의한다.



(그림 5) 음 A와 음 B간의 관계를 나타내는 음의 각 α

각 α 는 수평선을 기준으로 하여 시계 반대 방향이 양(+)의 방향, 시계 방향이 음(-)의 방향을 의미하는 방향성을 갖는다. 이를 이용해 두 음의 관계 정보를 $\tan \alpha$ 로 계산한다. 이는 식 (2)처럼 '앞 음 A와 뒤 음 B와의 상대적인 음높이차/앞 음 A의 길이'로 계산한다. 한 음악이

주어지면, 그 음악은 모든 음들의 상대적 음높이 집합과 기울기를 나타내는 $\tan\alpha$ 의 집합으로 특징지을 수 있다.

$$\tan\alpha = \frac{x_B - x_A}{Length_{x_A}} \quad (2)$$

- $Length_{x_A}$: 앞 음 A의 길이
- x_A : 앞 음 A의 상대적 음높이
- x_B : 뒤 음 B의 상대적 음높이

3.3 한계치 설정

본 논문에서 제안된 유사도 계산 알고리즘에서의 유사도는 음악의 선율 정보에 있어서 두 특징인 음을 나타내는 정보(상대적 음높이, x)와 음들간의 관계를 나타내는 정보(기울기, $\tan\alpha$)를 이용한다. 계산된 유사도는 정도의 값이므로, 두 특징의 차 값에 대한 한계치를 설정하고, 정해진 한계치와 대비시켜 유사도를 구한다. 먼저, 음악의 두 특징인 상대적 음높이 x 와 기울기 $\tan\alpha$ 에 대한 최소 한계치와 최대 한계치를 설정한다.

3.3.1 상대적 음높이차에 대한 한계치 설정

두 음의 상대적 음높이차에 대한 유사도의 한계치는 상대적 음높이차의 가장 유사하지 않은, 즉 유사도 0이라 할 수 있는 상대적 음높이차를 이용한다. 음의 상대적 음높이에 대한 한계치는 상대적 음높이의 최소값과 최대값을 의미하며, 최대 상대적 음높이와 최소 상대적 음높이의 차는 두 음의 상대적 음높이차의 한계치가 된다. 즉, 유사도 0이라 할 수 있는 상대적 음높이차가 된다. 상대적 음높이차에 대한 한계치는 식 (3)과 같다. Max_x 는 비교되는 두 곡에 존재하는 모든 음의 상대적 음높이의 최대값이 되고, Min_x 는 비교되는 두 곡에 존재하는 모든 음의 상대적 음높이의 최소값이 된다.

$$Limit_x = Max_x - Min_x \quad (3)$$

- $Limit_x$: 상대적 음높이차에 대한 한계치
- Max_x : 상대적 음높이의 최대값
- Min_x : 상대적 음높이의 최소값
- $Max_x = MAX(Max_{x1}, Max_{x2})$
- $Min_x = MIN(Min_{x1}, Min_{x2})$
- Max_{x1}, Max_{x2} : 각 곡의 최대 상대적 음높이
- Min_{x1}, Min_{x2} : 각 곡의 최소 상대적 음높이

3.3.2 기울기차에 대한 한계치 설정

두 음의 시작점을 이은 선분(비교 선분)의 기울기

($\tan\alpha$)차에 대한 한계치는 기울기차의 가장 유사하지 않은 즉, 유사도 0이라 할 수 있는 기울기차를 이용한다. 이는 두 음악에서 나타날 수 있는 기울기 값 중에서 최대 기울기 값과 최소 기울기 값을 의미하며, 기울기차의 한계치는 최대 기울기 값과 최소 기울기 값간의 차가 된다.

기울기차에 대한 한계치는 식 (4)와 같다. $Max_{\tan\alpha}$ 는 비교되는 두 곡에 존재하는 비교 단위 내에 존재하는 비교 선분의 최대 기울기가 되고, $Min_{\tan\alpha}$ 는 비교되는 두 곡에 존재하는 비교 단위 내에 존재하는 비교 선분의 최소 기울기가 된다.

$$Limit_{\tan\alpha} = Max_{\tan\alpha} - Min_{\tan\alpha} \quad (4)$$

- $Limit_{\tan\alpha}$: 기울기차에 대한 한계치
- $Max_{\tan\alpha}$: 최대 기울기
- $Min_{\tan\alpha}$: 최소 기울기
- $Max_{\tan\alpha} = MAX(Max_{\tan\alpha1}, Max_{\tan\alpha2})$
- $Min_{\tan\alpha} = MIN(Min_{\tan\alpha1}, Min_{\tan\alpha2})$
- $Max_{\tan\alpha1}, Max_{\tan\alpha2}$: 각 곡의 최대 기울기
- $Min_{\tan\alpha1}, Min_{\tan\alpha2}$: 각 곡의 최소 기울기

3.4 정규화

본 논문에서 제안된 유사도 계산 알고리즘은 크게 비교되는 두 음악의 비교 단위를 일치시키는 정규화와 비교 단위내의 음들의 발생 시간을 일치시키는 정규화 과정을 거치게 된다.

