

Query By Humming 응용을 위한 MIDI 파일에서의 자동

멜로디 트랙 선택 방법

김무정⁰, 낭종호
서강대학교 컴퓨터공학과
ilovesky0517@naver.com, jhnang@sogang.ac.kr

An Automative Melody Track Selection in MIDI Files for Query By Humming(QBH) Application

Moojung Kim⁰, Jongho Nang
Dept. of Computer Science and Engineering, Sogang University

요 약

기존의 작곡가, 곡명 등의 질의어로 대표될 수 있는 메타 데이터 기반 음악 검색은 음악 데이터의 양이 급증함에 따라 탐색 근거가 되는 사항을 모를 경우 사용자의 요구를 충족시켜 줄 수 없는 단점을 가지고 있다 [1]. 반면 음악의 내용으로 한 검색의 경우 이러한 제약에서 보다 자유로울 수 있다. 내용 기반 검색[2]의 연구에 있어서 음악 데이터로부터 사용자의 질의를 처리해주기 위한 멜로디를 추출해주는 것이 중요한 문제가 되는데, 데이터의 추출과 편집이 용이한 MIDI 파일이 많이 사용되고 있다. 본 논문에서는 사용자의 질의를 바탕으로 한 QBH system상에서의 음악 데이터 구축을 위해 MIDI파일에서 추출해 낼 수 있는 특징들을 이용해 MIDI파일의 멜로디 트랙 멜로디 트랙과 반주 트랙을 자동으로 구분하는 것을 목적으로, MIDI트랙에서 추출해서 이용할 수 있는 특징들에 대해 알아보고 그에 따른 간단한 분류 알고리즘의 제안과 실험 결과에 대해 소개한다.

1. 서 론

MIDI파일은 Query By Humming(QBH) 등의 음악 검색에서 데이터를 처리하기에 적합한 파일 형식으로 여겨지고 있다. 하지만 일반적인 MIDI파일은 하나 이상의 트랙으로 이루어져 있기 때문에 많게는 수만 곡 혹은 수십만 곡의 미디 파일에서 특징을 추출하기 위해 멜로디 트랙을 선택해주는 것을 직접 하기에는 많은 무리가 따르는 것이 사실이다. 따라서 이번 연구를 통해 MIDI내의 변별력 있는 특징을 통해 멜로디 트랙 선택을 최대한 자동화함으로써 QBH의 데이터베이스 구축의 효율성을 꾀하고자 한다. 또한 MIDI 파일은 format0[3]의 경우 단일 트랙으로 구성되어 있는데, 이러한 경우에는 하나의 연주를 담당하는 단위인 채널에 대해, 최대 16개의 채널에서 채널별로 정보를 분리해서 각 채널의 연주 정보를 트랙으로 여기고 분석을 하도록 하겠다.

2. 관련 연구

멜로디 선택과 관련하여 초기의 가장 주목할 만한 연구는 Michael TANG et al.[4]이 2000년에 발표한 논문을 들 수 있는데, 여기서 Michael TANG et al.는 음악 데이터 베

이스 구축을 위해 MIDI파일에서 멜로디 라인을 선택하는데 사용될 수 있는 특징들에 대해 제안하였다. 또한 2006년에 발표된 David Rizo et al.[5]의 논문에서는 MIDI 내에서 멜로디 선택을 위해 사용될 수 있는 특징들에 대해 정리하였다. 또한 2009년에 JIANGTAO LI et al.[6]는 MIDI 파일 내의 특징들을 신경망 네트워크로 학습시켜 멜로디를 선택하였다.

3. MIDI track의 유용한 특징들

MIDI은 내용적으로 보자면 크게 멜로디 부분과 반주 부분으로 구분할 수 있다. 사용자의 질의를 처리하기 위한 특징을 추출하기 위해서는 노래의 멜로디와 반주를 구분할 필요가 있다. MIDI파일에서 추출할 수 있는 특징[5]들은 음의 높이, 음의 길이, 음간의 간격 등을 바탕으로 할 수 있다.

이번 chapter에서는 MIDI 파일의 트랙들로부터 추출할 수 있는 유용한 특징들에 대해서 소개하도록 하겠다.

