

대표구간의 음악 특징에 기반한 음악 장르 분류 (Music Genre Classification based on Musical Features of Representative Segments)

이 종 인 [†] 김 병 만 ^{††}
(JongIn Lee) (Byeong Man Kim)

요약 일부 음악 장르 분류에 관한 기존 연구에서는 특징 추출을 위한 구간 선택 시 사람이 직접 곡의 주요 구간을 지정하는 방법을 사용하였다. 이러한 방법은 분류 성능이 좋은 반면 수작업으로 인한 부담으로 새롭게 등록되는 음악들에 대해 지속적으로 적용하기가 곤란하다. 수작업 없이 음악 특징을 추출하기 위해 최근 음악 장르 분류와 관련된 연구에서는 자동으로 추출구간을 선정하는 방법을 사용하고 있지만 이러한 연구의 대부분이 고정된 구간 (예, 30초 이후의 30초 구간)에서 특징을 추출하는 관계로 분류의 정확도가 떨어지는 문제점을 갖고 있다. 본 논문에서는 이러한 문제점을 해결하기 위해 곡 전체 구간에 대하여 반복구간들을 파악하고 이들의 위치와 에너지를 고려하여 곡을 대표할 수 있는 단일 대표구간을 선정한 후, 대표구간으로 부터 특징을 추출하여 장르 분류시스템에 적용하는 방법을 제안하였다. 실험 결과, 기존 고정구간을 사용한 방법에 비해 괄목할 만한 성능 향상을 얻을 수 있었다.

키워드 : 음악 장르 분류 시스템, 음악 대표 구간 선정, 음악 구조 분석, 음악 세그먼트 분류, 내용 기반 음악 특징 추출

Abstract In some previous works on musical genre classification, human experts specify segments of a song for extracting musical features. Although this approach might contribute to performance enhancement, it requires manual intervention and thus can not be easily applied to new incoming songs. To extract musical features without the manual intervention, most of recent researches on music genre classification extract features from a pre-determined part of a song (for example, 30 seconds after initial 30 seconds), which may cause loss of accuracy. In this paper, in order to alleviate the accuracy problem, we propose a new method, which extracts features from representative segments (or main theme part) identified by structure analysis of music piece. The proposed method detects segments with repeated melody in a song and selects representative ones among them by considering their positions and energies. Experimental results show that the proposed method significantly improve the accuracy compared to the approach using a pre-determined part.

Key words : Music Genre Classification System, Music Representative Part Detection, Musical Structure Analysis, Music Segment Detection, Content-based Music Feature Extraction

* 본 연구는 금오공과대학교 학술연구비에 의하여 연구되었음.

[†] 학석회원 : 금오공과대학교 소프트웨어공학전공
inisphier@kumoh.ac.kr

^{††} 종신회원 : 금오공과대학교 컴퓨터공학부 교수
bmkim@kumoh.ac.kr

논문접수 : 2008년 5월 20일

심사완료 : 2008년 10월 14일

Copyright©2008 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지 : 소프트웨어 및 응용 제35권 제11호(2008.11)

1. 서론

최근 멀티미디어 데이터베이스에 저장된 음악 정보가 급속히 증가하고 있으며, 인터넷을 이용한 디지털 음악을 수집하고 소장하는 사용자들이 증가하고 있다. 이에 따라 사용자들은 방대한 양의 디지털 음악을 효과적으로 분류 및 검색할 수 있는 시스템을 요구하고 있다. 그러나 지금까지 디지털 음악 데이터들은 수작업을 통해 제목, 가수 등의 텍스트 기반의 인덱스가 입력되고 관리되어져 오고 있다. 이러한 텍스트 기반의 검색 시스템은 사용자가 원하는 데이터를 찾기도 수월하지 않고, 수

작업에 따른 지속적 관리에 부담이 크다. 이에 최근에는 음악이 가지고 있는 내용 기반의 분류 및 검색하는 기술이 점차 중요시되고 있다.

음악 분류 중 일반적인 분류라 할 수 있는 음악 장르 분류에 관한 연구는 1990년대 후반, 2000년대 초반에 걸쳐 많은 연구들이 이루어졌다. 하지만 일반적으로 우리가 접하고 있는 음악 장르의 종류는 문화적, 음반 산업의 요구, 각각의 음악을 사용하는 그룹들과 관계하여 다양한 분류법들이 존재하고 있고, 이러한 장르 분류법을 내용 기반의 분류 및 검색에 적용시키기에는 무리가 있다[1]. 이에 대한 대안으로 내용기반 장르 판별, 즉 음악 오디오로부터 특징들을 추출하고 이를 기반으로 분류하는 연구들이 많이 이루어져 왔고, 그 결과, 음악 특징 추출에 대한 연구가 일정 수준 이상에 도달하게 되었다.

음악은 전체적인 흐름에 있어 그 변화량이 다양하여 지금까지 나온 특징들을 사용하여 전역적인 특징을 추출하기 위해서는 연산량이 매우 많아지거나, 전역적인 특징을 담기 위한 저장소의 공간적 문제가 발생할 수 있고, 또한 음악 내의 다양한 변화 부분 때문에 다른 장르의 특성 또한 포함되어 이를 분리/판별하는 것이 어렵다. 기본적으로 지금까지의 음악 장르 분류에 관한 연구들은 곡 내 일정 구간을 선택하여 그 구간 내에서 특징을 추출, 판정하는 절차를 거치게 되는데 그 구간을 결정함에 있어 전문가가 수작업으로 선택하거나 임의의 고정 구간 (30초 이후의 30초 구간) 을 선택하고 있다.

반약 곡내부의 구조 분석을 통하여 곡을 대표 할 수 있는 구간, 즉 대표 구간을 추출하여 이를 이용한 내용기반 장르 판별을 하게 된다면 전문가의 수작업 같은 많은 시간이 소모되는 작업이 생략되거나, 임의의 구간을 선택함에 따라 발생하는 오류에 대하여 보완 하는 이점이 있으리라 본다. 이러한 이유에서 본 논문에서는 곡의 반복구간을 검출 후, 이 중 곡 전체를 대표할 구간을 선정하고 이 대표 구간에서 특징을 추출하여 이를 자동 분류에 적용하는 실험을 하였다.

2장에서는 음악 장르 분류 및 판정에 관한 연구들과 음악 구조 분석 연구와 관련하여 썸네일(Thumbnail), 즉 곡의 주요 구간 판정에 관한 연구들에 대해 살펴보고, 3장에서는 곡 내에서 주요 멜로디가 반복되는 특징을 이용한 대표구간 선택 방법과 선택된 대표구간을 바탕으로 한 내용기반 음악 장르 판정 방법에 대해 살펴본다. 그리고 4장에서는 본 연구에서 제안한 방법과 기존 방법과의 성능 비교를, 5장에서는 결론과 본 연구의 차후 확장 방향에 대해서 살펴본다.