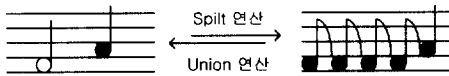
3.4.1 음악의 비교 단위의 일치

일반적으로 음악에는 일정한 흐름의 단절되는 부분이 존재하게 된다. 이는 악구나 악절로 정의되는데, 2마디 또는 4마디로 구성된다. 본 논문에서는 이를 비교 단위(comparative unit)로 정의하고, 비교 단위의 길이는 2마디로 규정하였다. 첫번째 정규화 과정은 이렇게 나뉜 두 곡의 단위박을 고려하여 비교 단위를 일치시키는 것을 의미한다. 예를 들어 비교되는 두 음악이 3/4박자와 6/8박자 곡일 경우, 3/4박자의 음악에서는 4분음표(♩)가, 6/8박자의 음악에서는 8분음표(♪)가 단위박이 된다. 따라서, 3/4박자에서의 4분음표의 길이와 6/8박자에서의 8분음표의 길이는 동일한 것으로 간주한다. 비교되는 두 곡의 정규화는 단위박이 4분음표인 3/4박자곡을 기준으로 하고, 3/4박자곡의 2마디를 비교 단위로 규정하게 된다. 이와 다르게 비교되는 두 음악의 박자표가 동일 내지 배수의 관계에 있지 않으

면, 비교되는 두 비교 단위내의 음 길이를 조정하여 두 비교 단위의 길이를 일치시킨다.

3.4.2 비교 단위내의 음들의 발생 시간의 일치

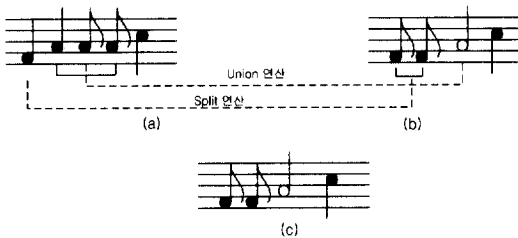
본 논문에서 제안된 유사도 계산 알고리즘은 동일한 시간대에 두 음악에서 나타나는 음의 정보를 이용하여 유사도를 계산하므로, 두 음악의 비교 단위 내에서 발생하는 음의 시간대를 일치시켜야 한다. 이를 위해 선율 정보를 이용한 유사도 계산 알고리즘에서는 Split연산과 Union연산을 이용한다[12].



(그림 6) Split연산과 Union연산

Split연산은 (그림 6)과 같이 한 음을 동일한 음높이를 가지는 여러 음들로 분리하는 연산을 의미한다. 단, 연산 후의 음들의 길이 합은 연산 전의 음 길이 합과 동일해야 한다. Union연산은 Split연산과는 반대로, 연산을 실시하는 대상 음들의 높이가 동일한 경우에 둘 이상의 음들을 한 음으로 합치는 연산을 말한다. Union연산도 Split연산과 동일하게 연산 후의 음 길이 합이 연산 전의 음들의 길이 합과 동일하여야 한다.

(그림 7)은 비교 단위 내에서 Split연산과 Union연산을 이용하여 정규화하여 두 음악의 유사도를 비교하는 과정을 설명한다.



(그림 7) 비교 단위 내의 정규화 예

(그림 7-a)와 (그림 7-c)는 각각 비교할 두 음악의 대상임을 가정한다. Split 연산과 Union 연산을 사용하여 정규화하기 이전에 두 음악은 서로 다른 형태를 갖고 있었으나 (그림 7-a)의 음악 내용을 정규화 과정을 통해서 (그림 7-b)와 같이 정규화를 할 수 있다. 정규

화 이후의 결과인 (그림 7-b)와 비교 대상인 (그림 7-c)를 비교하면 두 음악의 내용은 일치하는 것으로 평가할 수 있음을 알 수 있다.