3.1 화음의 비율

화음이란 동시에 여러 개의 음을 소리 낼 수 있는 악기

또는 그 상태를 가리키는데, 일반적으로 반주 트랙에서 코드를 포함하는 경우에 화음으로 연주가 되곤 한다 [7]. 멜로디 트랙이 대부분 단 선율로 이루어졌다고 가정했을 때 화음의 비율(Polyphonic Ratio)은 트랙 내의 전체 음표의 수에 대한 화음으로 연주되는 음표의 비율로써, MIDI 파일과 같이 정의할 수 있다.

$$PMRatio_j = \frac{1}{N_j} \sum_{i=1}^{N_j} P_{ji} \quad (1)$$

N_j 는 j 번째 트랙 내부의 전체 음표의 개수이고 P_{ji} 는 j 번째 트랙의 i 번째 음표가 화음에 속해 있으면 1의 값을 갖고 단 선율이면 0의 값을 갖는다. 화음은 ① 이전 음표가 시작되고 MIDI 파일 내의 delta time이 0인 상황에서 또 다른 note가 시작되는 경우, ② 이전 음표가 끝나지 않은 상황에서 새로운 음표가 시작되는 경우로 구분할 수 있다. 여기에 화음 임계치(polyphonic threshold)를 통해 화음의 비율이 임계치 이상인 트랙은 반주 트랙으로, 그렇지 않은 트랙은 멜로디 트랙으로 분류한다.

3.2 음 높이 변화의 평균

멜로디 트랙은 반주 트랙과 달라 음의 전체적인 흐름을 포함하게 된다. 따라서 반주 트랙에 비해 음의 갑작스런 변화가 작을 것이다 [6]. 따라서 연속된 음표의 높이 차를 추출해서 비교하면 멜로디 트랙과 반주 트랙간의 구별에 유용한 특징으로 삼을 수 있다.

MIDI파일 내의 j 번째 트랙의 음표 높이 차이의 평균 $note Var_j$ 는 다음과 같이 정의할 수 있다.

$$note Var_j = \frac{1}{N_j - 1} \sum_{i=1}^{N_j - 1} |d_j(i+1, i)| \quad (2)$$

$d_{j(i+1, i)}$ 은 j 번째 트랙의 i 번째 음표와 $i+1$ 번째 음표간의 음 높이의 차이를 의미한다.

3.3 등장 음표의 범위

우리는 또한 트랙 내부에서 등장하는 음표의 범위(range)를 이용해 트랙들을 분류할 수 있다. 멜로디 트랙은 사람에 의해 노래되는 음정으로 만들어지는 경우가 많기 때문에 [4], 음표들의 범위를 통해 멜로디 트랙이 될 수 있는 것들과 그렇지 않은 것들을 분류할 수 있다. j 번째 트랙의 음표들의 범위 $Range_j$ 는 다음과 같이 정의할 수 있다.

$$Range_j = \max(Pitch_j(i)) - \min(Pitch_j(i)), 1 \leq i \leq N_j \quad (3)$$

식 (3)에서 $Pitch_j(i)$ 는 j 번째 트랙의 i 번째 음표의 음 높이를 의미한다.

3.4 음 높이의 평균

일반적으로 멜로디 트랙에서의 음표의 분포는 멜로디가 사람의 음성을 반영하는 특징 때문에 반주 트랙에서의 그것보다 더 제한적이다 [6]. 따라서 이점을 이용하면 반주 트랙과 멜로디 트랙 구분의 성능 향상을 기대할 수 있다.

MIDI파일의 j 번째 트랙에서의 음 높이의 평균 $avgPitch_j$ 는 다음과 같이 정의할 수 있다.

