

멜로디 라인의 변곡점을 활용한 커버곡 검색 알고리즘

이보현, 김 명
이화여자대학교 컴퓨터공학과
Leeboh23@ewhain.net, mkim@ewha.ac.kr

Cover Song Search Algorithm Using Inflection Points of The Melody Lines

Bo-Hyun Lee, Myung Kim
Dept. of Computer Science and Engineering, Ewha Womans University

요 약

동영상 공유 플랫폼의 발전으로 다양한 영상들이 업로드 되고 있는데, 영상에 사용된 음악의 저작권 행사를 위해서는 음악 검색 알고리즘의 역할이 매우 중요하다. 그러나, 커버곡과 같이 원곡을 변형하여 사용한 경우에는 단순한 음악 검색 알고리즘으로는 이러한 상황을 탐지하기가 쉽지 않다. 음악의 높낮이나 빠르기 등이 바뀌어도 검색 성능이 유지되어야 하지만, 현재까지 연구된 알고리즘들은 이러한 변화에 취약하기 때문이다. 이에 본 연구에서는 멜로디 라인의 변곡점을 활용한 커버곡 검색 알고리즘을 제안한다. 본 알고리즘은 멜로디 라인을 중심으로 하여 높낮이 변화에도 높은 성능을 유지하고, 변곡점을 이용하기 때문에 빠르기 변화에 강하고 사용되는 메모리 양은 적다는 장점이 있다.

1. 서론

동영상 공유 플랫폼의 발전으로 인해 수많은 개인 영상 제작자들이 생겨났고, 폭발적으로 많은 다양한 종류의 영상들이 업로드되는 실정이다. 이러한 영상의 배경음이나 콘텐츠로 쓰이는 음악의 저작권 행사를 위해 음악 검색 알고리즘의 중요성이 대두되고 있다. 특히, 영상에 쓰이는 음악이 원곡 그대로일 경우에는 검색이 용이하지만, 원곡을 변형한 경우에는 검색 성능이 떨어지기 때문에 더욱 주의가 필요하다.

원곡을 변형해 사용하는 대표적인 사례로는 커버곡이 있다. 커버곡은 이미 있는 곡을 자신의 스타일대로 바꿔서 부르는 것을 뜻하며, 이 과정에서 음의 높낮이나 리듬 및 빠르기, 곡 전체의 구성 등이 변하게 된다[1]. 이러한 변형에 대비하기 위해 기존의 알고리즘들은 비트 동기 크로마 기법 등의 방법을 사용했지만, 성능이 좋지 않을 뿐 아니라 음악 전체의 정보를 가지고 있어야 하기 때문에 메모리 소모도 크다.

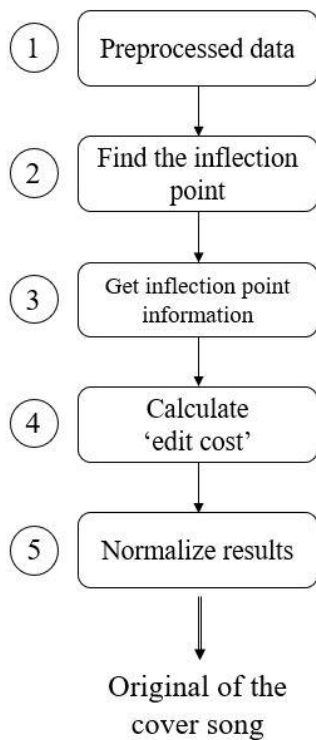
이에 본 논문에서는 멜로디 라인의 변곡점을 활용한 검색 알고리즘을 제안한다. 이는 음의 높낮이나 빠르기 변화에도 성능을 유지할 뿐 아니라, 곡 전체가 아닌 변곡점만을 활용하기 때문에 메모리 사용을 줄일 수 있다.