3.5 유사도 계산

본 절에서는 두 비교 단위 안에 존재하는 음들은 정규화를 거쳐 서로 대응되도록 정규화된 상태에서 비교되는 두 음악에 대한 유사도 계산식을 설명한다. 유사도 계산에서 선율 정보의 요소(음의 높이, 음의 길이) 각각에 대해 유사도를 계산하고, 계산된 값에 일정한 가중치를 부여한다. 가중치가 부여된 계산 값들은 종합되어 비교되는 비교 단위의 유사도 값이 된다. 동요 "달"과 "고드름"의 각 비교 단위별 유사도를 계산한 결과는 <표 3>과 같다. 음영으로 표시한 부분이 각 비교 단위별 최대 유사도를 나타낸다.

<표 3> 동요 "달"과 "고드름"의 유사도 계산 결과표

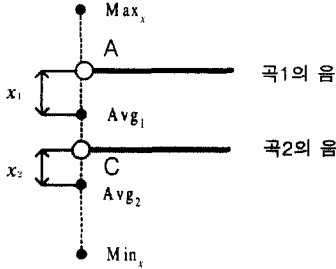
달 \ 고드름	1	2	3	4	5	6
1	0.701	0.724	0.766	0.669	0.669	0.669
2	0.724	0.690	0.775	0.549	0.834	0.609
평균	0.722=(0.724+0.742+0.775+0.586+0.836+0.669)/6					

두 음악에 대한 유사도는 계산된 비교 단위별 유사도 값에서 길이가 긴 곡의 비교 단위별 최대 유사도 값을 평균한 값이 된다. 동요 "달"과 "고드름"의 유사도는 $0.722=(0.724+0.742+0.775+0.586+0.836+0.669)/6$ 이 된다.

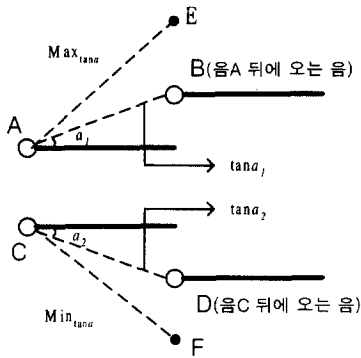
(그림 8)은 음악의 두 특징 요소를 설명한 그림이다. x_1 과 x_2 는 비교되는 두 곡의 상대적 음높이를 나타내고, \tan_{a1} 과 \tan_{a2} 는 비교되는 두 곡의 비교 선분의 기울기를 나타낸다. (그림 8-a)경우의 Max_x 와 Min_x 는 수식 (3)에서 사용된 상대적 음높이에 대한 최대 및 최소 한계치를 나타내고, (그림 8-b)경우의 $\text{Max}_{\tan a}$ 와 $\text{Min}_{\tan a}$ 는 식 (4)에서 사용된 기울기에 대한 최대 및 최소 한계치를 나타낸다.

본 논문에서 제안하는 두 음악에 대한 유사도 계산 알고리즘의 계산식은 아래와 같다. 식에서 사용되는 변수 $\text{Length}(i)$ 는 비교되는 비교 단위내의 i 번째 음의 길이를 나타내고, n 은 비교 단위내의 정규화된 음의 수를 나타낸다. Limit_x 와 $\text{Limit}_{\tan a}$ 는 식 (3), 식 (4)에서 계산된 상대적 음높이차에 대한 한계치와 비교 선분의 기울기차에 대한 한계치를 나타낸다.

- * 음A, B : 곡1의 음들
- * 음C, D : 곡2의 음들
- * Avg₁ : 곡1의 평균 음높이
- * Avg₂ : 곡2의 평균 음높이



(a) 상대적 음높이(x_1, x_2)



(b) 기울기($\tan \alpha_1, \tan \alpha_2$)

(그림 8) 음악의 두 특징 요소

1) 비교 단위내의 대응되는 i 번째 두 음의 상대적 음높이에 대한 유사도($P_Similar_i(x_1, x_2)$) 계산

$$P_Similar_i(x_1, x_2) = 1 - \frac{|x_1 - x_2|}{Limit_x} \quad (5)$$

2) 비교되는 두 비교 단위의 상대적 음높이에 대한 유사도($Unit_WP_Similar(x_1, x_2)$) 계산

$$Unit_WP_Similar(x_1, x_2) \quad (6)$$

$$= \frac{\sum_{i=1}^n (P_Similar_i(x_1, x_2) * Length(i))}{\sum_{i=1}^n Length(i)}$$

3) 비교 단위내의 대응되는 i 번째 두 비교 선분의 기울기에 대한 유사도($I_Similar_i(\tan \alpha_1, \tan \alpha_2)$) 계산

$$I_Similar_i(\tan \alpha_1, \tan \alpha_2) = 1 - \frac{|\tan \alpha_1 - \tan \alpha_2|}{Limit_{\tan \alpha}} \quad (7)$$

