

Gaussian Mixture Model을 이용한 다중 범주 분류를 위한 특징벡터 선택 알고리즘

준회원 문선국*, 최택성*, 정회원 박영철**, 종신회원 윤대희*

Feature Selection for Multi-Class Genre Classification using Gaussian Mixture Model

Sun-Kuk Moon*, Tack-Sung Choi* *Associate Members,*

Young-Cheol Park** *Regular Member,* Dae Hee Youn* *Lifelong Member*

요약

본 논문에서는 내용 기반 음악 범주 분류 시스템에서 다중 범주를 위한 특징벡터 선택 알고리즘을 제안한다. 제안된 특징벡터 선택 알고리즘은 분리 성능을 측정할 때 가우시안 혼합 모델(Gaussian Mixture Model: GMM)을 기반으로 GMM separation score을 측정함으로써 확률분포 및 분리 성능 추정의 정확도를 높였고, sequential forward selection 방법을 개선하여 이전까지 선택된 특징벡터들이 분리를 잘 하지 못하는 범주들을 기준으로 다음 특징벡터를 선택하는 알고리즘을 제안하여 다중 범주 분류의 성능을 높였다. 제안된 알고리즘의 성능 검증을 위해 음색, 리듬, 피치 등 오디오 신호의 특징을 나타내는 다양한 파라미터를 오디오 신호로부터 추출하여 제안된 특징벡터 선택 알고리즘과 기존의 알고리즘으로 특징벡터를 선택한 후 GMM classifier와 k-NN classifier를 이용하여 분류 성능을 평가하였다. 제안된 특징벡터 선택 알고리즘은 기존 알고리즘에 비하여 3%에서 8% 정도의 분류 성능이 향상된 것을 확인할 수 있었고 특히 낮은 차원의 특징벡터의 분류 실험에서는 분류 정확도 측면에서 5%에서 10% 향상된 좋은 성능을 보였다.

Key Words : Feature selection algorithm, Music Information Retrieval, Genre Classification, Gaussian Mixture Model

ABSTRACT

In this paper, we proposed the feature selection algorithm for multi-class genre classification. In our proposed algorithm, we developed GMM separation score based on Gaussian mixture model for measuring separability between two genres. Additionally, we improved feature subset selection algorithm based on sequential forward selection for multi-class genre classification. Instead of setting criterion as entire genre separability measures, we set criterion as worst genre separability measure for each sequential selection step. In order to assess the performance proposed algorithm, we extracted various features which represent characteristics such as timbre, rhythm, pitch and so on. Then, we investigate classification performance by GMM classifier and k-NN classifier for selected features using conventional algorithm and proposed algorithm. Proposed algorithm showed improved performance in classification accuracy up to 10 percent for classification experiments of low dimension feature vector especially.

* 연세대학교 전기전자공학과 디지털신호처리 연구실(skmoon@dsp.yonsei.ac.kr, ancestry@dsp.yonsei.ac.kr, dhyoun@yonsei.ac.kr),

** 연세대학교 컴퓨터정보통신공학부 (young00@dragon.yonsei.ac.kr)

논문번호 : KICS2007-04-195, 접수일자 : 2007년 4월 25일, 최종논문접수일자 : 2007년 9월 27일

I. 서론

정보통신 기술의 발달로 음악 정보의 양이 기하급수적으로 늘어나게 되고 음악 콘텐츠에 대한 정보 이용자의 다양한 요구가 발생함에 따라 음악 정보를 자동적이고 체계적으로 관리하기 위한 시스템이 필요하게 되었다. 내용 기반 범주 분류 시스템은 기존의 수작업으로 이루어지던 텍스트 기반의 분류 시스템과 달리 정보의 내용을 수학적으로 분석하여 구조화된 기준에 따른 대표적인 특성을 추출하고 각 범주마다 마련된 기준에 따라 자동적으로 범주를 분류한다. 이러한 분류와 그에 따른 검색 작업은 음악 콘텐츠의 신속하고 정확한 정보를 제공할 수 있기 때문에 멀티미디어 관련 응용 연구에 필수적인 기반 기술로 사용될 수 있다.

내용 기반 음악 범주 분류 및 검색 시스템은 처음 음악과 음성, 주변 잡음 등을 구분하는 연구에서 시작하여 음악 범주 안에서 정보를 검색하는 분야로 발전해 왔다^[1,2]. 이러한 내용 기반 음악 범주 분류 시스템은 디지털 신호처리를 이용하여 오디오의 Pitch, 음색 (timber), 박자(Rhythm) 등의 특징을 추출하여 이러한 특징들과 데이터베이스 내의 오디오와 비교하여 검색하는 기본구조를 갖고 있다^[2-6]. 대표적인 방법으로는 15초 미만의 음향 효과들에 대하여 신호의 크기, Pitch, brightness, 대역폭, 하모니 등의 특징들을 추출하여 DB내의 유사한 오디오를 검색하는 방법^[3]과 음악 장르의 계층적인 자동 분류를 위하여 STFT 기반의 오디오의 표면적 특성과 Wavelet Transform 기반의 리듬, Pitch 등의 오디오 특성 정보들을 추출하여 분류하는 방법^[2]이 있다. 특히 참고문헌 [2]의 방법에서는 k-NN, Gaussian, GMM 등의 classifier를 사용하여 음성/음악/잡음 분류에서 90%, Classic, Jazz, Folk, R&B 등 4가지의 Pop 음악 장르 분류에서는 약 74% 정도의 성공률을 보이고 있다.