2. 관련연구

음악 장르를 분류하기 위해선 일반적인 기계학습 분야의 프로세스 모델에 따라 전처리, 특징 추출, 학습, 판정의 단계를 거쳐 이루어진다. 지금까지 이루어지고 있는 음악 장르 분류에 관한 연구들은 이중 특히 장르 분류에 적합한 특징 추출과 각 특징들의 장단점을 비교하는 데 초점이 맞추어져 왔다. 그리고 이러한 연구들은 수작업을 통해 곡의 일부분을 추출하거나 곡들의 길이를 고정시킨 음악 데이터베이스를 사용하고 있다.

음악 특성상 곡 전체를 모두 처리하여 특징을 추출하기에는 저장소의 용량, 방대한 연산량에 따른 시간의 소요, 음악 내부적인 변화성 등의 문제로 어려움이 있다. 그래서 지금까지는 대부분 전문가가 수작업을 통하여 곡의 일부분을 추출하여 사용하거나 곡의 고정된 일정부분을 사용하고 있다. 전문가의 수작업에 의존하는 방식은 인식 성능 면에서는 이득이 있지만 새로운 음악마다 전문가의 손을 필요로 한다는 단점을 갖고 있다. 고정된 일정부분을 사용하는 방법은 수작업을 요하지 않는다는 장점이 있는 반면 곡에 따라 성능의 편차가 심해질 우려가 있다. 지금까지의 연구들처럼 사람의 수작업을 통하여 곡의 대표 부분을 추출을 하거나 고정된 일정부분을 사용하는 것보다 더 유용한 방법은 곡 내용을 기반으로 곡 전체 중에서 곡을 대표할 수 있는 구간을 파악하여 이 부분의 특징을 추출하여 사용하는 것일 것이다. 본 연구도 이러한 맥락에서 이루어졌다.

다음 절에서는 내용 기반 음악 장르 판정에 대한 기존 연구들을 살펴보고 본 연구의 토대가 되는 음악 구조 분석을 통한 썸네일 추출 방법들에 대해서 살펴본다.

2.1 내용 기반 음악 장르 판정 연구들

일반적으로 우리 일상에서 접하고 있는 디지털 음악의 경우 아날로그 음성 신호를 직접적으로 디지털로 변환하여 사용하고 있다. 간혹 신호 보다 상위 정보인, 악기, 음고(Pitch), 박자(Beat)등의 심볼릭(Symbolic) 데이터를 기반으로 한 MIDI와 같은 형태로도 사용되나, 새로운 음악이 나올 때 마다 이를 심볼릭 데이터 형태로 제공하는 것은 극히 드물다.

일반적인 음악 신호 정보, 즉 오디오 샘플을 사용한 내용기반 장르 판정에 관한 연구들은 대부분 장르 판정에 적합한 특징들을 추출하는데 초점이 맞추어져 왔다. 그 중 [2]는 다양한 음악 장르 판정 연구의 기반이나 비교 대상이 되는 연구로, 음악 장르 판정을 위하여 기존에 사용하는 음악 특징과 음악에 관련된 특징들에 대해 비교하였다. 10개의 장르를 대상으로 하였으며 다양한 특징점의 조합에 대하여 다양한 학습 알고리즘을 사용하여 비교 하였다. 이 연구에서 사용한 대표적인 특징점으로는 기존 음성인식 및 음향판별 분야에서 널리 쓰인 특징들인 음색적 특징(Timbral Features)인 Centroid,

Rolloff, Flux, ZCR(Zero Crossing Rate), MFCC(Mel-Frequency Cepstrum Coefficient)와 음악을 표현하는 특징으로 비트(Beat)와 하모니(Harmony)를 표현하기 위한 특징인 비트 히스토그램(Beat Histogram)과 피치 히스토그램(Pitch Histogram)을 사용하였다.

[3]에서는 음성인식 분야에서 시작된 멜 주파수 밴드(Mel-Frequency Band)기반의 MFCC보다 음악의 분류를 위해 옥타브 밴드(Octave Band) 기반의 특징인 Spectral Contrast라는 새로운 특징을 제안하여 바로크(Baroque), 로맨틱(Romantic), 팝(Pop), 재즈(Jazz) 그리고 락(Rock)의 5개의 장르에서 MFCC와 비교하여 더 좋은 성능을 나타내는 것을 보여주었다.

[4]에서는 DWCHs(Daubechies Wavelet Coefficient Histograms)라 불리는 웨이블릿 히스토그램에 기초한 새로운 음악 특징 추출 방법을 제안하였다. DWCHs는 기존 특징 추출에 많이 사용된 푸리에 변환과 달리 주파수 대역에 따른 다해상도(Multi-resolution) 특징을 나타내는 다우비치 웨이블릿 계수를 히스토그램 함으로써 국부적이고 전역적인 정보를 모두 표현하게 된다. [4]의 실험의 결과를 보면 [2]에 사용한 특징 중 음색적 특징과 함께 DWCHs를 함께 사용 하였을 시, 장르 판정에 있어 두 개의 장르분류에서는 최대 99%의 성능을 보였고 [2]에서 사용한 10개의 장르 분류에 적용하였을 때에도 78.5%의 높은 성능을 보여주었다.

음악 장르 분류를 위한 연구들의 경우, 각 연구별로 장르에 대한 표기법이 일관되지 못하고 변화량이 많은 음악의 특성에 의해 서로 다른 장르가 혼재하여 있는 경우를 고려하지 않았다. 음악 장르의 경우 문화, 역사, 음악 산업등의 다방면에서의 각기 다른 분류법이 산재한 문제점이 있고 일반적인 장르 분류법은 존재하지 않고 있다[1]. Li와 그의 동료는 [4]의 연구를 바탕으로 음악 장르 판별을 위하여 장르 분류법에 대하여 정리할 필요성을 제기하였고 [5]에서 기존 연구에서 사용한 장르 분류법을 계층형 분류법을 적용시켜 상위 레벨 1계층 장르를 분류하고 그 하위 세부 장르 분류에 대한 내용 기반 장르 판별을 위한 계층형 분류법에 대하여 실험 하여 10개의 장르에 관해서 0.3%, 6개 장르 분류에 관하여 3%의 성능 향상이 있음을 보여주었다. 다른 계층적 장르 판별 연구로 [6]에서는 가장 상위의 장르로 음악과 비음악으로 나누고, 음악의 장르에 대하여 계층화 한 후 계층별 판정에 좋은 특징들을 선별하는 방법을 사용하였다.