4) 비교되는 두 비교 단위의 기울기에 대한 유사도 ($Unit_WI_Similar(\tan \alpha_1, \tan \alpha_2)$) 계산

$$Unit_WI_Similar(\tan \alpha_1, \tan \alpha_2) \quad (8)$$

$$= \frac{\sum_{i=1}^{n-1} (I_Similar_i(\tan \alpha_1, \tan \alpha_2) * Length(i))}{\sum_{i=1}^{n-1} Length(i)}$$

5) 두 특징 요소에 대해 $w_1 : w_2$ 의 가중치를 주어 종합하여 비교되는 두 비교 단위간의 유사도($Complex_Similar$)를 계산한다. 단, 가중치는 음악 전문가의 지식에 따라 변경 가능하다. 그리고, $Unit_1$ 과 $Unit_2$ 는 비교되는 두 음악의 각 비교 단위를 나타낸다.

$$Complex_Similar(Unit_1, Unit_2) \quad (9)$$

$$= \frac{w_1 * Unit_WP_Similar(x_1, x_2) + w_2 * Unit_WI_similar(\tan \alpha_1, \tan \alpha_2)}{w_1 + w_2}$$

6) 정규화가 된 음악의 전체 길이를 기준으로 길이가 긴 곡의 각 비교 단위별 최대 유사도 값을 평균하여 두 음악에 대한 최종적 유사도($Music_Similar$)를 계산한다. 식 (10)의 $Max_Complex_Similar_i$ 는 길이가 긴 곡의 i 번째 비교 단위에 대한 상대 음악의 비교 단위의 최대 유사도를 나타낸다. m 은 정규화가 된 음악의 전체 길이를 기준으로 길이가 긴 곡의 비교 단위 수를 나타낸다.

$$Music_Similar = \frac{\sum_{i=1}^m Max_Complex_Similar_i}{m} \quad (10)$$

4. 실험 및 평가

4.1 실험 환경

본 실험은 Windows 98을 운영체제로 하는 개인용 컴퓨터(Pentium-III 450MHz, 192MB)에서 실행되었으며, Microsoft Visual C++ 6.0으로 구현하였다. 미디 파일은 케이크워크(Cakewalk)[13]를 사용하여 제작하였다. 그리고, 실험에 사용된 두 특징 요소에 대한 가

중치는 1 : 4의 비중으로 하였다. 즉, 음높이에 대한 특징 요소에 1을, 기울기에 대한 특징 요소에 4의 비중을 두어 계산하였다.

4.2 유사도 계산 알고리즘의 정확성 비교 평가

본 절에서는 기존의 내용기반 음악 정보 검색 시스템에서 사용하는 유사도 계산 기법의 정확성과 본 논문에서 제안된 유사도 계산 알고리즘의 정확성을 비교 평가하기 위해, 동요 "달" 원곡과 동요 "달" 안에 존재하는 모든 음들의 음높이를 한 음 올린 곡과 비교 실험을 하였다. 또한, 동요 "달"의 일부분만을 수정한 음악 파일을 실험 데이터로 이용하여 기존의 내용기반 음악 정보 검색 시스템에서 사용하는 유사도 계산 방법과 본 논문에서 제안된 유사도 계산 알고리즘의 정확성을 비교하는 실험을 실시하였다.

"달" 원곡 악보와 한음 올린 곡의 악보는 (그림 9)와 같다.



(a) 동요 "달" 원곡의 악보



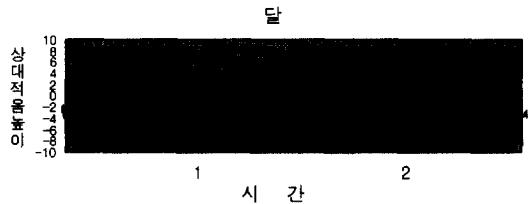
(b) 동요 "달"을 한음 올린 변형된 곡의 악보

(그림 9) 동요 "달"과 한음 올린 "달"의 악보

본 논문에서 제안된 유사도 알고리즘을 이용한 동요 "달" 원곡과 한 음 올린 "달"에 존재하는 모든 음들을 나타낸 그래프와 두 곡의 유사도 계산 결과표는 (그림 10)과 같다. Y축은 곡 안에 존재하는 모든 음들의 평균에 대한 상대적 음높이를 나타내고, X축은 시간축이다. (그림 10-a)에서 그래프 상의 점들은 각 음들의 시작점을 나타낸다. 그래프가 하나밖에 존재하지 않는 이유는 Y축이 평균에 대한 상대적 높이이기 때문에 두 곡에 대한 그래프는 동일하게 나타난다.