기본적인 구조를 가지고 있는 분류시스템의 성능을 높이기 위해서는 먼저 여러 가지 특징들 중 범주의 특성을 잘 나타내는 특징벡터를 선택하는 알고리즘이 필요하다. 특징벡터는 직관적으로 범주들 사이에서 Euclidean distance 등을 이용하여 특징벡터의 거리를 재는 것으로 확인할 수 있다^[7]. 혹은 inertia ratio maximization 방법을 사용할 수도 있다^[4]. 이와 같은 특징벡터간의 거리를 측정하는 방법은 sequential forward selection 방법^[7]이나 Genetic algorithm^[8] 등과 같은 여러 차수의 특징벡터를 선택하는 알고리즘으로 원하는 차수의 특징벡터를 선택

할 수 있다. 그러나 이러한 기존의 방법들은 특징벡터의 분포를 정규분포로 가정하기 때문에 실제 인식률과 차이가 날 수 있는 문제점이 있다. 또한 범주의 개수가 3개 이상인 다중 범주에 대하여 특징벡터를 선택할 때에는 상대적으로 분리가 잘 되는 범주간의 거리가 특징벡터에 따른 값의 변화가 더 심하기 때문에 여러 범주간의 거리를 동등하게 평균을 구하여 분리 성능을 측정하는 기존의 방법은 최적의 특징벡터를 선택하지 못하는 문제가 발생할 수 있다.

본 연구에서는 GMM을 기반으로 제안된 거리 측정 방법과 기존의 특징벡터를 선택하는 알고리즘을 비교하고 성능을 향상시킬 수 있는 방법을 제안한다. 또한 기존의 Sequential forward selection 방법을 개선하여 낮은 차수에서 효과적인 특징벡터를 선택하는 방법을 제안한다. 특징벡터 선택 알고리즘의 성능을 평가하기 위하여 classifier로 널리 사용되는 GMM classifier와 k-NN classifier로 실험하여 범주 분류 성능을 평가한다.

본 논문의 구성은 다음과 같다. 먼저 II장에서는 기존의 특징벡터 선택 알고리즘과 제안된 특징벡터 선택 알고리즘에 대하여 설명하였다. III장에서는 특징벡터 선택 알고리즘을 성능을 시험하기 위하여 논문에서 사용된 특징벡터들에 대해 알아보았다. IV장에서는 특징벡터 선택 알고리즘과 차수에 따른 범주 분류 성능을 비교 분석하였고, 마지막으로 V장에서 결론을 맺는다.

II. 특징벡터 선택 알고리즘

본 장에서는 음악 장르간의 특징을 비교적 잘 구분하는 특징벡터들을 효과적으로 선택하기 위해 사용되는 특징벡터 선택 알고리즘에 대해 설명한다. 먼저 비교의 대상이 되는 기존의 특징벡터 선택 알고리즘들을 알아보고 기존의 알고리즘의 성능을 높이기 위하여 제안된 알고리즘을 소개한다.

2.1 기존의 분리 성능 측정법

가장 간단한 형태의 선택 알고리즘은 각 분류에 해당하는 데이터의 평균으로 서로의 Euclidean distance를 측정하는 방법이다^[7]. 모두 M 개의 분류가 있고 각 분류의 평균을 m_i 라 할 때 다음과 같이 구해진다.

$$ED = \sum_{i=1}^M \sum_{j=i+1}^M \|m_i - m_j\| \quad (1)$$

한편 inertia ratio maximization 방법^[4]은 분류내의 편차는 작고 각 분류간의 편차는 큰 통계적 특성을 가진 특징벡터를 선택하는 방법이다. Inertia ratio는 분류간의 편차와 분류내의 편차의 비율로 다음과 같이 정의된다.

$$IR = \frac{\sum_{i=1}^M (n_i / N) \|m_i - m\|}{\sum_{i=1}^M \left((1/N_i) \sum_{n_i=1}^{N_i} \|x_{n_i} - m_i\| \right)} \quad (2)$$

여기서 M 은 분류의 개수를, n_i 는 i 번째 분류의 데이터의 개수를, N 은 전체 데이터의 개수를, m_i 는 i 번째 분류의 특징벡터의 평균을, m 은 전체 분류의 특징벡터의 평균을, x_{n_i} 는 n_i 번째 데이터를 각각 나타낸다.

평균의 Euclidean distance를 이용한 방식은 매우 간단하고 다중 차수의 특징벡터에 대해서도 쉽게 거리를 구할 수 있지만 특징벡터의 대푯값이 평균으로만 나타내어지기 때문에 분류 간의 분포가 다른 경우 실제 시스템에서의 인식률과 차이가 날 수 있다. 이러한 단점을 어느 정도 보완한 방법인 Inertia ratio maximization 방법은 분류 내의 편차를 기준에 포함시키기 때문에 분류간의 분포에 따른 영향이 줄어드는 장점이 있다. 그러나 분류간의 편차를 구할 때 Euclidean distance와 마찬가지로 분류의 대푯값이 평균으로만 계산되어 정보의 손실이 일어날 수 있고 훈련 데이터가 정규분포와 가깝지 않을 경우 분류의 분포간의 거리와는 차이가 날 수 있다. 실제 오디오 신호에서 추출된 특징벡터들은 정규분포로 가정하기에는 부적절한 것들이 많았다. 따라서 정규분포로 가정하기 어려운 특징벡터의 범주간의 거리도 잘 나타낼 수 있는 확률 분포 모델링 기법을 도입하여 특징벡터 선택 알고리즘을 고안할 필요가 있을 것이다.

2.2. GMM을 이용한 분리 성능 측정법

본 논문에서는 각 범주의 분포를 정규분포로 가정하는 대신 확률 분포를 추정하는 데 많이 사용되고 있는 가우시안 혼합 모델(Gaussian mixture model)을 이용하였다. 가우시안 혼합 모델은 여러 개의 Gaussian 분포를 사용하여 임의의 확률 분포를 모델링하는 기법으로써 Expectation-maximization 알고리즘으로 최적화될 수 있다^[4]. 가우시안 혼합 모델로 모델링 된 확률 분포는 다음과 같은 식으로 표현할 수 있다.