계층형 장르 판별의 다른 연구로 음악 장르 분류법의 확장을 용이 하도록 하는 연구[7]가 진행되고 있는데, 이는 새로운 장르가 추가 되었을 시, 기반이 되는 장르를 먼저 탐색 한 뒤, 하위 세부 장르를 탐색하는 방법에

대하여 연구하고 있다. 이 연구에서는 하위 장르 판별 및 실제 장르를 찾기 위하여 베이지안 집합(Bayesian Aggregation)을 통한 계층형 장르 트리의 탐색 및 판별 방법을 사용하고 있다.

2.2 음악 구조 분석을 통한 반복구간 탐지 및 추출

음악 내용 분석과 관련하여 내용기반 음악 분류 연구와 함께 음악의 구조 분석에 관한 연구 또한 진행되고 있다. 음악 구조 분석 연구에서는 가사부 탐색, 악기의 변화 구간 탐색과 같은 음색의 변화 구간탐색 방법[8,9], 음악 온셋(Onset)추출을 한 뒤 음표 리듬 패턴을 통한 구조 분석 방법[10], 특징 벡터간의 유사도 클러스터링을 통한 방법[11,12] 그리고 유사 멜로디의 반복적인 구조 탐색을 통한 방법[13]등을 사용하고 있다. 이러한 음악 구조 분석 연구를 바탕으로 한 응용분야로 이미지의 썸네일처럼 음악에서도 노래를 대표할 수 있는 부분을 추출하는 연구들[14,15]을 진행하고 있다.

음악의 구조 분석을 통한 썸네일 추출의 경우, 대개 멜로디(Melody)가 반복되는 구간의 추출이나 음악 내 특징점들의 유사도를 통한 방법을 사용하고 있다. [14]에서는 음악의 단일 썸네일이라 볼 수 있는 스니펫(Snippet)을 추출하기 위하여 크게 기본 특징 추출, 특이점 검출(Saliency Detection), 음악 구조 분석, 그리고 스니펫 생성이라는 네 가지 단계를 두어 스니펫을 추출하였다. 먼저, 기본 특징으로 Spectral Contrast와 MFCC를 단구간 푸리에 변환을 사용하여 각 프레임별로 추출하였다. 특이점 검출은 기본 특징들을 사용한 주요 멜로디의 반복 빈도 가중치, 구간별 에너지의 세기, 구간의 위치에 따른 가중치 이 세 가지의 가중치를 통하여 각 프레임의 특이점들을 파악하게 된다. 각 특이점은 가중치로 표현되는데 이를 특이점 가중치라 한다.

음악의 구조 분석에서는 곡의 소악절(Phrase)간 경계점을 탐색하게 되는데 이를 위해 템포를 추출한 후, 일반적으로 음악 이론상 한 개의 소악절은 네 개의 마디(Bar)로 이루어지고, 한 마디는 일반적으로 강한 두 비트(Beat)로 이루어진다는 가정 하에 8 비트씩 나누어 소악절간의 경계선을 추정한다. 그리고 소악절 경계선을 추정한 후 소악절 내의 연속 구간에서 가사 즉 사람의 음향이 연속되는 부분을 검출하여 소악절 경계점을 파악하게 된다. 이 때, 사람 음향이 연속되는 부분은 사람의 음향 주파수대역인 125~1000Hz의 에너지만을 추출하여 에너지의 최저점(Valley) 지점을 기반으로 파악하게 된다. 최종적으로, 특이점 검출 단계에서 파악된 특이점 중 가중치가 가장 높은 특이점을 선택하고 이를 포함하는 소악절을 식별함으로써 스니펫을 얻게 된다. 이렇게 얻어진 스니펫의 성능을 파악하기 위하여 개인적인 설문 평가로 좋음, 그러저러, 만족스럽지 못함을

나타내는 3, 2, 1의 세 가지 평가치로 조사하였는데 전체적으로 2.62 정도의 성능 평가를 얻었다.

[14]에서는 반복구간을 검출하기 위해 일반 음향처리 분야에서 사용되는 음향특징인 MFCC, Spectral Contrast를 사용하였으나, [15]에서는 음악 특징을 고려하여 리듬(Rhythm)과 하모니(Harmony)를 표현하기에 적합한 크로마그램(Chromagram)을 기반으로 한 썸네일 추출 방법을 제안하였다. 이 연구에서는 음악의 질을 파악하기 위해 프레임 별로 크로마(Chroma)를 추출하고 이를 통해 크로마그램을 만들어낸 후, 크로마그램간 유사도를 통하여 연속적이고 대칭적인 멜로디 구간 즉, 반복 구간 탐색하고 이 반복구간 중에 전주, 간주, 후렴부에 해당되는 부분은 제외하였다. 전주, 간주, 후렴부는 보통 사람의 음성 없이 악기 연주로만 구성되기 때문에 이를 탐지하기 위하여 ZCR과 에너지, 음고, 세 가지 정보를 통하여 사람의 음성이 들어간 부분을 탐지하였다. 최종적으로, 사람의 음성이 들어간 반복구간을 다중 썸네일로 추출하고 다시 사용자의 피드백을 기반으로 단일 썸네일을 추출하였다.

본 논문에서는 [15]의 방법과 거의 동일한 방법으로 대표구간을 선택하였다. 단, [15]와의 차이점은 [15]에서는 썸네일을 생성하기 위해 크로마그램을 통한 반복 구간 탐색 후 반복 구간간의 관계를 통한 구조분석을 사용하였으나, 본 연구에선 이를 제외한 채 단순한 반복구간만을 탐색하였고, [15]에서 다중 썸네일에서 단일 썸네일을 추출하기 위해 사용자 피드백에 기초한 방법을 사용하였지만 본 논문에서는 사용자 개입 없이 내용을 바탕으로 가중치를 계산, 자동으로 단일의 대표 구간을 추출하는 방법을 사용하였다.

3. 대표구간을 이용한 특징 추출

본 논문에서는 내용 기반 음악 장르 분류를 위하여 곡 전체 내용 중 대표 구간 하나를 선정하여 대표 구간 내의 특징들을 추출하고 이를 학습과 분류를 위한 특징으로 사용하는 방법을 사용한다.

전체 시스템 구조는 그림 1에서 보는 바와 같이 반복 구간 탐색을 통한 음악의 구조 분석, 분석된 반복 구간 정보를 바탕으로 한 대표구간 선택, 대표 구간으로부터 음악의 특징 벡터 추출, 특징 벡터를 사용한 학습 및 판정의 절차를 거친다.

