한국음향학회지 제38권 제3호 pp. 314~319 (2019)
The Journal of the Acoustical Society of Korea Vol.38, No.3 (2019)
https://doi.org/10.7776/ASK.2019.38.3.314

pISSN : 1225-4428
eISSN : 2287-3775

커버곡 검색을 위한 코드 기반 크로마그램 유사도

A code-based chromagram similarity for cover song identification

서진수[1†]

(Jin Soo Seo[1†])

[1]강릉원주대학교 전자공학과

(Received February 8, 2019; accepted May 24, 2019)

초 록: 음악 커버곡 검색 시스템 구현에 있어서 크로마그램 간 유사도 계산은 필수적인 구성 요소이다. 본 논문은 크로마그램 비교에 소요되는 저장공간 및 계산량을 줄이기 위한 크로마그램 코딩 방법을 제안한다. 음악별로 코드북을 학습하여 크로마그램 수열을 코드 수열로 변환하여 저장 공간을 줄이게 된다. 얻어진 코드 간 거리를 룩업 테이블에 저장하여 크로마그램 비교의 속도를 개선하였다. 두 가지 커버곡 실험 데이터셋에서 성능 비교를 수행하여, 제안된 코드 기반 방법과 기존 방법 간의 커버곡 검색 정확도, 저장 공간, 계산량을 비교하였다.
핵심용어: 음악 검색, 음악 유사도, 크로마 특징, 커버곡 판별, 룩업 테이블

ABSTRACT: Computing chromagram similarity is indispensable in constructing cover song identification system. This paper proposes a code-based chromagram similarity to reduce the computational and the storage costs for cover song identification. By learning a song-specific codebook, a chromagram sequence is converted into a code sequence, which results in the reduction of the feature storage cost. We build a lookup table over the learned codebooks to compute chromagram similarity efficiently. Experiments on two music datasets were performed to compare the proposed code-based similarity with the conventional one in terms of cover song search accuracy, feature storage, and computational cost.
Keywords: Music retrieval, Music similarity, Chromagram, Cover song identification, Lookup table

PACS numbers: 43.75.Zz, 43.60.Uv

I. 서 론

방대한 양의 디지털 음원을 대상으로 음악 추천 및 검색, 저작권 보호 등을 빠르고 신뢰성 있게 제공해 줄 수 있는 오디오 정보 처리 기술의 중요성이 증대되고 있다.[1-3] 다양한 음악 정보 처리 기술들 중에서 본 논문은 라이브 버전, 편집, 리메이크 등을 통해서 재녹음된 음악인 커버곡을 검색하는 방법에 관해서 다룬다. 커버곡 검색 기술은 웹하드 및 유튜브 등 데이터 공유 서비스에서 저작권 보호 및 음악 아카이브 정리 등에 활용될 수 있을 것으로 기대된다. 콘서트 실황, 거리 공연 등을 카메라로 촬영하여 유튜브 등을 통해서 공유하고 광고 등으로 수익을 얻는 경우가 있다. 이런 경우도 저작권을 침해한 것인데, 기존의 핑거프린팅 기술로는 라이브 음악의 저작권 침해를 적발할 수 없지만 커버곡 검색으로는 가능하다.

커버곡 검색을 위한 음악 유사도 비교를 위해서는 원곡과 커버곡 간의 공유되는 특성을 찾아야 한다. 하지만 커버곡을 만드는 과정인 라이브 버전, 편집, 리메이크를 거치면서, 가수와 악기의 차이로 인한 음색 변화, 연주 속도 및 스타일 차이로 인한 템포 및 리듬, 음악 키의 변조 등 원곡과 커버곡 간의 신호적 차이가 다양하다.[4,5] 이런 다양한 종류의 변형이 존재하므로 커버곡 검색은 여전히 어려운 문제로 남아 있다. 현재 연구들에서 가장 널리 사용되고 있는 원

†**Corresponding author:** Jin Soo Seo (jsseo@gwnu.ac.kr)
Department of Electronic Engineering, Gangneung-Wonju National University, 7 Jukhun-gil, Gangneung, Gangwon-Do 25457, Repubic of Korea.
(Tel: 82-33-640-2428, Fax: 82-33-656-0740)