동요 "달" 원곡과 한음 올린 변형된 "달"과의 실험을 통해 알 수 있듯이, 본 논문에서 제안된 유사도 계산 알고리즘은 특정 곡에 대해서 전체적으로 변형한

곡과의 유사도는 1로 계산된다. 다시 말해, 특정 곡의 모든 음높이 값을 일정하게 변형시키거나 음길이 값을 일정하게 변형시키더라도 원 곡과의 유사도는 음높이 독립성과 음길이 독립성에 의해 1로 계산된다.



(a) 실험 데이터 그래프

	달	1	2
한음올린달			
1		1.000	0.918
2		0.918	1.000
평균		1.000	
분산		0	

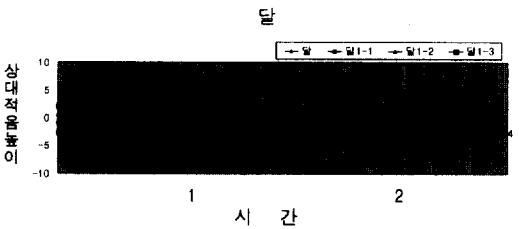
(b) 비교 단위별 유사도 계산표

(그림 10) 동요 "달" 원곡과 변형된 "달"에 대한 유사도 계산

만일, 기존의 내용기반 검색 시스템의 특징 정보 추출과 유사도 계산 기법을 이용하는 경우에도 원곡 동요 "달"과 전체적으로 한 음 올린 "달"과의 유사도 결과 값은 1로써 본 논문에서 제안한 유사도 계산 결과와 동일하다. 즉, 기존의 내용기반 검색 시스템은 음높이의 변화 방향을 이용하여 (그림 9)의 동요 "달" 원곡과 한 음 올린 "달"로부터 같은 문자열 "-USDUDD USDUSUSUDDUDDDS"을 얻을 수 있으며, 이 문자열을 기반으로 유사도 검색을 하면 두 문자열은 정확하게 같은 구성을 갖고 있기 때문에 결과 값으로 1을 갖게 된다. 따라서 음악의 모든 음표의 음 높이를 전체적으로 일관성 있게 변화시킨 것은 원곡을 조바꿈한 것으로 간주하여 기존의 유사도 계산 기법이나 본 논문에서 제안한 기법 모두는 원곡과 동일한 음악으로 판정하게 된다. 이러한 결과는 음악의 유사도 판정의 일반적인 원칙을 따른다고 볼 수 있다.

전체적인 변형을 하는 대신에 일부분을 변형시킨 경우에 실험된 유사도 계산 결과는 다음과 같다. 앞의 실험과 동일하게 동요 "달"에 대해 실험하였다. 동요 "달"의 첫 두 마디를 한 음 올렸을 경우, 두 음 올렸을

경우, 세 음 올렸을 경우 등의 세 경우에 대해 실험하였다. 본 논문에서 제안된 유사도 계산 알고리즘을 이용한 결과는 (그림 11)과 같다. 동요 “달”의 원곡과 변형한 세 곡에 대한 데이터를 그래프로 나타낸 것이 (그림 11-a)이고, (그림 11-b)는 변형된 비교 단위에 대한 유사도를 나타내는 표이다.



(a) 실험 데이터 그래프

	달1-1	달1-2	달1-3
달	0.947	0.907	0.878

(b) 첫번째 비교 단위에 대한 유사도 계산표

(그림 11) 동요 “달”의 원곡과 변형한 3곡의 유사도 계산

(그림 11)의 달1-1, 달1-2, 달1-3은 원곡 “달”의 첫마디 내에 존재하는 음들의 음높이를 한 음, 두 음, 세 음 올려서 각각 변형한 곡을 나타낸다. (그림 11-b)에서 알 수 있듯이, 동요 “달”의 원곡에 대해 변형을 많이 가하면 가할수록 유사도가 그만큼 작아짐을 알 수 있다.

기존의 내용기반 음악 정보 검색 시스템에서는 세 가지 실험 조건에 대해 모두 앞선 실험의 결과와 동일한 “-UUSDUDDUSDUSUSUDDUDDDS”를 특징 정보로 갖게 되며, 따라서, 변형된 곡들도 원곡과 동일한 곡으로 인식하게 된다. 그러나, 원곡 “달”의 첫마디 내에 존재하는 음들의 음높이를 한 음, 두 음, 세 음 올려서 일부가 변형된 각각의 곡들은 실제로 원곡과는 상이한 선율의 형태를 가짐을 확인할 수 있다.