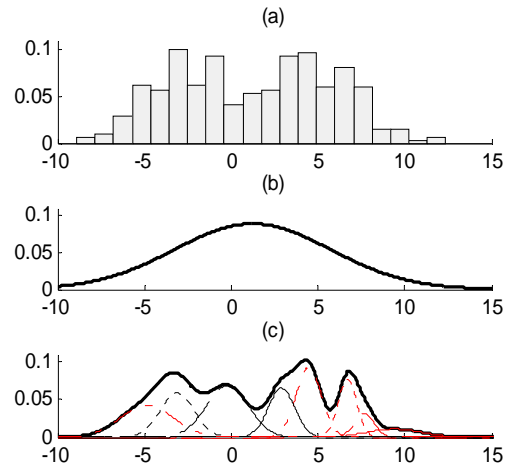


그림 1. (a) 3번째 MFCC의 히스토그램 (b) 3번째 MFCC를 정규 분포로 모델링 한 확률 분포 (c) GMM으로 모델링 한 확률 분포

$$p(\bar{x} | \lambda) = \sum_{i=1}^M w_i b_i(\bar{x}) \quad (3)$$

여기서 w_i 는 i 번째 Gaussian mixture의 가중치를, $b_i(\bar{x})$ 는 입력벡터 x 의 i 번째 Gaussian mixture의 확률 분포를 나타낸다^[15]. 그림 1에서는 Classic 장르의 Training data에 포함되어 있는 320곡에서 추출한 곡 전체의 3번째 MFCC 평균값들의 분포를 나타내고 있다. 에서 Classic 장르의 3번째 MFCC 평균값 파라미터를 정규 분포로 가정하여 모델링 한 분포와 8개의 mixture를 이용하여 GMM으로 모델링 한 분포를 비교할 수 있다. 그림에서 보는 것과 같이 MFCC의 3번째 파라미터는 여러 개의 봉우리가 존재하여 정규 분포로는 모델링이 잘 되지 않지만 GMM을 이용하면 실제 분포와 유사한 확률 분포를 얻어낼 수 있다.

두 개의 분류간의 분리 성능을 측정하기 위해 각 분류마다 특징벡터의 분포를 가우시안 혼합 모델을 사용하여 모델링한다. 추정된 두 분포에서 서로 겹치는 부분이 잘못 구별될 수 있는 부분이 된다. 특징벡터 선택 알고리즘에 적용할 때 이 부분이 0에 가까울수록 두 분류간의 좋은 분리 성능을 가지게 되고 1에 가까울수록 좋지 않은 분리 성능을 가지게 된다. 여기서 두 분포가 겹치는 부분을 오류 영역 (error area)라고 했을 때 GMM 분리 지수 (GMM separation score)를 다음과 같이 정의한다.

$$GMM \text{ separation score} = 1 - (\text{error area}) \quad (4)$$

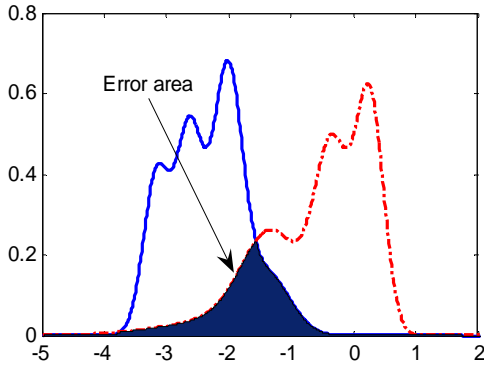


그림 2. 특징벡터의 가우시안 혼합 분포와 오류 영역 (error area)의 예

따라서 GMM 분리 지수가 크면 클수록 두 분류의 분리 성능이 높은 특징벡터가 된다고 할 수 있다. 그림 2에서는 Classic 장르 320곡과 Pop 장르 101곡에서 추출된 5번째 Valley의 평균값의 확률 분포와 오류 영역을 나타내고 있다. 여기서 실선은 Classic 장르에서 추출된 5번째 Valley의 평균값을, 파선은 Pop 장르에서 추출된 5번째 Valley의 평균값을 가우시안 혼합 모델로 모델링 한 것이다. 이와 같은 확률 분포를 가지는 특징벡터의 오류 영역(Error area)은 0.25이므로 따라서 GMM 분리 지수는 0.75가 된다.

2.3 다중 범주 분류에 강인한 특징벡터 선택 알고리즘

두 개의 분류를 구별하는 특징벡터를 선택할 때에는 기존의 Sequential forward selection^[7] 방법을 사용할 수 있다. 하지만 범주의 개수가 3개 이상인 다중 분류를 구별하는 특징벡터를 선택하는 문제로 확장되면 기존의 분리 성능 측정법과 Sequential forward selection 방법으로 최적의 특징벡터를 선택하지 못할 가능성이 존재한다. 왜냐하면 Euclidean distance나 inertia ratio를 이용한 방법은 여러 범주간의 거리를 동등하게 평균을 구하여 분리 성능을 측정하게 되는데 이 때 상대적으로 분리가 잘 되는 범주간의 거리가 특징벡터에 따른 값의 변화가 더 심하기 때문에 분리가 잘 되는 범주의 거리에 영향을 많이 받게 된다. 그러나 실제로는 분리가 잘 되지 않는 범주들을 분리가 더 잘 되게 하는 특징벡터를 선택하는 것이 더욱 성능을 향상시킬 수 있다.