2. 관련연구

음악 검색 알고리즘은 음의 높이 변화나 길이 분석을 중심으로 연구가 이뤄지고 있다. 대표적으로 [1]은 옥타브 차이가 나는 음악의 성분들을 계산하여 전체 주파수를 하나의 옥타브 안에 넣어 표현한 크로마그램을 이용한다. 이러한 방법으로는 멜로디 정보는 잘 알 수 있지만 옥타브 정보가 사라진다는 단점이 있다. [2]는 음의 높이 변화 방향을 U(up), D(down), S(same)라는 스트링으로 변환하여 사용하였고, [3]은 음의 높이 변화 정도를 정보에 포함하여 정확도가 향상된 검색 알고리즘을 제안하였다. 그러나, 이러한 알고리즘들은 곡의 빠르기 변화에는 취약하다는 단점이 있다. [4]는 빠르기가 변형된 곡을 검색할 수 있는 비트 동기 크로마 기법을 제안하였으나, 이는 곡 전반의 빠르기가 일정할 때에만 유효하다는 한계점이 있다.

3. 제안 알고리즘

본 논문에서 제안하는 알고리즘은 멜로디 라인의 변곡점을 찾아내고 변곡점의 특징값으로 음악의 유사도를 계산한다. 알고리즘은 크게 전처리, 변곡점 정보 추출, 커버곡의 원곡 검색으로 구성되며, 전체적인 흐름은 (그림 1)과 같다.



(그림 1) 알고리즘 플로우차트

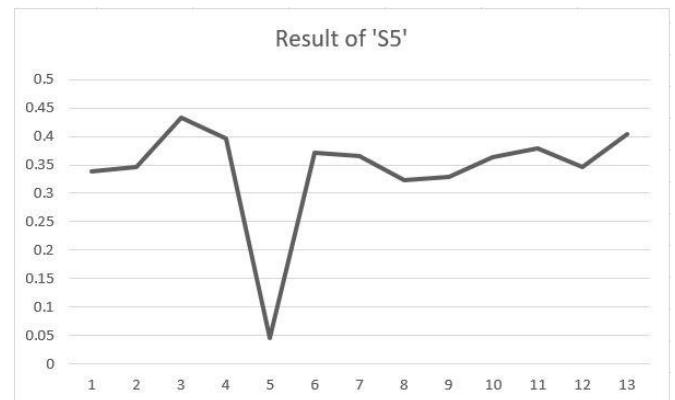
알고리즘은 5 단계로 구성된다. 단계 1 인 전처리 단계에서는 원곡의 후렴구와 커버곡 전체에서 멜로디를 추출한다. 단계 2 에서 전처리된 데이터에 대하여 멜로디 변화 양상이 바뀌는 변곡점 부분을 찾아낸다. 이 때 변곡점은 멜로디와 멜로디의 기울기가 모두 바뀌는 부분을 뜻한다. 즉, 일정한 기울기로 계속 증가하는 부분은 변곡점에 포함되지 않는다. 변곡점을 구한 후에 단계 3 에서는 변곡점에 대하여 전후 기울기 변화 양상(이하 타입)과 앞 뒤 변곡점과의 거리 비율(이하 세부 비율)을 이용해 변곡점 정보를 추출한다.

단계 4~5 에서 커버곡의 변곡점 정보를 모든 원곡의 변곡점 정보와 각각 비교하여 두 변곡점 시퀀스 간의 유사도를 계산하여 가장 유사도가 높은 곡을 찾아낸다. 이 때 유사도가 높다는 것은 ‘edit cost’ 값이 작다는 것을 의미하며, ‘edit cost’ 는 edit distance 를 변형한 방법으로, 두 변곡점의 타입이 같은 경우에는 세부 비율까지 계산에 넣음으로써 변곡점의 각 특징에 가중치를 다르게 둔다. ‘edit cost’ 는 두 변곡점의 타입이 다른 경우에는 edit distance 의 수식을 그대로 쓰고, 같은 경우에는 앞서 계산한 세부 비율을 수식에 더해 사용한다. 그리고 원곡 후렴구의 길이에 따라 ‘edit cost’ 값이 달라지기 때문에 후렴구의 길이로 나눠 값을 정규화시킨다. 가장 낮은 ‘edit cost’ 값을 가진 곡이 커버곡의 원곡이라고 할 수 있다.