3.1 반복 구간 탐색

곡의 대표 구간을 추출하기 위해선 우선 곡의 반복구간 탐색을 통한 구조 분석이 이루어져야 한다. 기본적으로 곡의 각 부분들은 소악절로 이루어져있고 이러한 소악절들 사이에는 반복적인 멜로디를 담고 있는 특징이 있다[15]. 우리가 일상적으로 접하고 있는 가요의 경우,

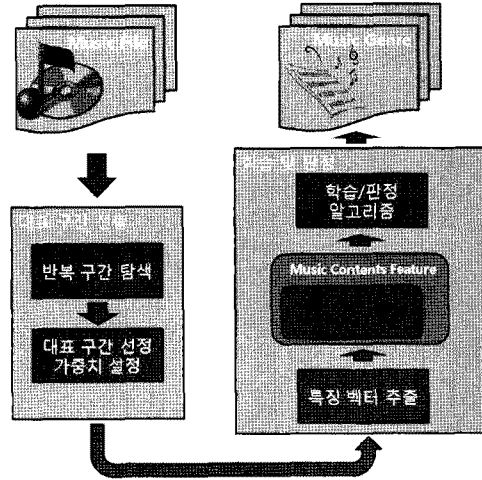


그림 1 전체 구조도

다른 가사이나 같은 멜로디를 가진 부분들이 반복되고 있는 것이 그 예이다. 본 논문에서는 비슷한 멜로디로 이루어진 반복구간을 탐색하기 위해 기본적으로 [15]에서 제안한 크로마그램 기반의 방법을 사용한다. [15]에서는 개략적인 방법만 소개되어 있어 구현에 필요한 부분(예, 크로마그램 그룹 간의 유사도, 반복구간 연결 알고리즘 등)을 추가하였다.

3.1.1 12 피치 클래스(Pitch Class)의 크로마그램 계산
크로마그램은 각 프레임(250ms 크기) 별로 계산하게 되는데 각 프레임에서 식 (1)[13]를 사용하여 12개의 키(key)(A, A#, B, C, C#, D, D#, E, F, F#, G, G#)에 대해 크로마를 구하게 된다. 음악분야에서는 모든 옥타브의 특정 피치 (또는 키)들의 집합을 피치클래스 (예: 피치클래스 C는 C_1, C_2, \dots)라 하며 같은 클래스에 속하는 피치들은 비슷한 질감 또는 색깔을 갖는 것으로 여겨지는데, 바로 이 피치클래스를 심리학자들은 크로마라고 한다[16].

$$v_c(t) = \sum_{h=Oct_L}^{Oct_H} \int_{-INF}^{INF} BPF_{c,h}(f_{cent}) \Psi_p(f_{cent}, t) df_{cent} \quad (1)$$

여기서, c 는 키를, t 는 프레임 인덱스를, h 는 옥타브를, f_{cent} 는 센트(cent) 단위의 주파수를, $\Psi_p(f_{cent}, t)$ 는 로그-스케일 기반의 파워스펙트럼을 나타낸다. $BPF_{c,h}(f_{cent})$ 는 Hann윈도우를 적용한 밴드-패스 필터로 옥타브 h 에 해당하는 영역에서 키 c 에 해당하는 주파수 영역만 추출하는 역할을 하며 식 (2)와 같이 구현된다. 크로마 정의[16]에 의하면 모든 옥타브에 존재하는 키를 대상으로 하여야 하지만 [13]에서는 Oct_L 옥타브에서 Oct_H 옥타브 범위를 지정하여 사용하였다. 본 논문의 실험에서는 1옥타브에서 5옥타브 범위($Oct_L = 1,$

$Oct_h = 5)$ 로 설정하여 크로마벡터를 추출하였다.

$$BPF_{c,h}(f_{cent}) = \frac{1}{2} \left(1 - \cos \left(\frac{2\pi(f_{cent} - (F_{c,h} - 100))}{200} \right) \right) \quad (2)$$

식 (2)에서 사용된 $F_{c,h}$ 는 옥타브 h 에서의 키 c 의 주파수를 센트 단위로 표현한 것으로 1200 센트는 1 옥타브, 100센트는 1키를 나타낸다(식 (3) 참고).

$$F_{c,h} = 1200h + 100(c + 1) \quad (3)$$

식 (1), (2), (3)에서 눈여겨 볼 것은 크로마그램을 구할 때 헤르츠 단위의 주파수 대신에 로그-스케일을 적용해서 얻은 센트 단위를 사용하였다는 것이다. 고속 푸리에 변환(FFT: Fast Fourier Transform)을 거쳐 나온 파워 스펙트럼으로부터 크로마그램을 추출하기 위해 선 주파수의 헤르츠 단위를 키의 특성상 로그-스케일의 단위로 변환시켜 주어야 할 필요가 있는데, 이 때 식 (4)을 사용한다.

$$f_{cent} = 1200 \times \log_2 \left(f_{hz} / 2^{12-5} \right) \quad (4)$$

3.1.2 크로마그램을 통한 반복 구간 탐색

곡 전구간의 크로마그램을 추출한 뒤 반복 구간을 탐색하기 위해서 추출한 크로마그램을 5초 간격(20개의 프레임)으로 3/4의 크기씩 중첩시키며 그룹화를 하게 된다. 이렇게 그룹화된 크로마그램 그룹(20개의 크로마그램으로 구성)을 사용하여 그룹간의 유사도를 계산하게 된다. 그룹간의 유사도는 거리에 반비례하는데, 그룹 G_1 의 크로마그램 그룹을 $x_1, x_2, x_3, \dots, x_N$ 라 하고 그룹 G_2 의 크로마그램 그룹을 $y_1, y_2, y_3, \dots, y_N$ 라 하면, G_1 과 G_2 의 거리는 표준 유클리디안 거리 $ED(x, y)$ 는 각 유클리디안 거리의 합으로 계산하게 된다(식 (5) 참고).

$$ED(x, y) = ED(x_1, y_1) + ED(x_2, y_2) + \dots + ED(x_N, y_N) \quad (5)$$

반복구간을 파악하기 위해서는 먼저 각 그룹에 대해 뒤 따르는 그룹 중 거리가 최소인 4개의 그룹을 선택하고 이 그룹들과의 거리를 유지한다. 앞으로, 각 그룹에 대한 이 4개의 거리 값을 편의 상 최소거리 목록이라 하겠다. 4개 그룹과의 거리만 유지하는 이유는 [15]에서와 마찬가지로 반복구간은 적어도 5번 이루어진다는 가정에 바탕을 둔다. 이 과정이 끝나면 그림 2에 기술된 방법에 따라 반복구간을 파악하게 된다. M번째 그룹에서 시작하는 N 개의 연속 그룹이 반복구간이 되기 위해서는 다음 조건을 만족시켜야 한다.