그림 3에서 여러 범주가 존재할 때 기존의 Sequential forward selection 방법을 사용하여 잘못 분류될 수 있는 경우를 가상의 데이터를 통하여 보여 주고 있다. 여기서 범주의 개수가 3개이고 2차원의

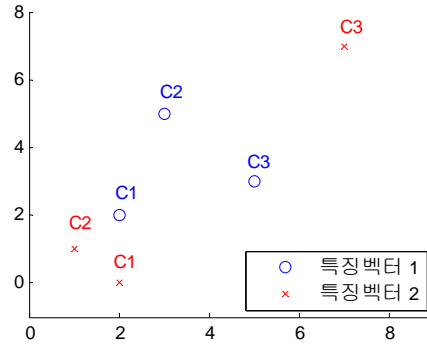


그림 3. 특징 벡터 선택의 오류

특징벡터를 선택하는 경우의 예를 설정하였고, 각 범주의 분산은 같다고 가정하였으며, 그림은 각 분류의 평균값을 나타내었다. 특징벡터 1과 특징벡터 2의 Euclidean distance는 각각 8.23과 18.50이다. 그러나 실제로는 특징벡터 2에서 C1과 C2의 모호성이 크기 때문에 다중 분류 {C1, C2, C3}에 대해서는 특징벡터 1이 더 높은 분리 성능을 나타낼 수 있다. 실제로 실험을 통하여 분리가 잘 되는 범주들을 더 잘 되게 만드는 특징벡터를 선택하는 것보다 분리가 잘 되지 않는 범주들을 잘 될 수 있도록 하는 특징벡터를 선택하는 것이 더 많은 성능 향상을 보여준다는 것을 4장에서 기존의 Sequential forward selection과 본 논문에서 수정하여 적용한 sequential forward selection의 비교 실험을 통하여 확인할 수 있다.

따라서 제안된 알고리즘은 Sequential forward selection 방법을 보완한 방법을 사용한다. 제안된 알고리즘은 기본적으로는 Sequential forward selection 방법과 같이 특징벡터를 하나씩 붙여나가는 방식으로 특징벡터를 선택한다. 그러나 기존의 알고리즘은 다중 범주의 모든 거리를 동등하게 평균을 이용하여 분리 성능을 측정하는 반면, 제안된 알고리즘에서는 이전까지 선택된 특징벡터들이 분리를 잘 하지 못하는 범주들을 기준으로 분리 성능을 측정한다. 즉, 그림 3에서 특징벡터 2가 선택되었다면 C1-C2, C1-C3, C2-C3간의 거리를 모두 고려하지 않고 C1-C2간의 분리 성능을 기준으로 다음 특징벡터를 선택한다. 이러한 방법으로 분리가 잘 되지 않는 범주의 분리 성능을 높임으로써 전체적으로도 향상된 성능을 얻을 수 있다.

제안된 알고리즘은 다음과 같은 과정으로 특징벡터를 선택한다. 먼저 범주 별로 각 특징벡터마다 EM 알고리즘으로 최적화된 가우시안 혼합 확률분포를

구한 후 다음 과정을 통하여 N 차원의 특징벡터로부터 N' 차원의 특징벡터를 선택한다.

- 1) 모델링 된 확률분포를 기반으로 각각의 feature에 대하여 분류간의 겹치는 부분을 측정하여 다음 식과 같이 GMM separation score가 가장 큰 feature를 선택한다. 여기서 f_k 는 특징벡터 선택 알고리즘에 의해 선택된 k번째 feature를, D는 전체 feature set, M은 범주의 개수이다. $S_d(i, j)$ 는 feature d에 대한 범주 i와 j간의 GMM separation score를 나타낸다.

$$f_i = \arg \max_{d \in D} \sum_{i=1}^M \sum_{j=i+1}^M S_d(i, j) \quad (5)$$

- 2) $S(i, j)$ 을 선택된 feature들에 대한 범주 i와 j간의 GMM separation score라고 할 때 모든 범주 ($1 \leq i \leq M, i < j \leq M$)간의 $S(i, j)$ 을 구한다. 이 때 선택된 feature가 2개 이상일 때에는 이전의 $S(i, j)$ 에 새로 선택된 feature의 GMM separation score를 더해 선택된 특징벡터의 거리가 모두 반영되도록 한다. 다음 feature를 선택할 때 가장 성능이 좋지 않은 범주의 분류 성능을 보완하기 위해 다음 식과 같이 $S(i, j)$ 가 최소값을 가지는 범주 i와 j의 GMM separation score를 기준으로 한다. 여기서 $\{c_1, c_2\}$ 는 기준이 되는 범주를 나타낸다.

$$\{c_1, c_2\} = \arg \min_{i, j} S(i, j) \quad (6)$$

- 3) 다음 식과 같이 선택되지 않은 feature들 중 $S_d(c_1, c_2)$ 가 최대가 되는 feature를 선택하여 feature set에 추가한다.

$$f_k = \arg \max_{d \in D} S_d(c_1, c_2) \quad (7)$$

- 4) 과정 2~3을 선택된 feature dimension이 N'가 될 때까지 반복한다.

제안된 알고리즘의 특징은 Gaussian mixture model을 이용하여 특징벡터의 분포를 비교적 정확히 모델링한다는 것이다. 또한 feature dimension을 차례로 선택할 때 가장 거리가 가까운 범주를 기준으로 선택

하면서 구분이 잘 되지 않는 범주들을 보완한다. 제안된 알고리즘을 적용하여 실험한 선택된 특징벡터의 차수에 따른 실험 결과는 4장에서 자세히 다루기로 한다.

III. 특징벡터의 추출

본 장에서는 특징벡터 선택 알고리즘의 성능을 시험하기 위하여 실험에서 사용된 특징벡터에 대하여 알아본다. 특징벡터의 추출은 오디오를 구간별로 분석할 때 오디오 신호의 특징을 나타내도록 간결하게 표현된 파라미터를 계산하는 과정이다. 음악 범주 분류 알고리즘 등 패턴 인식 시스템에서 특징을 잘 나타내어 주는 파라미터를 디자인하는 작업은 가장 중요한 문제라 할 수 있다. 본 절에서는 음악 장르를 나타내기 위해 쓰이는 특징 벡터들에 대해서 살펴보고자 한다.

본 논문에서 사용한 특징벡터들은 크게 음색 표현 특징벡터, 리듬의 특성을 나타내는 특징벡터, pitch의 특징을 나타내는 특징벡터로 구분될 수 있다.

