

음원의 주파수 변화율과 코사인 유사도 알고리즘을 이용한 음악 검색 시스템 개발

송지원*, 임은주**, 하성윤**, 우균*
*부산대학교 전자전기컴퓨터공학과
**부산대학교 정보컴퓨터공학부

e-mail:{jiwon, woogyun}*@pusan.ac.kr, eunjoo1991**@hanmail.net, hasy52**@naver.com

Development of Music Information Retrieval System Using Differentiation of Frequency and Cosine Similarity Algorithm

Ji Won Song*, Eun Joo Lim**, Seong Yoon Ha**, Gyun Woo*

*Dept of Electronic and Computer Engineering, Pusan National University

**Dept of Computer Science Engineering, Pusan National University

요 약

대중음악과 스마트폰 기술이 발달하면서 사용자가 직접 음악을 검색할 수 있는 내용 기반 음악 검색 기술이 연구되었다. 그 결과 허밍을 사용하여 음악을 검색할 수 있는 음악 검색 시스템이 개발되었지만, 검색 속도가 느리고 검색 결과가 부정확한 시스템이 많다. 본 논문에서는 음원의 주파수 변화율을 측정하고 이를 코사인 유사도 알고리즘을 이용하여 유사도를 측정하는 음악 검색 시스템을 설계하였고, 각 설계요소를 설명한다. 새로 설계한 음악 검색 시스템을 기반으로 한 실험을 통하여 기존의 음악 검색 시스템과 유사한 성능이 나오는 것을 확인하였으며 본 논문에서 제시한 새로운 음악 검색 시스템은 기존 음악 검색 시스템보다 구조가 단순하면서도 유사한 결과를 내고 있다.

1. 서론

멀티미디어 기술과 스마트폰 환경이 발달하면서 음악 Tag를 사용한 것이 아닌 음악 그 고유 데이터를 사용한 음악 검색 시스템이 발달하였다. 특히 사용자가 직접 노래를 불러(Humming) 음악을 검색할 수 있는 내용 기반 음악 검색 기술에 대한 연구가 시작되었다[1]. 이를 이용하여 음악 데이터를 일부분의 허밍 데이터를 이용하여 음악 전체 결과를 추출해내는 음악 검색 시스템이 있다.

허밍 데이터를 이용하여 음악을 검색하는 기존 시스템으로 네이버 음악검색[2], SoundHound[3] 등이 있다. 이 두 시스템은 주파수를 사용하여 원하는 음악 정보를 얻을 수 있으나, 그 구조가 복잡하고 검색 속도가 느려 원하는 결과를 얻기까지 오랜 시간이 걸린다. 그리고 허밍 데이터와 원곡 데이터의 음과 일치율이 떨어질 때 전혀 엉뚱한 결과가 나올 수 있다.

본 논문에서 새로운 음악 검색 시스템을 제안한다. 허밍 데이터를 입력받아 문서 유사도 알고리즘 중 하나인 코사인 유사도 알고리즘을 검색 알고리즘으로 사용한다. 검색 결과가 나오기까지 복잡한 과정을 거치는 기존의 시스템과 달리 이 시스템은 검색 과정을 단순화한 동시에 기존의 음악 검색 시스템과 유사한 결과를 낸다.

본 논문의 구성으로는 서론, 관련 연구, 음악 검색 시스템 설계 및 구현, 실험 및 결과, 결론 및 향후 과제로 구성되어 있다. 1장은 서론으로서 문제제기 및 본 논문을 소개한다. 2장은 관련 연구로서 내용 기반 음악 검색 시

템, Query by Humming, 음악 검색 알고리즘을 설명한다. 3장에서는 음악 검색 시스템 설계 및 구현이며, 위 논문에서 새로 제안하는 음악 검색 시스템의 개발 설계도와 각 설계 요소를 설명한다. 4장은 실험 및 결과를 작성하였으며 설계도를 토대로 한 음악 검색 시스템을 구현한 후 간단한 실험을 통하여 시스템의 성능을 증명한다. 5장인 결론 및 향후 과제에선 본 논문을 정리함과 동시에 시스템의 한계점을 정리하고, 또한 음악 검색 시스템의 활용 방안을 제시한다.

2. 관련 연구

2-1. 내용 기반 음악 검색 시스템

허밍 데이터를 이용한 음악 검색 시스템은 음악 태그를 사용하는 기존의 음악 검색 방법을 사용하지 않고 내용 기반 음악 검색에 초점을 둔다[4]. 내용 기반 음악 검색은 사용자가 직접 노래를 부르거나 오디오 파일을 재생하여 음악을 검색하는 방법이다[5]. 내용 기반 음악 검색 방법은 사용자가 일부 음악 데이터를 이용하여 원본 음악 데이터를 찾을 수 있다.