곡과 커버곡 간의 공통점은 음들의 시간적 연결을 의미하는 선율(멜로디)이다. 선율을 표현하는 신호처리적인 특징에는 대표적으로 크로마그램이 있다. 크로마그램은 신호의 음색 특성과 음의 크기와는 무관하므로 커버곡 검색에 도움이 된다. 크로마그램은 음악 신호를 짧은 길이(수 십~수 백 ms)의 프레임 단위로 나누고 각 프레임에서 스펙트럼을 구하고 옥타브 차이나는 음악 스펙트럼 성분들을 가산하여 음악의 전체 스펙트럼 성분들을 하나의 옥타브 안으로 접어서 표현한 것이다. 즉, 크로마그램은 인간 청각 시스템은 옥타브 차이가 나는 두 음을 유사하게 지각하는 것에 기반한다. 일반적으로 크로마그램은 서양 음악의 12음계에 따라서 하나의 옥타브 내에서 주파수 대역을 나누어 각 음계의 스펙트럼 성분들을 합산하므로 프레임별로 12차 벡터 형태로 주어진다. 크로마그램은 스펙트럼 성분들을 옥타브 단위로 합산하는 과정에서 가수의 목소리와 악기에 연관된 음색 보다는 음악 자체의 화성적 구조에 보다 밀접하게 연관되므로 커버곡 검색에 적합한 특징이다. 크로마그램 추출에 관한 상세한 설명은 Reference [6]에 있다.

본 논문에서는 커버곡 검색 방법들 중에서 음악 신호로부터 얻어지는 크로마그램 수열을 직접 비교하여 시간축 상에서 정합하는 수열 직접 비교 방법에 대해서 다룬다.[4,7,8] 이런 수열 직접 비교 방법은 음성 인식과 DNA 수열 분석에서 사용해왔던 dynamic time warping이나 SW(Smith-Waterman) 알고리즘[9] 등을 활용한다. 수열 직접 비교 방법들은 커버곡 검색에서 우수한 성능을 보였지만, 수열 비교에 많은 계산량이 요구되고 전곡의 크로마그램을 모두 저장해야하므로 저장 공간이 많이 필요하다. 본 논문에서는 계산량과 저장 공간을 줄이기 위해서 크로마그램 코드북을 학습하는 방법을 제안한다. 코드북을 이용하여 특징을 변환하여 검색 및 분류에 활용하는 것은 컴퓨터 비전에서 널리 사용되어 왔다.[10,11] 본 논문에서는 코드북을 이용하여 음악 신호의 크로마그램 수열을 코드 수열로 변환하여 저장 공간을 줄인다. 따라서 크로마그램 수열 비교는 코드 수열 비교 문제로 변환되게 된다. 이때 코드간의 거리를 미리 계산하여 LUT(Lookup Table)에 저장하면 코드 수열 시간축 정합을 효율적으로 수행하여 계산량을 크게 줄일 수 있다. 또한 코드북 사이즈를 조정할 수 있으므로, 커버곡 검색 정확도와 계산량 및 저장 공간 등의 비용 간의 균형점을 커버곡 검색 서비스의 요구조건에 맞춰서 조정할 수 있다.

본 논문은 코드북 기반 크로마그램 수열 정합을 통한 커버곡 검색에 관한 연구이다. II장에서 크로마그램 수열 정합을 위한 SW 알고리즘을 살펴보고, 코드북을 통한 성능 개선 방안을 제안한다. III장에서 제안된 방법의 성능을 실험하고 결과를 비교 분석한다.

II. 코드 기반 크로마그램 유사도

크로마그램 수열 직접 비교 방법 중 SW 알고리즘을 이용한 방법을 살펴보고, 코드북 학습을 통해서 계산량 및 특징 저장 공간을 줄이는 방법을 제안한다.