전체적인 특징벡터의 추출은 참고문헌^[2]에서 제시하는 방법을 사용하였다. 자세한 과정은 다음과 같다. 먼저 음색 표현 특징벡터들은 프레임 단위로 추출된다. 이 특징벡터는 20ms의 Hamming window를 인접한 프레임과 50%씩 중복되게 이동하면서 추출하여 프레임 가장자리에서 표현되는 특징의 손실을 최대한 억제하였다. 리듬의 특성을 나타내는 특징벡터와 pitch의 특징을 나타내는 특징벡터들은 곡 단위로 추출된다. 이 특징벡터는 히스토그램 기반의 특징 벡터들로써 한 곡의 리듬과 pitch 정보를 프레임 별로 추출하여 이들의 히스토그램으로부터 곡의 특징을 추출한다는 특징이 있다.

3.1 음색 표현 특징벡터

음색으로 표현되는 특징벡터는 음악-음성 구분 알고리즘과 음성 인식 등의 분야에서 기본적으로 사용되는 특징벡터이다^[9]. 음색을 나타내기 위해서 일반적으로 주파수 영역에서 신호를 해석하게 되는데 주로 단구간 푸리에 분석(STFT) 등을 이용한다. 본 논문에서 내용기반 음악 검색에서 많이 사용하는 특징 벡터를 이용하였으며 특징벡터는 다음과 같다.

- 1) MFCC: MFCC coefficient^[10].
- 2) SR: Spectral rolloff^[2]
- 3) SF: Spectral flux^[2].

- 4) ACC: Autocorrelation coefficient. 여기에서는 1번째 계수부터 12번째 계수까지 사용하였다^[5].
- 5) RMS: RMS energy. 한 프레임 안에서의 오디오 신호의 세기를 표현하는 특징벡터이다.
- 6) MAX: Maxima. 한 프레임 안에서 에너지의 최대 값을 표현하는 특징벡터이다.
- 7) TDZC: Time domain zero crossing^[2,5].
- 8) TMM: Temporal moment. 프레임 안에서 시간 영역에서의 평균, 분산, skewness, kurtosis 값을 특징벡터로 이용하였다^[5].
- 9) ARC: AR coefficient. 여기에서는 첫 2개의 계수를 사용한다^[5].
- 10) SFM: Spectral flatness measure^[5,11].
- 11) SCF: Spectral crest factor^[5,11].
- 12) SS: Spectral slope^[5].
- 13) SD: Spectral decrease^[5].
- 14) VAEELY: Valley^[6]. 여기에서는 20~200Hz, 201~400Hz, 401~800Hz, 801~1600Hz, 1600~3200Hz, 3200~8000Hz의 6개의 옥타브 영역에서 특징벡터를 추출하였다.
- 15) SC: SC^[6].
- 16) OBSI: Octave band signal intensity^[5].
- 17) OBSIR: Octave band signal intensity ratio^[5].
- 18) Nrel: relative specific loudness^[11].
- 19) SHARP: Sharpness^[11].
- 20) SPREAD: Spread^[11].
- 21) HR: Harmonic ratio^[12].
- 22) MHR: Modified harmonic ratio^[12].

3.2 리듬 표현 특징벡터

음악의 범주 중 특히 음악 장르는 리듬에 따라 분류되는 경향이 강한 것을 관찰할 수 있다^[2]. 따라서 안정적인 리듬 추출 알고리즘은 음악의 범주를 구분하는 알고리즘에서 매우 중요한 위치를 차지한다. 본 연구에서는 참고문헌 [2]에서 사용한 방법으로 리듬 표현 파라미터로 리듬 성분을 추출한 후 리듬 히스토그램을 형성한 후 히스토그램의 특징을 나타내는 파라미터를 추출하여 사용하였다.

3초의 윈도우에서 20~200 BPM(Beat per minute)의 리듬을 추출하여 곡 전체에 대한 히스토그램을 만든 후 다음과 같은 특징을 계수로 사용하였다.

- 1) A0, A1: 히스토그램에서 첫 번째와 두 번째 최대치의 상대적 크기를 히스토그램의 전체 합으로 나눈 값.
- 2) RA: 첫 번째 최대치와 두 번째 최대치의 비.
- 3) P1, P2: 첫 번째와 두 번째 최대치의 BPM.
- 4) SUM: 전체 리듬 히스토그램의 합이다.

3.3 피치 표현 특징벡터

피치는 음악을 이루는 중요한 구성 요소이다. 피치 표현 특징벡터도 리듬 표현 특징벡터와 같이 히스토그램으로부터 추출된 특징으로 값을 구성하였다. 오디오 신호에서는 일반적으로 여러 개의 피치가 존재하기 때문에 이에 맞는 알고리즘을 적용해야 한다. 본 논문에서는 다중 피치 검출 알고리즘^[13]을 사용하여 피치를 추출하였다.

이렇게 추출한 피치는 악보에서 사용되는 값으로 바꾸고 곡 전체에 대하여 히스토그램을 구성한 후 다음과 같은 특징을 계수로 사용하였다^[2].

- 1) FA0: 접힌 히스토그램에서 최대치의 크기.
- 2) UPO: 펼쳐진 히스토그램의 최대치의 피치 값.
- 3) FPO: 접힌 히스토그램의 최대치의 피치 값.
- 4) IPO1: 접힌 히스토그램에서 첫 번째와 두 번째 최대치의 피치 값의 간격.
- 5) SUM: 피치의 세기를 나타내는 계수.

3.4 특징벡터의 결합법

본 연구에서 사용된 특징벡터는 크게 프레임 기반의 특징벡터와 히스토그램 기반의 특징벡터로 나눌 수 있다. 프레임 기반의 특징벡터는 2.1절의 음색 표현 특징벡터이고, 히스토그램 기반의 특징벡터는 2.2절의 리듬 표현 특징벡터와 2.3절의 피치 표현 특징벡터이다. 프레임 기반의 특징벡터들은 20ms 윈도우 안에 포함된 오디오 신호에 따라 매우 급격하게 변할 수 있다. 이렇게 변화가 심한 특징벡터가 장르 등을 표현하는 특징으로써 의미가 있기 위해서 여러 프레임 동안 특징벡터의 통계적 특성을 추출하여 고차원의 특징벡터를 구성하였다. 이를 위해 여러 프레임이 모인 구간을 texture window로 정의한다.