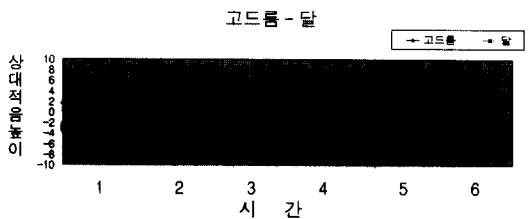
비교 실험을 통하여 Query by Humming, Melody Index, 음표 열 탐색 알고리즘을 사용하는 음악 정보 검색 시스템과 같은 기존의 내용기반 음악 정보 검색 시스템에서 선율에 대한 특징 정보 추출과 추출된 특징 정보를 이용한 검색 방법은 서로 다른 곡에 대해서도 정확히 일치하는 곡이라는 잘못된 판정을 내리는 부정확성을 가짐을 알 수 있다. 그러나, 본 논문에서

제안된 유사도 계산 알고리즘을 사용할 경우에는 보다 정확한 특징 정보의 추출과 유사도 계산 알고리즘으로 기존 시스템의 부정확한 검색 문제를 보완하고, 서로 다른 곡들간의 정확한 유사도를 계산할 수 있음을 알 수 있다.

4.3 제안된 유사도 계산 알고리즘의 일반 실험

일반적인 경우의 동요에 대한 유사도 계산 실험을 위해서 동요 “달”, “고드름”, 그리고 “가을밤”에 대해 실험을 하였다. “고드름”과 “달”, “가을밤”과 “달”, “가을밤”과 “고드름” 등의 3경우로 나누어서 실험하였다. 결과표는 2마디를 비교 단위로 하여 모든 비교 단위별 유사도를 계산하였다. 각 인덱스는 비교 단위에 대한 인덱스이다.

(그림 12)는 동요 “고드름”과 “달”의 음높이와 음의 시작점을 연결한 그래프(a)와 계산된 결과표(b)이다. “달”의 비교 단위가 2개이고, “고드름”의 비교 단위는 6개이다. 따라서, (그림 12-a)와 같이 6개의 비교 단위별 실험 결과 값이 계산된다. (그림 12-b) 테이블에서 가로축은 “고드름”의 비교 단위 인덱스이고, 세로축은 “달”의 비교 단위 인덱스이다. 그리고, 음영 부분은 “고드름”을 기준으로 하여 각 비교 단위 별 최대 유사도를 가지는 부분을 나타내는 것이다.



(a) 실험 데이터 그래프

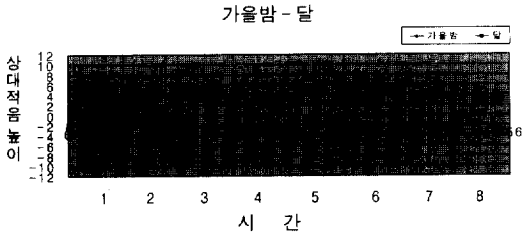
	고드름	1	2	3	4	5	6
달							
1		0.701		0.766			
2			0.690		0.549	0.834	0.609
평균		0.722	(0.724+0.742+0.775+0.586+0.836+0.669)/6				

(b) 비교 단위별 유사도 계산표

(그림 12) 동요 “고드름”과 “달”에 대한 유사도 계산

(그림 13)은 동요 “가을밤”과 “달”의 음높이와 음의 시작점을 이은 그래프(a)와 계산된 결과표(b)이다. “달”

의 비교 단위가 2개이고, “가을밤”의 비교 단위는 8개이다. 따라서, (그림 13-b)와 같이 8개의 비교 단위별 실험 결과 값이 계산된다.



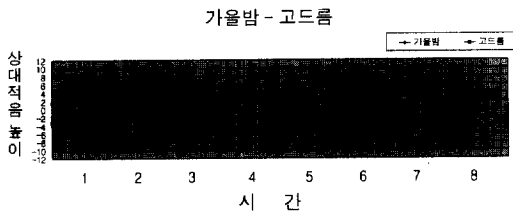
(a) 실험 데이터 그래프

달 \ 가을밤	1	2	3	4	5	6	7	8
1	0.764		0.796					
2		0.616		0.668	0.672	0.764	0.672	0.668
평균	0.729=(0.764+0.658+0.800+0.719+0.685+0.809+0.677+0.719)/8							

(b) 비교 단위별 유사도 계산표

(그림 13) 동요 “가을밤”과 “달”에 대한 유사도 계산

(그림 14)는 동요 “가을밤”과 “고드름”의 음높이와 음의 시작점을 연결한 그래프(a)와 계산된 결과표(b)이다. “가을밤”의 비교 단위가 8개이고, “고드름”의 비교 단위는 6개이다. 따라서, (그림 14-b)와 같이 8개의 비교 단위별 실험 결과 값이 계산된다.