2.1 기존 SW 알고리즘 기반 커버곡 검색

수열 직접 비교 방법들 중에서 MIREX 커버곡[1] 검색 경연에서 가장 좋은 성능을 보인 방법은 OTI (Optimal Transposition Index)를 이용하여 크로마그램 벡터간 유사도를 판단하고, SW 알고리즘을 통해서 수열 정합을 수행한 OTI-SW 방법이다.[4] Fig. 1은 OTI-SW 방법의 블록선도이다. OTI-SW 방법은 크로마그램 수열 추출, 유사도 행렬 계산, SW 알고리즘을 통한 수열 정합의 세 과정으로 이루어진다. 각 음악으로부터 크로마그램 수열을 추출하고, 얻어진 수열 간에 OTI를 기준으로 이진 유사도 행렬을 계산한다. 얻어진 이진 유사도 행렬에 SW 알고리즘을 적용하

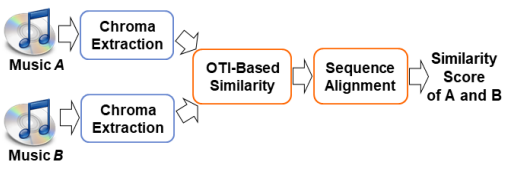

Fig. 1. The music-similarity computation for the cover song identification based on the optimal transposition index and sequence alignment.

1) 커버곡 경연: https://www.music-ir.org/mirex/wiki/ 2007: Audio Cover Song Identication Results

여 가장 유사도가 높은 부분 수열을 찾아서 두 곡간의 유사도 값으로 사용한다. 유사도 값의 역수를 두 음악의 길이로 정규화하여 두 음악 간의 최종 거리 값을 구한다.

OTI-SW 방법을 상세하게 살펴보면 다음과 같다. 커버곡 생성과정에서 음악의 조변화가 자주 발생하며, 음악의 조변화는 크로마그램 상에서 크로마축으로 원형이동으로 나타나게 된다. OTI는 이러한 음악의 조변화에 대해서 불변성을 얻기 위해서 제안되었다. Fig. 1의 두 음악의 프레임 개수가 각각 M과 N일 때, 크로마그램 수열을 얻어진 시간 순으로 각각 A, B로 표기하면, 다음과 같이 a_m과 b_n은 L차 크로마그램 벡터가 된다(일반적으로 L = 12).

$$A = \{a_1, a_2, \cdots, a_M\}, \\ B = \{b_1, b_2, \cdots, b_N\}. \tag{1}$$

음악이 조변화되면 크로마그램 벡터는 순환 이동하게 된다. 벡터 a_m이 주어졌을 때, 벡터간 유사도가 최대가 되도록 b_n이 순환 이동해야하는 빈의 개수인 $OTI_{m,n}$은 벡터간 내적<,>으로 다음과 같이 정의된다.

$$OTI_{m,n} = \underset{l}{\arg\max} < a_m, cshift(b_n, l) >. \tag{2}$$

Eq. (2)에서 $cshift(b_n, l)$는 b_n 벡터를 l만큼 순환 이동한 벡터를 가리키며, l을 0에서 L-1까지 가변하면서 최대가 되는 l값을 찾는다. 전곡 평균 크로마그램 벡터를 각각 g_A와 g_B라고 하면, OTI_g는 다음과 같이 정의된다.

$$OTI_g = \underset{l}{\arg\max} < g_A, cshift(g_B, l) >. \tag{3}$$

$OTI_{m,n}$과 OTI_g를 비교하여, A와 B의 프레임간 이진 유사도 행렬 S를 다음과 같이 얻는다.

$$S_{m,n} = \begin{cases} \mu_+ & \text{if } OTI_{m,n} = OTI_g \\ \mu_- & otherwise \end{cases}. \tag{4}$$

Eq. (4)에서 μ_+와 μ_-는 각각 정합과 부정합을 나타내는 상수로 Reference [4]에 나온 바와 같이 $\mu_+ = 1, \mu_- = -0.9$를 사용하였다.