- i) N은 3 이상이어야 한다.
- ii) M번째 그룹에서 시작하는 N 개의 그룹의 최소거리 목록 집합들을 $T_M, T_{M+1}, \dots, T_{M+N-1}$ 라 하면, $T_M \cap T_{M+1} \cap \dots \cap T_{M+N-1} \neq \emptyset$ 이어야 한다.
- iii) M번째 그룹에서 시작하는 N+1 개의 그룹의 최소거리 목록 집합들을 $T_M, T_{M+1}, \dots, T_{M+N-1}, T_{M+N}$,

공통의 최소거리 목록 집합 S 를

$$T_M \cap T_{M+1} \cap \dots \cap T_{M+N-1} \text{라 하면,}$$

$$T_{M+N} \cap S = \emptyset \text{이어야 한다.}$$

본 논문에서는 파악되는 반복구간 수를 줄이기 위해 [15]에서와 동일하게 반복구간의 길이는 10초 (중첩을 고려하면 최소 3개의 연속 그룹이 합쳐져야 10초를 넘을 수 있다) 이상이 되어야 하는 것으로 가정하였다. 첫 번째 조건은 이를 나타낸다. 두 번째 조건은 N 개 그룹의 최소목록에 적어도 공통된 값이 있어야 함을 나타낸다. 이는 공통된 값을 d 라하면 N 개 그룹이 d 그룹 후에 반복됨을 의미한다. 즉, M번째 그룹에서 시작하는 N 개의 연속 그룹이 반복구간이 되면 $d \in S - S'$ (여기서, $S = T_M \cap T_{M+1} \cap \dots \cap T_{M+N-1}$, $S' = S \cap T_{M+N}$) 만큼 떨어진 위치에 있는 N 개의 연속 구간, 즉 G_{M+d} , $G_{M+d+1}, \dots, G_{M+d+N-1}$ 도 반복 구간이 된다. 이러한 관계는 재귀적으로 반복되지는 않기 때문에 G_{M+d} , $G_{M+d+1}, \dots, G_{M+d+N-1}$ 이 반복구간이 되었다고 해서 G_{M+2d} , $G_{M+2d+1}, \dots, G_{M+2d+N-1}$ 가 반복구간이 되지는 않는다. 세 번째 조건은 N 개 그룹의 최소목록 공통 값이 N+1 번째 그룹의 최소목록에는 없어야 함을 나타낸다. 이는 공통 값을 갖는 최대 연속그룹이 반복구간이 됨을 의미한다.

그림 2의 알고리즘은 각 그룹을 시작 그룹으로 하여 시작그룹과 위의 세가지 조건을 만족하는 연속 그룹들

```

M = 0 // 시작그룹의 인덱스
T_M = { D | D=ED(M, A),
        A ∈ M과 거리가 최소인 4개의 그룹}

WHILE ( !IsLast(M) )
    S = T_M
    LINK_M = EMPTY // 검출된 반복구간의 연결
                    정보가 저장.
    N = 1
    WHILE( !IsEmpty(S) )
        S' = S ∩ T_{M+N}

        IF N>4 THEN
            LINK_M = LINK_M ∪ (S - S')
        END_IF

        S = S'
        N = N + 1
    END_WHILE
    M = M + 1
END_WHILE
    
```

그림 2 반복 구간 연결 알고리즘

Group	0	1	2	3	4	5	...	9	10	11	12	13
최소거리 목록 T	4	9	5	5	6	8						
	7	11	9	9	8	12						
	9	15	10	11	9	15	...					
	18	20	12	12	10	20						

그림 3 반복구간 탐색 예

을 찾는 알고리즘이다. 첫 번째 while 문은 각 그룹을 처리하기 위한 것이며 내부의 while 문은 연속 3개 이상의 그룹이 최소목록 공통 값을 갖고 있는지를 조사하게 된다. 내부 while 문을 마치고 나며 $LINK_M$ 에는 연속 3개 이상의 그룹의 최소목록 공통값이 저장된다.

그림 3은 간단한 최소거리 목록의 예를 나타낸다. 그림 3의 첫 번째 열은 각 그룹의 번호, 각 행은 해당 그룹의 최소거리 목록 T를 나타낸다.

이 예를 통하여 위에서 기술한 반복구간이 되기 위한 조건을 확인해 보도록 한다. 그룹 0부터 4까지는 최소거리 목록 중 공통의 값 9를 가지고(두 번째 조건), 0~4까지의 N의 값은 5가 된다(첫 번째 조건). 그리고 그룹 5에서는 그룹 0에서 4까지의 공통의 최소거리 값인 9가 없기 때문에(세 번째 조건) 더 이상의 구간 연결이 이루어지지 않고 0~4의 그룹을 연결하여 한 개의 반복구간이 되고 그리고 이에 반복되는 쌍으로 아홉 번째 떨어진 9~13의 그룹군이 대응되는 반복구간으로 설정된다.

3.1.3 대표 구간 선택

반복구간 탐색이 완료된 후, 음악 특징을 추출하기 위해 하나의 구간, 즉 대표 구간을 추출하여 사용하게 된다. 이를 위해 반복구간이 얼마나 내용 추출에 있어 중요한지를 판별하여야 하는데, 본 논문에서는 두 가지 가중치를 고려하였다. 첫 번째는 위치에 대한 가중치 (W_p)로, 기존의 음악 썸네일 추출관련 연구들[14,15]에 따르면 대다수의 음악들의 주요 내용이 곡 전체 중 중앙 부분에 위치하는 경향이 있는 것으로 나타나고 있다. 이를 바탕으로 곡 전체 길이 S_t 의 중앙 위치인 $S_t/2$ 에 가중치를 1로 하여 외부로 갈수록 점차 낮아져 시작부분과 끝부분은 0이 되는 Hamming윈도우를 사용하여 위치 가중치를 부여하였다. 또 다른 가중치는 평균 에너지를 에너지 가중치 W_E 로, 이는 반복 구간 중 낮은 에너지를 가지는 부분은 서주나 간주일 가능성이 높고, 곡 내에 하이라이트 구간이 다른 구간에 비해 에너지가 높다는 가정에 근거한 것이다.

본 논문에서는 이 두 가중치의 유효성을 판단하기 위해 4장에 기술된 바와 같이 각 가중치를 단독으로 사용할 경우와 식 (6)과 같이 두 가중치를 곱하여 사용할 경우의 실험을 하였다. 실험 전에는 두 가중치를 곱하여

사용할 경우가 성능이 좋을 것으로 예상했으나 실제 실험 결과는 그렇지 않았다. 결론적으로 위치 가중치를 고려하지 않아도 에너지 가중치 만으로도 충분히 대표구간을 선택할 수 있음을 알 수 있었다.

$$W = W_p \times W_E \tag{6}$$

여기서, W는 대표구간 가중치를, W_p 는 위치 가중치를, W_E 는 구간의 에너지 가중치를 나타낸다.