$$avgPitch_j = \frac{1}{N_j} \sum_{i=1}^{N_j} Pitch_j(i) \quad (4)$$

3.5 기타 트랙 내부 정보

트랙의 기타 내부 정보(Track description)란 트랙 내에서 Note on과 Note off를 제외한 Channel event나 기타 meta event를 의미한다. 그 중 가장 의미 있을 만한 정보로는 트랙 내부에서 타악기가 사용되었는지 여부와, 트랙의 이름을 정보를 담을 수 있는 meta event 내에 멜로디 트랙임을 지칭해줄 수 있는 정보가 포함되어 있는지 여부이다. 우선, 특정 트랙이 타악기 연주로 이루어졌을 때, 해당 트랙은 멜로디 트랙일 확률이 지극히 낮다고 할 수 있다. 이는 타악기가 일정한 음정을 가진 악기와 음정을 갖지 않는 악기로 분류될 수 있기 때문인데 [8], 음정을 갖지 않는 타악기 같은 경우(ex. 팀파니, 실로폰, 마림바 등)에는 멜로디를 연주하기에 적합하지 않은 악기이다. 실제로 MIDI파일에서는 채널 10을 타악기에 할당하고 음의 높이로 악기를 선택한다. 미디 파일 내에서 j 번째 트랙의 타악기 사용 여부 $percUse_j$ 는 다음과 같이 정의할 수 있다.

$$percUse_j = \begin{cases} 1: j\text{-th 트랙에 타악기가 사용됐을 때} \\ 0: j\text{-th 트랙에 타악기가 사용되지 않았을 때} \end{cases} \quad (5)$$

채널 내부 프로그램의 변경을 통해 타악기가 연주에 사용되는지 여부를 판단해 이진형식으로 나타낸다.

다음으로 트랙의 이름을 설정했을 경우인데, 종종 트랙의 이름으로 'melody' 등의 키워드를 삽입함으로써 멜로디 트랙임을 지칭해 주는 경우가 존재한다. 트랙의 이름에서 특정 키워드를 발견하면 $description_j$ 를 1로 설정해 줌으로써 키워드의 존재유무를 판단할 수 있다.

$$description_j = \begin{cases} 1: j\text{-th 트랙에 특정 키워드 발견} \\ 0: j\text{-th 트랙에 키워드 없음} \end{cases} \quad (6)$$

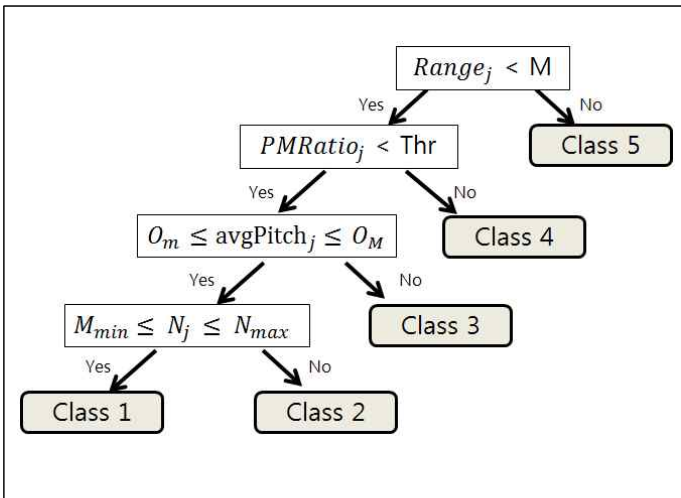
3.6 기타 특징들

기존의 멜로디 선택 관련 선행 연구의 경우에는 음표의

세기의 평균, 무음구간과 연주구간의 비율, 트랙 내 음표의 개수 등을 특징으로 사용하고자 하는 시도가 있었다 [4, 5, 6]. 여러 가지 특징들 중에 음표 세기의 평균[4] 같은 경우에는 멜로디 트랙의 음의 세기가 기타 반주 트랙보다 클 것이라는 가정에서 도입한 특징인데, 실험적으로 관찰해본 결과 그 변별력이 크지 않음을 알 수 있었다. 또한 음표의 길이와 관련해서 길이 변화의 평균이나 길이를 기준으로 한 음표의 구성 비율을 특징으로 삼는 것도 유용한 연구가 될 수 있을 것이라고 생각한다.

4. 멜로디 트랙 선택 알고리즘

그러면 이제 앞서 제시한 특징들을 토대로 멜로디 트랙과 반주 트랙을 분류해야 하는데, 여러 특징들을 독립적으로 적용해보고 또한 보다 변별력을 높이기 위해 복수개의 특징들을 동시에 적용해 멜로디 트랙과 반주 트랙을 분류하면 단일 기준을 가지고 비교했을 때에 비해 성능 향상을 기대할 수 있다. 본 논문에서는 복수 개의 이진 분류기를 사용하여 멜로디 트랙을 선택했는데 사용한 알고리즘은 다음과 같다.