내용 기반 음악 검색 방법은 음악 태그와 가사 데이터가 없어도 음악 검색을 할 수 있다는 장점이 있다. 내용 기반 음악 정보 검색 시스템에 대한 연구가 활발히 이루어졌다. 색인 검색을 이용한 음악 정보 검색 시스템에 관한 연구와 동적 시간 왜곡을 이용한 연구가 있으며, 유전자 알고리즘을 음악 검색에 이용하려는 연구가 있다[6-8].

2-2. Query by Humming

Query by Humming이란 허밍 데이터를 이용하여 전체 데이터를 찾는 내용 기반 음악 검색 시스템의 한 부분이다[9]. Query by Humming은 사용자가 내는 허밍 데이터는 원본 데이터의 일부로 간주할 수 있다. 따라서 원본 데이터가 포함된 데이터베이스로부터 원본 데이터의 특정한 부분과 허밍 데이터와 일치 여부를 검색 알고리즘을 사용하여 그 결과를 도출해 낸다.

2-3. 음악 검색 알고리즘

내용 기반 음악 검색 시스템을 구현하기 위하여 다양한 검색 알고리즘을 사용할 수 있다. 기존의 시스템에서 사용하는 음악 검색 알고리즘은 동적 시간 왜곡을 사용하는 방법[7], 유전자 검색 알고리즘을 사용하는 방법[8] 지문 검색 알고리즘을 사용하는 방법 등이 있다[12].

동적 시간 왜곡이란 서로 다른 시간 데이터를 물리적 거리에 따라 시간 흐름을 왜곡하여 최적의 정렬을 구하는 방법이다[10]. 동적 시간 왜곡은 동적 계획법(Dynamic Programming)을 사용하여 각 멜로디 간의 거리를 계산하는 행렬을 구하여 두 음악 데이터의 유사도를 계산할 수 있다.

유전자 알고리즘이란 자연계에서 적자생존의 원리에 따라 세대가 지나면서 우수한 형질을 지닌 개체가 생성되는 일반적인 유전의 법칙을 사용한 알고리즘이다[11]. 입력 음악 데이터가 원본 음악 데이터의 유전자로서 일치하는지 유전자 검색 알고리즘을 사용한다. 이는 검색 결과를 정확하게 찾을 수 있지만, 입력 데이터에 변형이 가해지면 검색 결과가 현저히 다르게 나타날 수 있다.

음악 검색에서 사용하는 지문법은 음악 인식을 위한 지문 정보를 FFT(Fast Fourier Transform)를 거쳐 데이터베이스에 저장한 후 입력데이터의 지문 정보와 비교하여 결과를 내는 기술이다[12]. 음악 데이터가 FFT를 거친 지문 정보는 유일한 데이터이므로 두 음악간 유사도를 평가하기 수월하다. 지문 검색 알고리즘은 음악 검색과 저작권 필터링 등에 활용된다[13].

3. 음악 검색 시스템의 설계 및 구현

이 논문에서 제안하는 음악 검색 시스템의 시스템 설계도는 그림 1과 같다. 이 시스템은 안드로이드 애플리케이션과 검색 서버로 나누어져 있다. 안드로이드 서버는 허밍 데이터를 입력받아 서버로부터 허밍 데이터를 보내고, 서버로부터 검색 결과를 받아 이를 출력해주는 부분이다. 서버 부분에는 안드로이드 애플리케이션으로부터 허밍 데이터를 입력받아 유사도 분석을 통하여 가장 유사한 검색 결과를 애플리케이션으로 전송해준다. 각 부분에 대한 설명은 아래와 같다.