3.2 특징 벡터 추출

대표 구간이 추출되면 내용 기반 장르 판별을 위해 곡의 특징 벡터를 추출해야 한다. 본 연구에서는 기존 연구들에서 사용된 음악 특징들을 특징 벡터의 요소로 선택하였다. 선택된 특징들과 각 특징의 간략한 설명은 표 1과 같다.

곡 전체에 푸리에 변환을 적용하여 음색 특징들을 추출하게 되면 푸리에 변환의 특성 상 시간 정보가 사라지게 되고 전역적인 주파수의 정보만 남게 되어 변화량이 많은 음악의 특색을 추출하기가 어렵게 된다. 따라서 음향 신호 부분마다 차이가 많이 나는 음색 특징들을 추출하기 위해서 먼저 작은 윈도우를 사용한 단구간 푸리에변환(STFT)을 적용하여 국부적인 음색특징을 추출한 후 다시 이들의 전역적인 표준 분산의 특징(평균, 표준편차)을 추출하게 된다. 이러한 방법을 사용하게 되어 음향 신호의 변화량이 많은 특징에 따라 전체적 프레임들의 특징 계수의 분산의 차이가 심하게 나타나 표준 분산으로 모델링하기에 무리가 있다. 이를 위해 [2]에서는 Texture Window와 Analysis Window를 제안하여 음색 특징의 추출에 사용하였다. 일반적으로 고속 푸리에 변환을 사용하기 위하여 각 신호에 적용시키는 작은 윈도우로 Analysis Window를 사용하고 음향 신호의 변화량에 따른 국지적인 표준 분산의 특징을 적용시키기 위해 Texture Window를 적용시켜 국지적인 음색특징의 표준 분산적 특징(평균, 표준편차)을 표현하게 하였다(그림 4 참고).

표 1 실험에 사용한 특징

특정군	특징	설명	
음색 특징	Spectral Shape	Centroid	주파수의 무게 중심
		Spread	무게중심으로부터 분산 범위
		Roll-off	파워 스펙트럼의 Rolloff 지점
	Spectral Contrast	Flux	인접 프레임간의 파워스펙트럼의 분산량
		Valley	주파수 밴드별 낮은 에너지의 합
		Contrast	주파수 밴드별 에너지의 대비정도
MFCC	0~12 order	사람 청각 모델에 기반한 특징	
DWCHs	7 Level coefficients	밴드별 웨이블릿 히스토그램 계수	

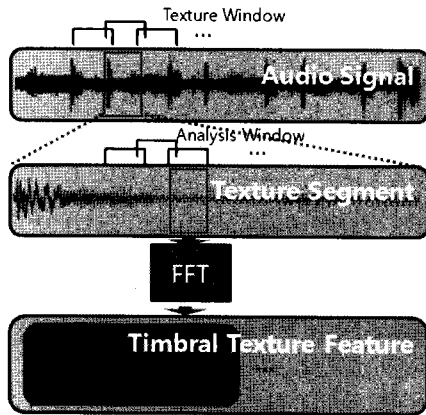


그림 4 Timbral Texture 특징 추출

본 논문에서도 이 Texture Window를 적용시켜 음색 특징을 추출하였다.

음색 특징을 추출하기 위한 파라미터 값으로는 23ms의 Analysis Window를 20ms씩 이동시키며 음색 특징을 추출하였으며, 0.6s의 Texture Window를 0.5s씩 이동시키며 국부적 Timbral Texture의 분산 특징으로 Texture 내의 평균과 표준편차를 추출하였다.

음색 특징에는 여러 가지가 있으나 본 논문에서는 주파수 스펙트럼의 여러 분산학적 특징을 나타내는 Spectral Shape, 사람의 청각 모델에 기반한 MFCC 그리고 옥타브 밴드 기반의 음색의 대비를 표현하는 Spectral Contrast를 사용하였다. MFCC 특징 추출 시 본 논문에서는 이산 코사인 변환(DCT)의 상위 13차 계수까지를 사용하였다. 그리고 Spectral Contrast 특징 추출 시, [3]의 연구에서와 동일하게 각 6 옥타브 밴드 - 0~500, 500~1000, 1000~2000, 2000~4000, 4000~8000, 8000~16000(Hz) - 별로 통과한 파워 스펙트럼의 하위 $\alpha\%$ 에너지합인 $Valley_{band}$, 그리고 이 $Valley_{band}$ 값과 상위 $\alpha\%$ 에너지합인 $Peak_{band}$ 의 차이인 $SC_{Cont_{band}}$ 로 구성된 12차의 특징벡터를 추출하여 사용하였다. 이때 α 값으로 0.9를 사용하였다. [3]에서는 12차원의 특징벡터를 추출한 뒤 차원 축소를 위해 K-L변환을 사용하였으나 본 연구에서는 K-L변환을 사용하지 않고 12차의 특징벡터 모두를 사용하였다.

DWCHs는 [4]에서 제안된 음악 특성으로 장르 판별에 좋은 성능을 나타내어 본 논문에서도 이 특징을 사용하였다. DWCHs 특징을 추출하기 위하여 음원을 다우비치 웨이블릿 필터(Db8)를 사용하여 7레벨 변환 후 각 주파수 밴드별로 웨이블릿 계수를 히스토그램화하여 각기 첫 세개의 모멘트(평균, 분산, 왜도)와 밴드별 에너지(웨이블릿 계수 제곱의 합)를 추출하여 사용하였다.

4. 음악 장르 분류 실험

4.1 실험의 구성

분류된 장르의 음악 데이터베이스를 구축하기 위하여 2005 MIREX¹⁾에서 제공한 음악 데이터들과 MIREX 학습용 데이터의 경우 장르별 파일의 수가 클래식 집중된 경향이 있어 장르별 데이터의 수를 맞추기 위해, 인터넷으로부터 수집한 MP3 파일²⁾을 추가한 후 음악 정보 웹사이트인 AllMusic³⁾의 분류에 맞춰 8개의 장르로 분류된 데이터베이스를 구축하였다. 각 8개의 장르는 클래식(Classic), 일렉트로닉(Electronic), 재즈(Jazz), 메탈(Metal), 리듬 앤 블루스(R&B), 락(Rock) 그리고 월드(World)로 각 장르별 약 100개의 파일로 구성되어 총 700개의 파일로 이루어졌다. 음악 데이터베이스를 구성하고 있는 모든 음악 파일의 포맷은 MP3이고 샘플링 레이트는 44100Hz이다.