4. 구현 및 성능평가

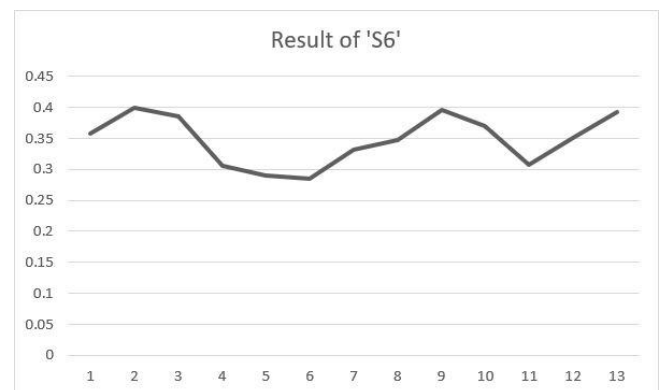
본 연구에서는 멜로디 추출 방법으로 악보를 사용하여, 한 마디에 16 개의 미디 값을 추출하는 방법을 사용하였으며, 연구의 원활한 진행을 위해 커버곡은 1 절만 사용하였다. 테스트 결과 본 알고리즘은 ‘edit cost’ 를 정규화 하지 않고 그대로 사용했을 때는 80%, 정규화 했을 때는 90%의 정확도를 보였다.

(그림 2)는 커버곡 ‘S5’ 와 원곡 후렴구 13 개의 edit cost 를 계산하고 정규화 한 결과 그래프이다. 커버곡 ‘S5’ 의 원곡은 5 번곡이었는데, 5 번의 결과값만 확연히 낮음을 확인할 수 있다.



(그림 2) 커버곡 ‘S5’ 의 결과 그래프

(그림 3)은 커버곡 ‘S6’ 의 결과 그래프이다. 커버곡 ‘S6’ 의 원곡은 6 번곡이었는데, 6 번 곡의 ‘edit cost’ 값이 가장 작긴 하지만 다른 곡들과 큰 차이를 보이지는 않는다. 이는 커버곡 ‘S6’ 가 원곡에 비해 바이브레이션 등의 변주가 많았기 때문으로 보인다.



(그림 3) 커버곡 ‘S6’ 의 결과 그래프

5. 결론

본 논문에서는 곡의 멜로디 라인에서 변곡점을 추출하고, 변곡점 정보를 활용해 음악의 유사도를 계산하여 커버곡의 원곡을 검색하는 알고리즘을 제안하고 성능 평가를 하였다. 이는 음악의 높낮이 변화뿐 아니라 빠르기 변화에도 검색 성능을 유지하였고, 곡 전체의 정보가 아니라 변곡점 정보만을 사용하기 때문에 메모리 사용도 적다는 장점이 있다. 그러나, 바이트레이션과 같이 한 음에 작은 차이로 여러 번 변주를 두는 경우에는 검색 성능이 다소 감소하는 경향을 보이는데, 이를 개선하는 연구를 진행 중이다. 이와 같은 점을 보완하면 커버곡 탐색을 위한 성능 높은 알고리즘이 될 것으로 기대한다.

참고문헌

- [1] 서진수, “강인한 크로마그램 성분 추출을 통한 커버곡 검색 성능 개선,” 멀티미디어학회논문지, 제 22 권, 6 호, pp. 639-645, 2019.
- [2] A. Ghias, J. Logan, D. Chamberlin, B. C. Smith, “Qeury By Humming – Musical Information Retrieval in An Audio Database,” *MULTIMEDIA '95: Proceedings of the third ACM international conference on Multimedia*, San Francisco California USA, 1995, pp. 231-236.
- [3] J. Jee, H. Oh, “Design and Implementation of Music Information Retrieval System,” *The Transactions of the Korea Information Processing Society*, Vol.5, No.1, pp.1-11, 1998.
- [4] D. P. W. Ellis, G. E. Poliner, “Identifying ‘Cover Songs’ with Chroma Features and Dynamic Programming Beat Tracking,” *2007 IEEE International Conference on Acoustics, Speech and Signal Processing – ICASSP '07*, Honolulu, HI, 2007, pp. IV1429-IV1432.