3-1. 허밍 데이터 입력

시스템은 안드로이드 운영체제 기반의 마이크에서 허밍 데이터를 입력받아 RAW 파일(헤더가 없는 가공되지

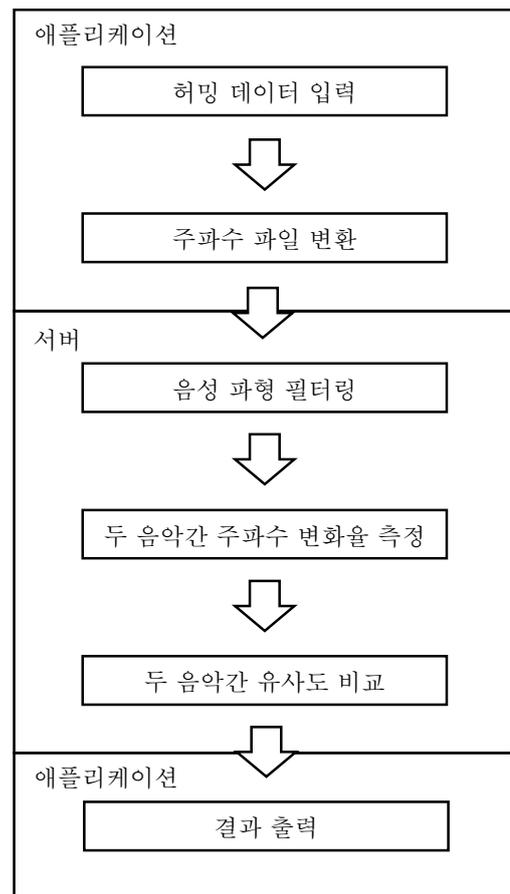
않은 파일)을 생성한다. 그 후 안드로이드 애플리케이션에서 RAW 파일을 WAV 파일(Waveform Audio Format)로 변환한 후 이를 서버로 전송한다.

3-2. 주파수 파일 변환

WAV 파일의 유사도를 분석하기 위하여 주파수를 저장한 데이터 파일로 변환한다. WAV 파일은 음성을 구성하는 여러 가지 파형의 합성 파의 가장 긴 주기를 간격으로 푸리에 변환(Fourier Transform)을 한다. 푸리에 변환으로 가청 주파수(80 ~ 1500Hz) 내 음성 파형을 뽑아낸다.

3-3. 음성 파형 필터링

푸리에 변환을 통해서 가청 주파수 내 음성 파형 중 가장 낮은 주파수 대역을 추출한다. 가장 낮은 주파수 대역 중 주파수가 없는 부분은 바로 직전 주파수의 값으로 치환한다. 만약 한 옥타브 이상 주파수만큼 차이가 날 경우에는 잡음으로 치환하여 해당 주파수를 이전 주파수 값으로 치환한다.



(그림 1) 음악 검색 시스템 설계도

3-4. 두 음악간 주파수 변화율 측정

필터링을 거친 음성 파형을 이용하여 해당 파형 주파수의 변화율을 측정한다. 파형의 주파수를 특정 시간 간격으로 나누었을 때, 해당 음의 주파수와 이전 음의 주파수

와 비교하여 한음 이상 변화가 보일 경우 그 변화를 벡터로 저장한다.

3-5. 두 음악간 유사도 비교

푸리에 변환으로 뽑아낸 음성 파형을 서버로 전송한 후 일정한 시간 간격으로 나누어 각 시간 사이의 주파수 변화량을 벡터 데이터로 저장한다. 음성 파형과 비교할 음악 데이터 역시 주파수 변화율을 측정하여 벡터 데이터로 저장한다. 원곡 음악 데이터와 음성 파형 데이터를 코사인 유사도 비교 알고리즘을 이용하여 두 음악 간의 유사도를 측정한다.

코사인 유사도는 내적 공간 두 벡터 간 각도의 코사인 값을 이용하여 측정된 벡터 사이의 유사한 정도를 의미한다[14]. 코사인 유사도는 정보 검색 및 텍스트 마이닝 분야에서 서로 다른 두 문서 데이터 사이의 유사도를 측정하는 방법이다.

$$S_m(A,B) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (1)$$

두 음악간 유사도를 S_m 이라고 하고 두 음악을 각각 A, B 라고 하였을 때 두 음악 간의 유사도는 식 1과 같이 계산된다. 이렇게 계산된 유사도는 -1에서 1 사이 값을 가지며, -1은 서로 완전히 반대되는 경우, 0은 서로 독립적인 경우, 1은 서로 완전히 같은 경우를 의미한다. 따라서 코사인 유사도의 양수 값으로 서로 유사하지 않은 경우와 유사한 경우를 가릴 수 있다. 또한, -1에서 1 사이의 값으로 두 음악이 서로 얼마나 유사한지 알 수 있다.

3-6. 결과 출력

음성 데이터와 음악 데이터베이스 간 모든 음악 데이터를 위의 과정을 통하여 유사도를 측정한다. 그리고 그중 가장 유사도가 높은 상위 3곡의 음악 데이터를 찾는다. 유사도 비교 과정에서 찾아낸 가장 유사도가 높은 상위 3곡의 음악 데이터의 정보를 안드로이드 애플리케이션으로 전송하여 이를 사용자에게 전달한다.