내용 기반 분류를 위한 학습 알고리즘은 일반적으로 학습에 있어 분류율이 좋기로 알려진 Support Vector Machine (SVM)⁴⁾, 인공신경망 (ANN) 그리고 k값이 1인 Nearest-Neighbor(k-NN(1))와 k값이 5인 Nearest-Neighbor(k-NN(5)) 알고리즘을 사용하였고, 각 학습 알고리즘의 성능을 평가하기 위해 n-Fold Cross Validation⁵⁾을 사용하였다. 각 학습 알고리즘에 대한 평가는 데이터마이닝 라이브러리인 Weka⁶⁾를 사용하였다.

4.2 특징 비교

우선 실험에 사용될 특징은 3.2 절에 기술된 음색 특징인 Spectral Shape(SS), Spectral Contrast(SC), MFCC와 DWCHs 특징을 사용하였다. 우선 각 특징들의 성능 비교를 위해, 곡의 특징은 중앙에 있을 확률이 높다는 가정 하에 곡의 중앙 부분 15초에서 특징들을 추출, 7가지 특징 조합에 대한 성능을 살펴보면 표 2와 같다.

[4]의 결과에 의하면 DWCHs의 특징이 장르 판정에 있어 좋은 성능을 보여주는 것으로 되어 있으나 본 실험 결과에 의하면 음색 특징(SS_SC_MFCC)만을 사용하였을 때와 크게 차이가 없는 것으로 나타났다. 또한, 옥타브 밴드 기반의 Spectral Contrast을 다른 음색적 특징과 같이 사용한 경우가 사용하지 않은 경우보다

1) <http://www.music-ir.org/mirex/2005/index.php/>

Audio_Genre_Classification

2) 모든 MP3파일은 스테레오 형식이고 실험을 위한 특징 추출을 위해 모노 채널 형식으로 변환하여 처리하였다.

3) <http://www.allmusic.com/>

4) SVM은 커널 함수(kernel function)의 따라 파라미터(parameter)의 선택에 따른 성능의 차이가 많이 나게 된다. 본 논문은 동일한 비교를 위하여 선형 커널 함수(linear kernel function)를 사용하였다.

5) 본 실험의 Fold 계수 n은 5로 지정

6) <http://www.cs.waikato.ac.nz/ml/weka/>

표 2 특징에 따른 장르 분류 성능 (음악 중앙 구간 추출)

학습 알고리즘	특징들의 조합 방법						
	SS_DWCH	SS_MFCC	SS_MFCC_DWCH	SS_SC	SS_SC_MFCC	SS_SC_DWCH	ALL
SVM	53.8462	58.6895	57.6923	67.2365	67.5214	68.0912	66.3818
ANN	53.5613	56.1254	55.6980	64.5299	66.9516	66.2393	65.3846
k-NN(1)	47.1510	51.1396	51.7094	60.3989	59.6866	60.5413	57.9772
k-NN(5)	51.7094	53.7037	54.2735	62.9630	63.6752	61.9658	60.6838

(SS_MFCC vs. SS_SC_MFCC) SVM 알고리즘을 기준으로 약 10% 정도의 높은 성능 향상을 보여주는 것으로 나타났다.

SS 특징에 SC 특징을 조합했을 때 그 성능 향상이 가장 뚜렷하며 SS와 SC 조합에 다른 특징을 추가했을 경우는 성능 향상이 미미하거나 학습 알고리즘에 따라서 더 떨어지는 경우도 있었다. 표 1에 기술한 특징 전체를 조합했을 경우에도 마찬가지로 SS와 SC 조합에 비해 뚜렷한 성능 향상을 얻을 수 없었다. 하지만 특징 조합에 의해 약간의 성능 차이가 있을 수 있기 때문에 다음에 살펴볼 것 대표구간 선택에 의한 성능 평가에서는 표 1의 특징들을 모두 사용하여 평가하였다.

4.3 반복구간을 이용한 대표구간 선출과 대표구간을 이용한 장르 판정 성능

대표구간을 이용할 경우의 성능 평가를 위하여 두 가지의 비교 실험을 하였다. 첫 번째는 본 논문에서 제안한 방법에 의해 선택된 대표구간에서 특징을 추출한 경우와 기존 방법에서 사용하는 고정구간에서 특징을 추출할 경우의 성능 비교를 하였다. 두 번째는 대표구간 선택 방법의 타당성을 보이기 위해 여러 반복구간 중에서 대표구간을 선택하는 방법을 달리하면서 실험하였다.

기존 고정구간 방법과의 성능 비교를 위하여 노래 전체 길이의 20%, 50%, 80% 지점에서 특징을 추출할 경우와 본 제안 방법에 의해 선택된 대표구간에서 추출한 경우의 비교 실험을 하였다. 실험 결과는 표 3과 같다. 실험 결과를 살펴볼 때, 고정구간 방법에서는 곡의 80% 위치에서 특징을 추출하고 이를 신경망 학습알고리즘을 사용하여 학습할 경우의 성능이 가장 좋을 수 있다. 본 연구에서 제안한 방법인 대표구간 선택에 의한 장르 분류인 경우도 신경망 학습알고리즘을 사용할 경우가 가장 좋았으며 고정위치의 최대 성능인 68.6610%

표 3 고정 구간 추출과 대표 구간 추출 방법의 학습 알고리즘별 성능 비교 표

학습 알고리즘	추출 구간			대표구간 선택 (Select Part)
	고정 구간 추출			
	20%	50%	80%	
SVM	66.0969	66.3818	66.9516	69.6275
ANN	66.0969	65.3846	68.6610	72.0630
k-NN(1)	53.2764	57.9772	55.6980	64.3266
k-NN(5)	56.9801	60.6838	58.8319	64.8997

보다 약 4% 향상된 72.0630% 분류 성능을 보여주었다 (표 3 참고).

대표구간은 3.1.2절에 기술한 반복구간 탐색 방법에 의해 파악된 반복구간들 중에서 3.1.3절의 휴리스틱한 방법을 적용시켜 하나를 선택하였다. 대표구간 선택 방법에 따라 성능이 달라질 수 있기 때문에 본 논문에서는 기본적인 선택 방법으로 반복구간 중 첫 구간, 마지막 구간을 선택할 경우, 3.1.3절에 기술한 위치가중치와 에너지 가중치를 단일로 사용할 경우와 제안된 두 가중치를 모두 사용하여 선택한 경우의 성능 비교 실험을 하였다. 실험 결과는 표 4와 같다.