<그림 1> 이진 분류를 통한 MIDI 트랙 분류

<그림 1>은 파일 내의 모든 트랙을 Class 1부터 Class 5로 분류하고, Class 번호가 낮은 트랙부터 트랙 순서에 오름차순으로 최종 우선순위를 부여한다. $Range_j$ 의 경우 최대 40의 범위를 넘지 않는 것으로 멜로디 트랙을 한정했는데, 이는 약 3옥타브의 범위를 표현하는 것으로 멜로디 트랙이 사람의 발성과 연관이 되는 경우가 많기 때문에 정상적인 남성의 발성 영역을 반영한 수치이다. (여자는 약 2.5 옥타브로 남성의 그것보다 낮았다.)

$PMRatio_j$ 는 1에 가까울수록 화음이 빈번하게 등장하는 것이고 0에 가까울수록 단음 연주로 가까운 트랙임을 가리

킨다. 따라서 임계값으로 0에 가까운 값(본 논문에서는 0.1로 설정)으로 설정해 기준으로 삼았다.

평균 음 높이에 대해서는 130Hz ~ 1000Hz (4옥타브 ~ 6옥타브)로 범위를 설정하였는데, 사람이 발생 가능한 주파수 대역은 15Hz ~ 8,500Hz이지만 멜로디 트랙에서의 음의 분포를 통해 살펴보았을 때 주로 130Hz ~ 1000Hz (4옥타브 ~ 6옥타브)의 범위 내에서 주된 분포를 보였고, 음표들의 평균을 구할 경우에는 약 300Hz ~ 900Hz (5옥타브 ~ 6옥타브)의 범위의 값을 갖는 관찰 결과를 보였다. 반면 반주 트랙은 사람의 음성을 반영해야 한다는 제약에서 자유롭기 때문에 보다 더 다양한 음의 분포를 보였고, 평균을 구해 봤을 때 100Hz(4옥타브의 라)보다 작은 값을 갖는 트랙도 존재하였다. 또한 전체 음표의 개수가 너무 적거나 혹은 너무 많지 않을 경우를 조건으로 설정하였다. 이는 짧은 음을 반복해서 연주하는 반주 트랙을 제거하기 위한 것으로서, 또한 화음의 등장이 빈번한 트랙의 경우에도 상대적으로 많은 음표를 가질 수 있으므로 유용할 수 있다. 본 실험에서는 파일 내부의 모든 트랙의 음표 개수의 평균을 구한 뒤, 평균값의 $\frac{1}{2}$ 이상이며 2배 이하인 트랙들에게 최종적으로 높은 우선순위를 부여하였다.

모든 조건을 만족하는 트랙이 하나 이상일 가능성이 있기 때문에 정확성의 향상을 위해 한 개의 멜로디 트랙 선택 대신 priority에 의한 멜로디 트랙으로서의 가능성이 높은 상위 5개의 후보군(candidate)을 제시해 주는 방식을 취하였다.

5. 실험 결과

실험에 사용된 음악 데이터는 인터넷에서 다운로드 받을 수 있는 다양한 장르의 MIDI파일로서, 이 연구가 시작되게 된 배경이 한국 대중음악을 기반으로 한 QBH의 개발에 있었기에 한국 대중음악을 우선시 하여 한국 음악 60곡, 해외 음악 60곡에 대하여 실험을 수행하였다.

[표 1]은 멜로디 트랙의 평균 음높이와 음의 범위에 대한

표 1. 멜로디 트랙의 음표 정보

	avgPitch	Range
평균	70.97725	17.5
MIN	62.37975	8
MAX	90.9045	31
표준편차	4.338929	4.36043

정보를 나타낸다. 각 값은 MIDI 파일 내에서의 음 표현 방식인 0~127까지의 값을 기준으로 표현한 것이다.