커버곡을 만드는 과정에서 음악 연주 속도인 템포에 변화를 주는 경우가 자주 있다. 이런 템포 변화에 대한 강인성을 얻기 위해서 SW 알고리즘을 적용하여 가장 유사도가 높은 부분 수열을 찾는다. 프레임간 이진 유사도 행렬 S의 시간축 방향 연속성을 고려하여, 다음과 같이 시간축 정합 행렬 H를 구한다.

$$H_{m,n} = \max \begin{cases} H_{m-1,n-1} + S_p - \delta(S_{m-2,n-2}, S_p) \\ H_{m-2,n-1} + S_p - \delta(S_{m-3,n-2}, S_p) \\ H_{m-1,n-2} + S_p - \delta(S_{m-2,n-3}, S_p) \\ 0 \end{cases}. \tag{5}$$

Eq. (5)에서 $S_p = S_{m-l,n-l}$이며, 정합 제약 조건 함수 $\delta(\,)$는 다음과 같이 주어진다.

$$\delta(p,q) = \begin{cases} 0 & \text{if } q > 0 \\ 0.5 & \text{if } q \le 0, \text{and } p > 0 \\ 0.7 & \text{if } q \le 0, \text{and} p \le q \end{cases}. \tag{6}$$

정합값 $H_{m,n}$은 A수열의 m, B 수열의 n번 위치까지의 최대 시간축 정합값을 의미한다. 커버곡 검색을 위해서 수열 A와 B 간의 거리는 H의 최대값의 역수와 두 수열의 길이에 따라 다음과 같이 구한다.

$$d(A,B) = \frac{M+N}{\max(H)}. \tag{7}$$

2.2 제안한 코드 기반 크로마그램 유사도

2.1에서 살펴본 기존 OTI-SW 방법[4]이 우수한 커버곡 검색 성능을 보이지만, OTI를 이용한 이진 유사도 행렬을 구하는 과정에서 계산량이 많이 필요하고, 전체 크로마그램 수열을 저장하는데 큰 저장 공간이 요구된다. 두 가지 문제점을 해결하는 방안으로 크로마그램 벡터 수열을 코드 수열로 변환하고, LUT를 이용한 거리 비교 방법을 제안한다. Eq. (1)의 크로마그램 수열 A에 대해서 K개의 L차원 벡터로 이루어지는 코드북 $C_A = \{c_{Ak}\}$를 다음의 벡터간 내적의 합이 최대가 되도록 유도한다.

$$O(C_A) = \max \sum_{m=1}^{M} < a_m, c_{A W_{Am}} > \qquad (8)$$
$$where\ W_{Am} = \underset{1 \le k \le K}{\arg\max} < a_m, c_{Ak} > .$$

Eq. (8)을 최대화하는 코드북을 구하는 것은 어려우므로 *k*-means 알고리즘과 같이 임의로 초기 코드북을 생성하고 반복 학습을 통하여 코드북을 업데이트 하였다. 더 이상 $O(C_A)$가 개선되지 않으면 코드북 업데이트를 중지하고 반복 학습을 멈추게 된다. 이러한 *k*-means 알고리즘을 이용한 코드북 학습은 영상처리[10,11] 및 음악 검색[3]에 널리 적용되었다.

Eq. (1)의 두 크로마그램 수열 *A*와 *B*로부터 각각 얻은 코드북 C_A 와 C_B 를 각각 이용하여, Fig. 2에 나온 바와 같이 크로마그램 벡터 수열 *A*와 *B* 를 코드 수열

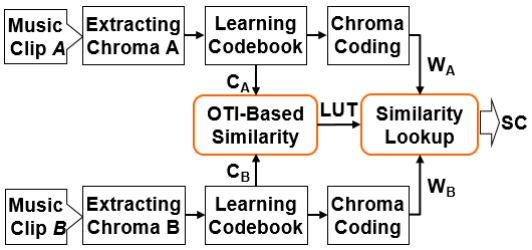

Fig. 2. Computation of the pairwise similarity matrix SC by using the lookup table obtained from the learned codebooks.