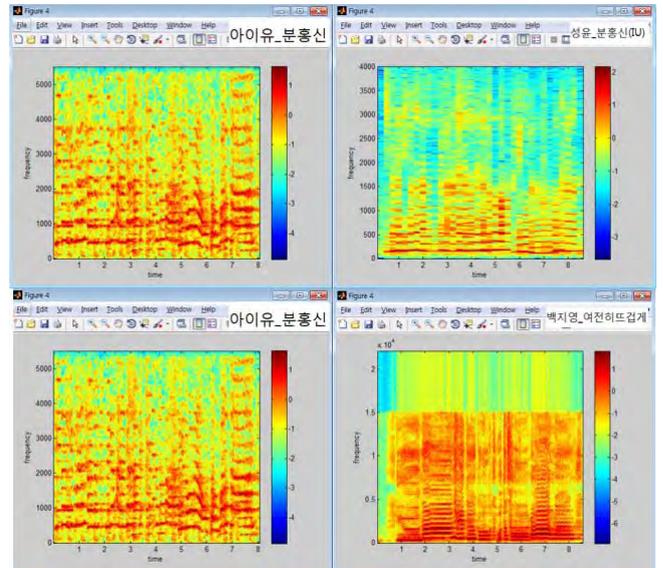
4. 실험 및 결과

이 시스템의 설계도를 기반으로 한 음악 유사도 검증 시스템을 만든 후 실험을 하였다. 실험에서 사용된 검증 테스트 셋은 표 1과 같다.

<표 1> 음악 유사도 검증 테스트 셋

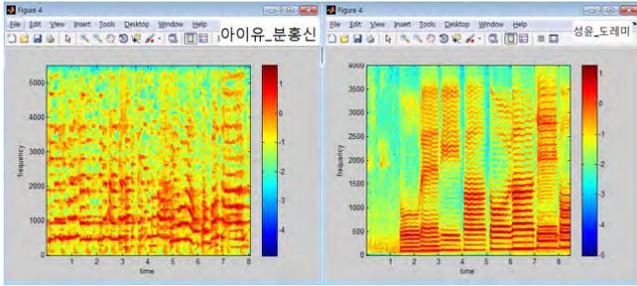
구분	곡명
원곡	아이유 - 분홍신
Testset 1	원곡을 따라부른 남학생의 허밍 데이터
Testset 2	백지영 - 여전히 뜨겁게
Testset 3	원곡과 다른 멜로디 허밍 데이터

먼저 원곡 “아이유-분홍신”과 이를 따라 부른 남학생의 허밍 데이터 간 유사도는 0.414만큼 일치한다. 하지만 입력 데이터를 다른 가수의 노래 “백지영-여전히 뜨겁게”를 입력할 경우 원곡과 남학생 허밍 데이터 간 유사도 수치보다 낮은 0.113을 기록하였다.



(그림 2) 원곡(아이유-분홍신)과 허밍 데이터 간 유사도 비교 분석 화면(위), 원곡과 다른 음악(백지영-여전히 뜨겁게)간 유사도 비교 분석 화면(아래)

그리고 원곡과 다른 멜로디로 허밍을 한 경우는 비슷한 멜로디로 허밍을 준 경우보다 유사도가 낮게 나온다. 그림 3은 원곡(아이유-분홍신)과 다른 멜로디를 부른 남학생 허밍 데이터 간 유사도 비교 결과 화면이다. 그림 3의 경우 유사도가 -0.169로 코사인 유사도 분석 알고리즘의 결과로 위 두 음악은 서로 다른 멜로디의 음악임을 알 수 있다.



(그림 3) 원곡(아이유-분홍신)과 다른 멜로디 허밍 데이터 간 유사도 분석 결과

5. 결론

본 논문에서는 허밍 데이터와 원곡 음악 데이터를 각 주파수의 변화율을 측정한 후 이를 코사인 유사도 비교 알고리즘을 이용한 내용 음악 검색 시스템을 설계하고 이를 바탕으로 한 유사도 검증 시스템을 구현하였다. 실험 결과, 원곡과 허밍 데이터 간의 비교를 통해 본 논문에서 제시한 음악 검색 시스템은 기존의 음악 검색 시스템과 유사한 결과를 추출해 낼 수 있음을 알 수 있다. 즉, 본 논문에서 제시한 음악 검색 시스템은 코사인 유사도 비교 알고리즘을 사용하여 복잡한 기존 시스템의 구조를 간단히 함과 동시에 기존 시스템과 유사한 결과를 추출함을 알 수 있다.