(a) 실험 데이터 그래프

고드름 \ 가을밤	1	2	3	4	5	6	7	8
1	0.643		0.619		0.495	0.672		
2	0.656	0.444	0.671	0.455	0.625	0.501	0.612	0.455
3	0.677	0.459		0.525		0.730	0.588	0.525
4	0.512	0.529	0.439	0.517	0.347	0.429	0.521	0.517
5		0.547	0.824	0.593	0.770		0.633	0.593
6	0.510	0.405	0.553	0.362	0.532	0.421	0.554	0.362
평균	0.736=(0.726+0.708+0.890+0.675+0.827+0.749+0.636+0.675)/8							

(b) 비교 단위별 유사도 계산표

(그림 14) 동요 “가을밤”과 “고드름”에 대한 유사도 계산

(그림 12)에서 “달”의 2번째 비교 단위(5-8번째 마디)와 “고드름”의 4번째 비교 단위(7-8번째 마디)의 유사도(0.549)가 가장 낮고, “달”의 1번째 비교 단위(1-4번째 마디)와 “고드름”의 5번째 비교 단위(9-10번째 마디)간의 유사도(0.836)가 가장 높음을 보인다. (그림 12-a)그래프에서 보듯이 “달”의 2번째 비교 단위와 “고드름”의 4번째 비교 단위가 많이 다름을 우리의 육안으로도 확인할 수 있다. 이에 반해 “달”의 1번째 비교 단위와 “고드름”의 5번째 비교 단위가 많이 유사함을 볼 수 있다. (그림 14)를 살펴보아도 결과와 동일함을 볼 수 있다. 유사도가 가장 낮은 비교 단위는 “고드름”의 4번째 비교 단위와 “가을밤”의 5번째 비교 단위가 되고, 유사도가 가장 높은 비교 단위는 “고드름”의 3번째 비교 단위와 “가을밤”의 3번째 비교 단위가 된다.

5. 결 론

멀티미디어 정보 검색 시스템에서 사용자의 편리한 정보 검색 서비스 및 정확한 정보 검색을 제공하기 위해서는 내용 기반의 정보 검색을 지원해야 한다. 최근 까지 내용 기반의 멀티미디어 정보 검색 시스템에 대한 연구는 텍스트 및 이미지 정보에 대한 내용 기반 검색에 대한 연구가 대부분이었다. 그러나 점진적으로 인터넷 환경에서 음악 정보의 검색 서비스가 활발하게 진행되고 있으므로 음악 정보에 대한 내용 기반의 정보 검색 기법이 필요하다. 음악 정보의 내용 기반 검색이 가능하기 위해서는 음악 파일간의 유사도를 음악의 선율 정보를 기반으로 판단할 수 있는 유사도 계산 알고리즘이 필요하다.

본 논문에서는 미디(MIDI) 형식의 음악 파일로부터 음높이와 길이 정보를 추출하여 음악의 특징 요소를 정의하고 이를 이용하여 두 음악 파일간의 유사도를 계산하는 알고리즘을 제안하였다. 즉, 음의 선율 정보를 나타내는 상대적 음높이와 음의 길이를 반영하여 음들간의 관계 정보를 나타내는 비교 선분의 기술기를 이용하여 두 음악간의 유사도를 계산하는 알고리즘을 제안하였다. 제안된 알고리즘은 두 음악 전체에 대해 비교하기보다는 음악을 흐름의 측면에서 비교 단위로 구분하여 비교한다. 제안된 알고리즘에 대한 타당성 검증을 위해 동요 3곡을 선정하였고, 선정된 동요 3곡

을 들썹 묶어 유사도를 계산하였다. 실험 결과에 따르면 기존의 Query by Humming, Melody Index, 음표열 탐색 알고리즘을 사용하는 음악 정보 검색 시스템 등과 같은 기존의 내용기반 음악 정보 시스템보다 본 논문에서 제안된 유사도 계산 알고리즘이 타당성과 정확성을 가짐을 검증해 주었다. 따라서 본 연구에서 제안된 유사도 계산 알고리즘은 음악 정보 검색 시스템에서 내용기반 정보 검색을 지원하기 위한 선율 정보기반의 유사도 계산 알고리즘으로 평가할 수 있다.