표 1. 특징벡터의 구성

Indx	Frame / Histogram	Order
MFCC	프레임	12×2=24
SR	프레임	1×2=2
SF	프레임	1×2=2
ACC	프레임	12×2=24
RMS	프레임	1×2=2
MAX	프레임	1×2=2
TDZC	프레임	1×2=2
TMM	프레임	4×2=8
ARC	프레임	2×2=4
SFM	프레임	5×2=10
SCF	프레임	5×2=10
SS	프레임	1×2=2
SD	프레임	1×2=2
VALLEY	프레임	6×2=12
SC	프레임	6×2=12
OBSI	프레임	6×2=12
OBSIR	프레임	5×2=10
Nrel	프레임	37×2=74
SHARP	프레임	1×2=2
SPREAD	프레임	1×2=2
HR	프레임	1×2=2
MHR	프레임	1×2=2
RHYTHM	히스토그램	6
PITCH	히스토그램	6

본 연구에서는 texture window를 1초에 해당하는 100개의 프레임으로 두고 평균과 분산을 구하였다. 따라서 각 파라미터마다 2개의 고차원 파라미터가 추출된다. 예를 들어 2분 분량 음악의 경우 최초 30초와 마지막 30초를 제외한 중간부분 60초 동안 10ms 간격으로 특징벡터가 추출되며 이를 이용하여 다시 100개의 프레임 동안 모아 한 곡당 60개의 특징벡터가 추출되도록 하였다.

추출된 특징벡터들은 성능을 높이기 위해 하나의 벡터로 결합하게 된다. 이 때 프레임 기반 특징벡터와 히스토그램 기반 특징벡터는 서로 다른 시간단위에서 추출되었기 때문에 이를 맞춰 주어야 한다. 본 연구에서는 한 곡에 대하여 하나의 특징벡터가 나오도록 설계하였다. 프레임 기반 특징벡터는 1곡 전체에 대하여 하나의 특징벡터로 표현되어야 한다. 이를 위해 한 곡 전체에 대하여 texture window를 통해 구한 고차원 특징벡터의 평균을 구하여 하나의 특징벡터를 추출하였다.

이와 같은 방법으로 추출된 특징벡터는 표 1과 같이 구성하였다.

IV. 실험 및 결과 고찰

제안된 알고리즘의 실험을 위해 사용된 음악 DB는 ISMIR2004 Audio Description Contest에서 장르 구분 대회를 위해 제공된 음악을 사용하였다^[6]. 이 데이터베이스는 classical, electronic, jazz_blues, metal_punk, rock_pop, world의 6개의 범주로 구성되어 있으며 training set과 development set이 각각 729곡으로 구성되어 있다. Training set는 classical이 320곡, electronic이 115곡, jazz_blues가 26곡, metal_punk가 45곡, rock_pop이 101곡, world가 122곡으로 이루어져 있다. 오디오 파일들은 44100Hz, 16-bit, 모노 형식으로 저장되었다. 특징벡터를 선택할 때와 모델을 만들 때에는 training set을 사용하였고 테스트를 할 때에는 development set을 사용하였다.

특징벡터를 추출할 때에는 training set과 development set의 모든 곡에서 최종적으로 하나의 특징벡터를 추출하였다. 음악의 최초 30초와 마지막 30초는 샘플링, 묵음 구간 등 음악 장르를 잘 나타내지 못하는 특징이 포함될 수 있으므로 특징 추출에서 배제하였다. 프레임 기반의 특징벡터는 texture window마다 추출된 고차원의 특징벡터를 한 곡 전체에 대하여 평균을 구하여 히스토그램 기반의 특징벡터와 결합하였다.

특징벡터를 선택할 때에는 각각의 특징벡터들이 다른 도메인에 존재하기 때문에 거리를 측정할 때 특징벡터의 단위의 영향을 제거하기 위해 각 특징벡터의 평균과 분산에 대하여 정규화하는 과정을 먼저 수행하였다. 또한 거리 측정 방식 중 모든 표본을 직접 사용하는 방식은 거리를 측정할 때 평균과 멀리 떨어진 표본의 영향을 더 많이 받게 된다. 따라서 표준편차의 3배 (정규 분포로 가정할 경우 평균에서 가장 먼 1%)가 넘는 표본은 제외하였다. 특징벡터의 차수에 따라 1차부터 100차까지의 특징벡터를 구성하여 실험하였다.

범주 분류의 성능을 알아보기 위해 분리 성능을 측정할 때 사용한 GMM classifier와 classifier로 널리 쓰이고 있는 k-NN classifier를 사용하였다. 여기서 k=3으로 설정하였고 729개의 훈련 데이터를 모델로 development set의 전체 곡에 대하여 테스트하였다.

성능 평가는 전체 샘플 중 범주를 제대로 찾는 샘플의 비인 정확도(Accuracy)로 평가하였다.