표 4에서 보듯이, 반복구간 중 첫 번째와 마지막 구간을 선택한 두 경우에 비해 가중치를 적용하여 선택한 경우가 모두 높은 성능을 보여주고 있다. 단일의 가중치를 적용시켜 선택한 경우를 보면 에너지 가중치만을 사용한 경우가 더 좋은 성능을 보여주고 있다. 또한 본 연구에서 제안한 3.1.3의 두 개의 가중치 모두를 사용한 대표구간 선택방법과 에너지 가중치만을 사용한 구간 선택의 성능을 살펴보면 1%내의 차이로 서로 대동소이한 성능을 보이고 있다. 이러한 결과는 단순히 위치의 정보만을 고려하여 특정 위치의 구간을 선택할 경우 어

표 4 반복구간 중 고정된 구간 선택과 가중치별 선택 방법의 성능 비교 표

학습 알고리즘	추출 구간				
	검출된 반복구간 중 선택		가중치 적용		
	First	Last	위치(중앙)	에너지	위치*에너지
SVM	57.1633	66.7622	67.3820	70.3863	69.6275
ANN	59.0258	65.3295	67.3820	70.8155	72.0630
k-NN(1)	44.5559	52.7221	58.5122	64.8069	64.3266
k-NN(5)	47.2779	57.3066	61.2303	66.2375	64.8997

는 정도의 성능 향상을 얻을 수 있지만, 다양한 구조를 가지는 음악의 특성을 고려할 때 그리 권장할 방법이 아님을 의미한다. 이와 반대로 에너지 정보는 위치 정보가 갖는 이상의 정보를 제공해 줄 수 있다.

5. 결론

본 논문에서는 내용기반 장르 분류를 위하여 곡 내 반복구간 탐색 후 위치와 에너지 가중치를 통하여 대표 구간을 선택하고 이 구간에서 특징을 추출하여 이를 활용하는 방법에 대해 살펴보았다. 곡의 반복구간 탐색에 관련된 연구는 음악의 썸네일 추출이나 지능형 구간 탐색과 같은 음악 청취자나 검색자를 위한 연구에서 진행되어 왔으나 장르분류와 같은 내용기반 분류 및 판별 분야 쪽에서는 연구 사례가 없었다. 반복구간을 이용한 대표구간 탐색 방법을 장르 분류 시스템에 적용시켜 본 결과, 단순한 곡의 고정된 구간에서 특징을 추출한 것보다 주목할 만한 성능 향상이 있었다.

본 연구에서는 여러 반복구간 중 하나를 대표구간으로 선택하고 이로부터 특징을 추출하여 사용하였다. 나머지 반복구간의 특징은 전혀 고려되어지지 않고 있는데 이를 효과적으로 이용한다면 좀 더 분류 성능을 향상시킬 수 있을 것이다. 앞으로 곡의 반복구간 중 하나의 대표구간이 아닌 여러 반복 구간을 선택하여 이로부터 특징을 추출하는 연구를 수행하고자 한다. 다중 대표 구간을 선택하는 방법과 선택된 대표구간의 정보를 조합하는 방법을 달리하면서 성능을 비교해 보고자 한다.

참고 문헌

- [1] Aucouturier, J.-J. and Pachet, F., "Representing musical genre: A state of the art," *Journal of new musical research*, Vol. 32, No. 1, pp. 83-93, 2003.
- [2] Tzanetakis, G. and Cook, P., "Musical genre classification of audio signals," *Speech and Audio Processing*, IEEE Transactions on Vol. 10, No. 5, pp. 293-302, 2002.
- [3] Jiang, D., Lu, L., Zhang, H., Tao, J. and Cai, L., "Music type classification by spectral contrast feature," *Proc. of ICME '02*, Vol. 1, pp. 113-116, 2002.
- [4] Li, T., Ogihara, M. and Li, Q., "A comparative study on content-based music genre classification," *Proc. of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 282-289, 2003.
- [5] Li, T. and Ogihara, M., "Music genre classification with taxonomy," *Proc. of ICASSP '05*, Vol. 5, pp. 197-200, March 2005.
- [6] Burred, J. J. and Lerch, A., "A Hierarchical Approach to automatic Musical Genre Classification," *Proc. of the 6th Int. Conference on Digital Audio*

Effects (DAFx-03), 2003.

- [7] DeCoro, C., Barutcuoglu, Z., and Fiebrink, R., "Bayesian Aggregation for Hierarchical Genre Classification," *Proc. of International Symposium on Music Information Retrieval 2007*, 2007.
- [8] Kim, Y. K. and Brian, Y., "Singer Identification in Popular Music Recordings Using Voice Coding Features," *Proc. of Int'l Conf. on Music Information Retrieval*, 2002.
- [9] Zhang, T., "Automatic Singer Identification," *Proc. of IEEE Int'l Conf. on Multimedia and Expo*, IEEE CS Press, 2003.
- [10] Shao, X., Maddage, N.C., Xu, C. and Kankanhalli, M.S., "Automatic music summarization based on music structure analysis," *Proc. of ICASSP'05*, Vol. 2, pp. 1169-1172, 2005.
- [11] Shiu, Y., Jeong, H. and Kuo, C.-C. J., "Musical structure analysis using similarity matrix and dynamic programming," *Proc. of SPIE, Multimedia systems and applications*, Vol. 3, pp. 398-409, 2005.
- [12] Paulus, J. and Klapuri, A., "Music Structure Analysis by Finding Repeated Parts," *Proc. of ACM AMCM'06*, pp. 59-67, 2006.
- [13] Goto, M., "SmartMusicKIOSK: music listening station with chorus-search function," *Proc. of the 16th annual ACM symposium on User interface software and technology*, pp. 31-40, 2003.
- [14] Lu, L. and Zhang, H., "Automated extraction of music snippets," *Proc. of the 11'th ACM international conference on Multimedia*, pp. 140-147, 2003.
- [15] Zhang, T. and Samadani, R., "Automatic Generation of Music Thumbnails," *Proc. of IEEE International Conference on Multimedia and Expo*, pp. 228-231, 2007.
- [16] http://en.wikipedia.org/wiki/Pitch_class



이 중 인

2007년 금오공과대학 컴퓨터공학부 졸업(학사). 2007년~현재 금오공과대학 컴퓨터공학부 석사과정 관심분야는 인공지능, 패턴인식, 소프트웨어 공학



김 병 만

1987년 서울대학교 컴퓨터공학과(공학사). 1989년 한국과학기술원 전산학과(공학석사). 1992년 한국과학기술원 전산학과(공학박사). 1992~현재 금오공과대학교 교수. 1998년~1999년 미국 UC, Irvine 대학 방문교수. 2005년~2006년 미국 콜로라도 주립대학 연구교수. 관심분야는 인공지능, 정보검색, 정보보안