이 시스템은 톤이 다른 두 음악간 검색 결과는 수월하게 나올 수 있으나, 톤이 비슷한 두 음악간 검색 결과는 실제 유사하게 나올 수 없는 상황에도 높은 유사도를 보이며 검색결과에 나온다. 여학생이 노래를 부른 허밍 데이터와 같은 여학생이 단순히 글을 읽은 허밍 데이터 간 유사도는 비교적 높은 0.276의 유사도를 보인다. 이는 목소리 톤이 비슷할 경우 정확한 결과를 내기 힘들다는 것을 의미한다. 따라서 비슷한 톤의 허밍 데이터 간 유사도 측정을 정확히 할 수 있는 연구가 필요하다.

이 시스템은 두 음악의 주파수 수치를 비교하는 것이 아니라 주파수의 변화율을 비교한다. 즉 두 음악간 멜로디 변화를 이용하여 유사도를 측정하기 때문에 위 논문에서 제안한 음악 검색 시스템 외에도 서로 다른 여러 음악간 표절 여부를 가릴 수 있는 측도가 될 수 있다. 또한, 위 시스템을 활용하여 유사도가 높은 서로 다른 여러 음악을 추천할 수 있는 음악 추천 시스템을 구축할 수 있다.

참고문헌

[1] Yunyue Zhu, Dennis Shasha, "Warping Indexes with Envelope Transforms for Query by Humming.", Proceedings of the 2003 ACM SIGMOD international conference on Management of data, pp.181-192, ACM, 2003.
 [2] 박소연, "주요 포털들의 멀티미디어 검색 서비스 비교 분석", 한국문헌정보학회지, 제44권, 제4호, pp.395-412,

2010

[3] SoundHound Inc, SoundHound, <https://www.soundhound.com/>
 [4] Zhendong Zhao, Xinxi Wang, Qiaoliang Xiang, Andy M. Sarroff, Zhonghua Li, Ye Wang, "Large-scale music tag recommendation with explicit multiple attributes.", Proceedings of the international conference on Multimedia, pp. 401-410, ACM, 2010.
 [5] 노정순, "내용기반 음악검색 시스템의 비교 분석", 한국정보관리학회, 정보관리학회지, 제30권, 제3호, pp.23-48, 2013.
 [6] 구경이, 임상혁, 이재현, 김유성, "주제 선을 색인을 이용한 내용 기반 음악 정보 검색 시스템", 한국정보과학회, 데이터베이스 소사이어티 데이터베이스 연구 제19권, 제3호, pp.34-45, 2003.
 [7] Ning Hu, Roger B. Dannenberg, and George Tzanetakis, "Polyphonic audio matching and alignment for music retrieval.", Computer Science Department, 2003.
 [8] Seungmin Rho, Ning Hu, Eenjun Hwang, and Minkoo Kim. "Music information retrieval using a GA-based relevance feedback.", Multimedia and Ubiquitous Engineering, 2007. MUE'07. International Conference on, pp.739-744, IEEE, 2007.
 [9] Asif Ghias, Jonathan Logan, David Chamberlin, Brian C. Smithl. "Query by humming: musical information retrieval in an audio database.", Proceedings of the third ACM international conference on Multimedia, pp.231-236. ACM, 1995.
 [11] 이준승, 조성배, "대화형 유전자 알고리즘을 이용한 감성적 음악검색", 한국정보과학회 학술발표논문집, 제29권, 제2호. pp.175-177, 2002.
 [12] Avery Li-Chun Wang. "An Industrial Strength Audio Search Algorithm." ISMIR. 2003.
 [13] 정형용, 김준형, 박현민, 이정준, "하둡을 이용한 내용 기반 음악 시스템의 설계", 한국정보과학회, 학술발표논문집, 제38권, 제1호(B), pp.377-380, 2011.
 [14] Wikipedia. Cosine Similarity. http://en.wikipedia.org/wiki/Cosine_similarity, 2014년 10월 1일 방문
 [15] Wikipedia. MIDI. <http://en.wikipedia.org/wiki/MIDI>, 2014년 8월 18일 방문
 [16] 문화기술(CT) 심층리포트, 한국콘텐츠진흥원, 2012, 2014년 10월 6일 방문