향후 연구 과제로는 본 논문의 유사도 알고리즘을 이용하여 동요뿐만 아니라 동요보다 더 복잡한 가요, 팝, 그리고 클래식 음악에 대해서도 적용시켜 실험할 필요가 있을 것이다. 또한, 본 논문에서 제안된 유사도 계산 알고리즘을 이용한 내용기반 음악 정보 검색 시스템을 개발함으로써 제안된 유사도 계산 알고리즘의 유용성을 보일 수 있다.

참 고 문 헌

[1] 양철웅, 양우석, 이윤준, 김명호 “멀티미디어 데이터베이스 기술”, 한국정보과학회지, 제14권 제9호, 1996.
 [2] 최기호, 문회정, 염성주, 김우생, “내용을 기반으로 한 이미지 검색 데이터베이스 시스템”, 한국정보과학회지 제13권 제1호, 1995.
 [3] 이정규, 오해석, “선율을 이용한 음악 정보 검색 시스템의 설계 및 구현”, 정보처리논문지, 제5권 제1호, 1998.
 [4] 조효임, 이동남, 주대창, 새로운 음악통론, 학문사, 1998.
 [5] 조봉행, 음악 이론과 용어 해설, 다라, 1995.
 [6] Siegmund Levarie, Ernst Levy, 음이란 무엇인가, 삼호, 1988.
 [7] Ghias, A., Logan, J., Chamberlin, D. and Smith, B. C., “Query by Humming : Musical Information Retrieval in an Audio Database,” Proc. of ACM Multimedia'95, 1995.
 [8] Rodger J. McNab, Lloyd A. Smith, David Bainbridge and Lan H. Witten, “The New Zealand Digital Library MELody inDEX,” D-Lib Magazine, 1997.

[9] 김종화, MIDI 총론, 세운, 1994.
 [10] 유재수, “내용 기반 이미지 검색을 위한 이미지 DB 기술 동향”, '99 바다 사용자 그룹 워크샵, 1999.
 [11] Guojun Lu, “An approach to image retrieval based on shape,” Journal of Information Science, 1997.
 [12] Davood Rafiei and Alberto Mendelzon, “Similarity -Based Queries for Time Series Data,” Proc. of ACM SIGMOD, pp.13-25, 1995.
 [13] CAKEWALK music software, CAKEWALK Pro Audio 8 User's Guide, cakewalk, 1998.



모 종 식

e-mail : jsmo@easy.co.kr
 1998년 인하대학교 전자계산공학 (공학사)
 2000년 인하대학교 대학원 전자계산공학과(공학석사)
 1999년~현재 (주)이지시스템 부설연구소 연구원

관심분야 : 데이터베이스, 멀티미디어



김 소 영

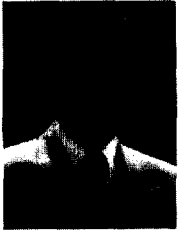
e-mail : g1991278@inhavision.inha.ac.kr
 1999년 인하대학교 전자계산공학과(공학사)
 2001년 인하대학교 대학원 전자계산학과(공학석사)
 관심분야 : 멀티미디어 데이터베이스, 정보검색



구 경 이

e-mail : g2001419@inhavision.inha.ac.kr
 1997년 인하대학교 전자계산공학과(공학사)
 1999년 인하대학교 대학원 전자계산학과(공학석사)
 2000년 현재 인하대학교 대학원 전자계산공학과 박사과정

관심분야 : 이동 컴퓨팅, 데이터베이스, 정보 검색



한창호

e-mail : chhan@inha.ac.kr
 1980년 성균관대학교 전자공학과
 (공학사)
 1982년 서울대학교 전자공학과
 (공학석사)
 1991년 The Univ. of Texas at Austin
 전기및컴퓨터공학과
 (공학박사)

1983년~1986년 한국전자통신연구소 연구원
 1991년~1992년 Cadence Design Systems, Senior Member
 of Technical Staff
 1992년~현재 인하대학교 공과대학 전자계산공학과
 부교수
 관심분야 : 컴퓨터음악, 컴퓨터그래픽스, 설계자동화



김유성

e-mail : yskim@inha.ac.kr
 1986년 인하대학교 전자계산
 공학과(이학사)
 1988년 한국과학기술원 전산학과
 (공학석사)
 1992년 한국과학기술원 전산학과
 (공학박사)

1990년~1992년 삼성전자 컴퓨터부문 주임 연구원
 1996년~1997년 미국, 퍼듀대학교 전산학과 방문연구원
 1992년~현재 인하대학교 전자계산공학과 부교수
 관심분야 : 멀티미디어 정보검색, 이동 데이터베이스