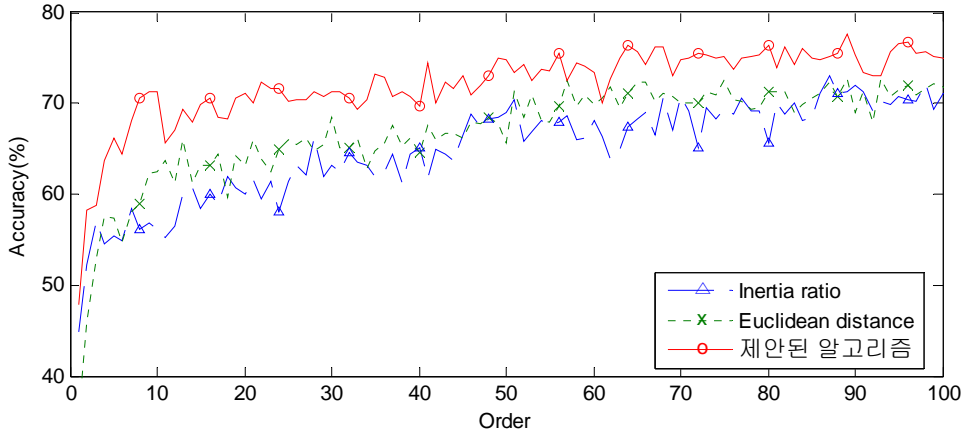


그림 4. GMM classifier에서 특징벡터의 차수에 따른 범주 분류의 정확도

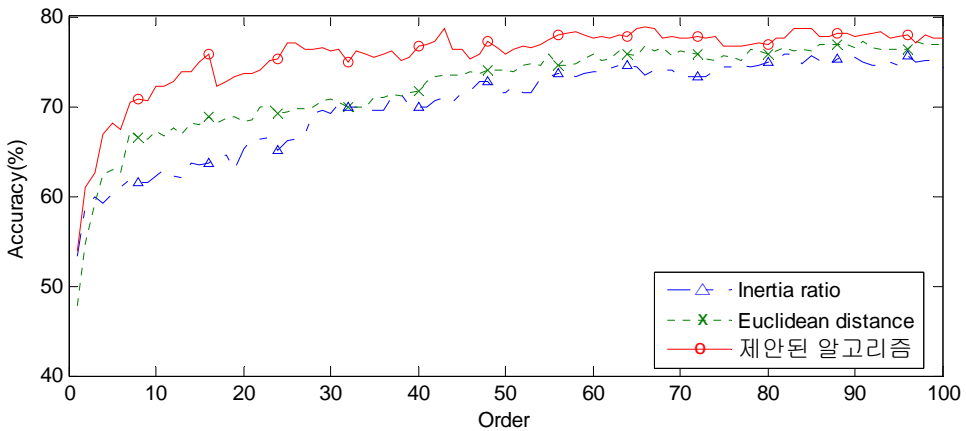


그림 5. k-NN classifier에서 특징벡터의 차수에 따른 범주 분류의 정확도

표 2. 분류기와 특징벡터 선택 알고리즘에 따른 범주 분류의 정확도

GMM classifier			
방법 / 차수	1~20	21~70	71~100
Inertia ratio	57.0%	65.3%	69.8%
Euclidean distance	58.9%	67.7%	70.9%
제안된 알고리즘	66.5%	72.5%	75.2%
KNN classifier			
방법 / 차수	1~20	21~70	71~100
Inertia ratio	61.7%	71.1%	74.7%
Euclidean distance	64.9%	72.9%	76.3%
제안된 알고리즘	69.9%	76.7%	77.7%

그림 4와 그림 5는 특징벡터 선택 알고리즘에 따른 GMM classifier와 k-NN classifier를 통해 실험한 결과이다. 성능은 알고리즘과 특징벡터의 차수에 따라 평가하였다. 특징벡터의 차수가 늘어남에 따라 정확도가 늘어나는 추세로 나타났지만 데이터의 통계적 특성이 달라졌기 때문에 단조 증가하지 않는 경우도 있었다.

실험 결과에서 알 수 있듯이 낮은 차수에서 다중 범주에 대하여 분류할 때 제안된 알고리즘이 가장 우수한 성능을 나타내었다. 특히 20차 미만의 낮은 차수에 대해서는 5~10%까지의 높은 성능을 나타내었다. 낮은 차수의 특징벡터에서는 분리가 잘 되지 않는 분류의 특징벡터가 성능에 더 많은 영향을 미치는 것을 확인할 수 있다. 또한 높은 차수에 대해서 2~6%의 성능 향상을 보여주었다.

표 2는 범주 분류의 에러 비율을 특징벡터 선택 알고리즘에 따라 나타낸 것이다. 전체 차수에 대한 평균 성능은 기존의 알고리즘보다 제안된 알고리즘이 2%에서 10% 정도 높은 성능을 나타내고 있고, 낮은 차원이라고 할 수 있는 1차부터 20차까지의 평균 성능은 5%에서 10% 정도 높은 성능을 나타내고 있다.

특징벡터의 차원이 낮을 때 성능 차이가 두드러진 데 비해 특징벡터의 차원이 높을 때 성능이 상대적으로 낮은 원인을 다음과 같이 분석해 볼 수 있다. 먼저 가우시안 혼합 모델을 이용하여 분리 성능을 측정할 때 overlap area는 다차원이 아닌 하나의 차원에서 구하여 평균을 내는 방식을 사용하고 있다. 따라서 차원이 높아짐에 따라 발생할 수 있는 파라미터 간의 중복 등의 문제에 대해서는 취약한 점이 있다. 다음으로 다중 분류를 구분하는 특징벡터를 선택하는 알고리즘에서 구분이 잘 되지 않는 분류들의 거리만을 기준으로 하는 것은 특징벡터의 차원이 높아짐에 따라 특징벡터의 조합에 의해 그 영향이 적어질 수 있다. 이 경우 다른 특징벡터의 조합이 전체 분류 성능을 높이는 결과가 발생할 수 있다.

본 논문에서 제안한 알고리즘은 특히 저전력 저연산을 요구하는 응용 프로그램이나 실시간 음악 범주 분류 시스템 등에 적용될 수 있을 것이다. 특징벡터의 선택과 분류기의 학습은 분류기의 적용 분야에 따라 시스템을 설계할 때 한 번만 수행하는 것을 고려할 때 같은 종류의 classifier를 사용한다면 시스템은 입력 데이터의 특징벡터를 추출하는 연산이 대부분이므로 시스템의 연산량은 특징벡터의 차수에 비례한다고 할 수 있다. 제안된 알고리즘으로 실험한 경우 20차 미만에서 얻을 수 있는 성능을 기존의 알고리즘으로는 40차에서 50차 정도의 특징벡터를 사용해야 얻을 수 있는 부분에 주목해 볼 필요가 있다. 이와 같은 결과는 비슷한 성능의 분류 시스템을 1/2에서 1/4만의 연산량만으로 구현할 수 있는 시스템을 설계할 수 있다는 것이다.