반면, [표 2]는 반주 트랙의 평균 음높이와 음의 범위에

표 2. 반주 트랙의 음표 정보

	avgPitch	Range
평균	56.94937	16.76
MIN	3.364341	0
MAX	88	50
표준편차	16.01871	11.44051

대한 정보를 나타낸다. [표 1]과 [표 2]를 비교해보면, 멜로디 트랙의 경우 평균 음 높이와 음의 범위가 평균의 범위에서 편차가 크지 않은 값을 나타내지만, 반주 트랙의 경우에는 멜로디 트랙에 비해 편차가 큰 것을 볼 수 있다. 또한 반주 트랙의 경우 평균은 약 56에 해당하지만 avgPitch에 대한 하한 임계치로서 설정했던 4옥타브의 도(48)보다 낮은 값을 갖는 트랙이 전체 반주 트랙의 약 30%에 이르기 때문에 상당수의 반주 트랙을 후보군에서 제거할 수 있다.

[표 3]은 이진 분류를 통한 멜로디 트랙 선택의 성능을 나타낸다. 전체 81곡의 MIDI파일에 대해서 각 항목별로 rank1의 정확도부터 최종 rank5의 정확도를 보여준다. i번째 rank의 정확도는 다음과 같이 정의할 수 있다.

$$\text{정확도} = \frac{i\text{-th rank가 멜로디 트랙을 찾은 개수}}{\text{전체 MIDI파일 수}}$$

[표 3.] 멜로디 트랙 선택의 성능

Rank	92곡 기준	rank 별 정확도	누적 정확도
1	75	81.52%	82.72%
2	8	8.70%	91.42%
3	2	2.17%	93.59%
4	1	1.09%	94.68%
5	2	2.17%	96.85%

6. 결론

앞서 기술한 내용을 통해 MIDI 파일 내에서 멜로디 트랙과 반주 트랙을 분류할 수 있을만한 특징들에 대해 기술하였다. MIDI파일의 정보를 통해 얻어낼 수 있는 각각의 특징들을 분석하여 멜로디 트랙과 반주 트랙과의 차이점에 대해 조명해 보고, 여러 특징들의 결합을 통해 멜로디 트랙을 선택하는 방안을 보였다. 음악 검색에 있어서 멜로디 선택은 음악 데이터를 처리하기 위한 중요한 전 처리 과정으로써 상당한 수준의 정확도를 요구한다. 이러한 멜로디 트랙 선택이 높은 정확도를 가지고 자동화 될 수 있다면 음악 데이터베이스 구축에 상당한 효율성을 기할 수 있게 된다. 현재는 여러 특징들에 대해 각각의 특징들에 대한 이진 분류 형식으로 트랙을 선택했지만, 머신 러닝의 학습 과정을 통하여 보다 체계적인 성능 향상이 기대 된다. 본 논문에서

는 멜로디 트랙 선택이 반자동화로 진행되었는데, Rank1에 의한 정확도를 높이면 멜로디 트랙 선택의 자동화가 가능할 것으로 기대된다.

References

[1] 이윤주, 문성빈, “내용기반 음악정보검색시스템의 질의 인터페이스 설계를 위한 이용자 연구,” *지식처리연구* 5권 1/2호, 2004.

[2] Seungmin Rho, Eenjun Hwang and Jong Hyuk Park, “M-MUSIC: An Intelligent Mobile Music Retrieval System,” *Multimedia Systems, Online FirstTM*, 2010.

[3] MIDI manufacturer association, 'www.midi.org'.

[4] Michael Tang, YIP Chi Lap and Ben KAO, “Selection of Melody Lines for Music Databases,” *Proceeding of the Computer Software and Applications Conference 2000*.

[5] David Rizo and Antonio Pertusa et al. “A Pattern Recognition Approach for Melody Track Selection in MIDI Files,” *Proceeding of International Society for Music Information Retrieval 2006*.

[6] Jiangtao Li, Xiaohong Yang and Qingcai Chen, “MIDI Melody Eextracion Based On Improved Neural Network,” *Proceeding of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, pp 12-15, 2009*.

[7] Justin Zobel et al. “Manipulation of Music For Melody Matching,” *Proceedings of the sixth ACM international conference on Multimedia 1998*.

[8] T.D. Rossing, *Science of percussion instruments*. World Scientific, pp 1-4, 2001.