V. 결론

본 논문에서는 GMM을 이용한 새로운 특징벡터 선택 알고리즘을 제안하고 GMM classifier와 k-NN classifier를 사용하여 기존의 알고리즘과 성능 비교 실험을 수행하였다. 제안된 알고리즘은 Gaussian mixture model을 이용하여 특징벡터의 분포를 비교적 정확히 모델링하였고 따라서 실제 분포에 따라 구분

이 잘 되는 특징벡터를 선택할 수 있었다. 또한 다중 범주에 관한 특징벡터를 선택할 때 구분이 잘 되지 않는 범주간의 거리를 기준으로 설정함으로써 다중 범주에 대한 성능을 향상시킬 수 있었다.

표 2에서 볼 수 있듯이 제안된 알고리즘은 기존 알고리즘에 비하여 2%에서 10% 정도 높은 성능을 나타내었고 특징벡터의 차수가 20차 미만인 실험에서 특히 좋은 성능을 나타내는 것을 확인할 수 있었다. 제안된 알고리즘으로 저연산 및 실시간 시스템을 구현할 수 있는 20차 미만의 적은 특징벡터를 선택하는 경우 더 적은 차수로 기존의 알고리즘으로 40차에서 50차의 특징벡터를 추출하였을 때와 비슷한 성능을 보임을 확인할 수 있었다.

본 논문에서 제안된 알고리즘 기반으로 향후 다양한 classifier에 대하여 실험하여 classifier에 관계없이 좋은 성능을 보이는 알고리즘에 대한 연구를 계속할 것이다. 이를 위해 실험에서 발견되었던 가우시안 혼합 모델의 overlap area를 다차원에서 구하는 문제와 다중 분류를 구분하는 알고리즘을 개선해야 할 것이다. 또한 기존의 Euclidean distance나 확률분포를 기반으로 한 Divergence distance, Bhattacharyya distance와 같은 거리 측정 방법과 결합하여 특징벡터 선택 알고리즘의 성능을 강화시키는 방안에 관한 연구를 계속할 것이다.

참고 문헌

- [1] E. Scheirer, M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, pp. 1331-1334, 1997
- [2] G. Tzanetakis, P. Cook, "Musical Genre Classification of audio signals", *IEEE Transaction on Speech and Audio Processing*, vol. 10, No. 5, pp. 293-302, 2002
- [3] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search and retrieval of audio", *IEEE Multimedia*, vol. 3, No. 3, pp. 27-36, 1996.
- [4] G. Peeters, "Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization," *Proc. 115th AES Convention*, New York, Oct. 2003.
- [5] S. Essid, G. Richard, B. David, "Instrument rec-

ognition in polyphonic music based on automatic taxonomies,” *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, No. 1, pp. 68-80, Jan. 2006

[6] D.-N. Jiang, L. Lu, H.-J. Zhang, J.-H. Tao, and L.-H. Cai. “Music type classification by spectral contrast feature,” In Proceedings of IEEE International Conference on Multimedia and Expo (ICME02), Lausanne Switzerland, Aug 2002.

[7] S. Theodoridis, K. Koutroumbas, “Pattern recognition (third edition),” *Academic Press*, 2006

[8] F. J. Ferri, P. Pudil, M. Hatef, J. Kittler, “Comparative study of techniques for large-scale feature selection,” Gelsema, E.S., Kanal, L.N. (Eds.), *Pattern Recognition in Practice* vo. IV, pp. 403-413, 1994.

[9] E. Scheirer and M. Slaney, “Construction and evaluation of a robust multifeature speech/music discriminator,” *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, pp. 1331- 1334, 1997.

[10] Beth Logan, “Mel Frequency Cepstral Coefficients for music modeling,” *Proceedings of the First International Symposium on Music Information Retrieval (ISMIR)*, 2000.

[11] G. Peeters, “A large set of audio features for sound description (similarity and classification) in the CUIDADO project,” *CUIDADO I.S.T. Project Report*, 2004.

[12] J. J. Burred, A. Lerch, “A hierarchical approach to automatic musical genre classification,” *Proc. 6th Int. Conference on Digital Audio Effects*, London, UK, September 2003.

[13] T. Tolenen and M. Karjalainen, “A computationally efficient multipitch analysis model,” *IEEE Trans. Speech Audio Processing*, vol. 8, pp.708-716, Nov. 2000.

[14] T. Hastie, R. Tibshirani, J. Friedman, “The elements of statistical learning - data mining, inference, and prediction,” *Springer*, 2000

[15] D. A. Reynolds, R. C. Rose, “Robust text-independent speaker identification using Gaussian mixture speaker models,” *IEEE Transaction on Speech and Audio Processing*, vol. 3, No. 1, pp. 72-83, January 1995.

[16] http://ismir2004.ismir.net/genre_contest/index.htm, 2004.

문 선 국(Sun-Kuk Moon)

준회원



2006년 2월 연세대학교 전기전
자공학과(학사)
~2006년 3월현재 연세대학교 전
기전자공학과(석사과정)
<관심분야> 오디오 신호처리,
3D오디오, 내용기반 음악인식

최 택 성(Tack-Sung Choi)

준회원



2003년 2월 중앙대학교 전기전
자공학과(학사)
2005년 2월 연세대학교 전기전
자공학과(석사)
2005년 9월~현재 : 연세대학교
전기전자공학과(박사과정)
<관심분야> 오디오 신호처리,
3D오디오, 내용기반 음악인식

박 영 철(Young-Cheol Park)

정회원

한국통신학회 논문지 제29권 6C호 참고

윤 대 희(Dae Hee Youn)

종신회원

한국통신학회 논문지 제29권 6C호 참고