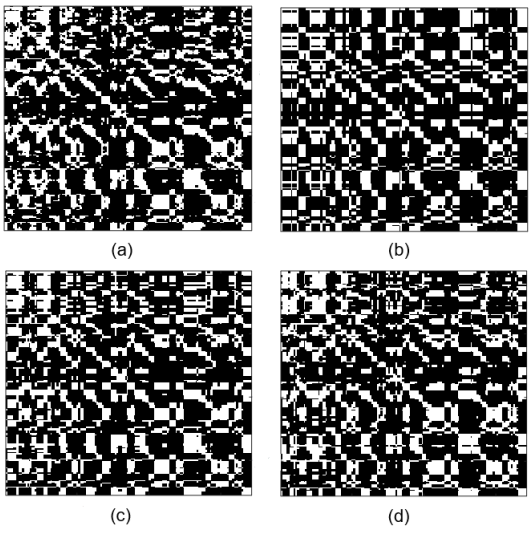

Fig. 3. The pairwise similarity matrix between the original song (“More than words”) and its cover version. (a) S from OTI. (b) SC with K = 16. (c) SC with K = 48. (d) SC with K = 80.

$\boldsymbol{W_A} = \{W_{Am}\}$와 $\boldsymbol{W_B} = \{W_{Bn}\}$ 으로 변환한다. 코드북 C_A 와 C_B 의 코드들 간의 OTI인 *K*행 *K*열 *OTIC* 행렬은 다음과 같이 주어진다.

$$OTIC_{i,j} = \underset{l}{\arg\max} < c_{Ai}, cshift(c_{Bj}, l) > . \qquad (9)$$

코드간의 OTI 거리인 *LUT*는 다음과 같이 구한다.

$$LUT(i,j) = \begin{cases} \mu_+ & \text{if } OTIC_{i,j} = OTI_g \\ \mu_- & otherwise \end{cases} . \qquad (10)$$

위 *LUT*를 이용하여 크로마그램 수열 *A*와 *B* 의 크로마그램 벡터 간의 유사도 행렬 *SC*를 직접 계산하지 않고 *LUT*를 참고하여 다음과 같이 구할 수 있다.

$$SC_{m,n} = LUT(W_{Am}, W_{Bn}) . \qquad (11)$$

Eq. (4)와 같이 직접 크로마그램 벡터들 간의 OTI를 계산하여 유사도 행렬을 구할 경우 총계산량은 *LMN* 번의 내적이지만, 코드 기반 OTI로부터 구한 *LUT*를 참조하여 유사도 행렬을 구하면 계산량을 LK^2 번의 내적으로 줄일 수 있다. 크로마그램 수열을 직접 사용할 경우 $L(N+M)$개의 실수를 저장해야하지만, 코드 기반 방법의 경우 $(N+M)\log_2(K)$ 비트로 줄일 수 있다. 얻어진 이진 유사도 행렬 *SC*에 기존 방법과 같이 SW 알고리즘을 적용하여 Eq. (5)의 *H* 행렬을 구하고 Eq. (7)과 같이 두 음악간 거리 값을 구한다. Fig. 3은 코드를 사용하지않고 구한 이진 유사도 행렬인 *S*와 코드 기반 유사도 행렬인 *SC*를 비교하기 위해서 코드북 사이즈 *K*값을 바꿔 가면서 얻은 행렬을 도시한 것이다. 밝은 값을 가진 부분이 OTI가 일치한 부분이고 어두운 부분은 OTI가 일치하지 않은 부분이다. 코드북 사이즈 *K*값이 커짐에 따라 *SC*와 *S*의 차이가 줄어드는 것을 확인할 수 있으며, *K*값이 크지 않더라도 *S*와 *SC*의 차이가 크지 않음을 알 수 있다. 특히 Fig. 3의 예시에서 원곡과 커버곡 간 대응되는 정합이 있는 *S*와 *SC*의 대각행렬 부분은 *K*값이 크지 않더라도 거의 같음을 확인할 수 있다.

III. 실험 결과

본 장에서는 제안한 코드 기반 OTI 유사도와 기존 OTI 유사도의 커버곡 검색 성능을 비교하였다. 커버곡 성능 비교를 위해서 음원 및 성능이 공개되어 있는 covers80 데이터셋과 자체적으로 수집한 covers330 데이터셋을 사용하였다. 미국 콜롬비아 대학에서 커버곡 실험을 위해서 수집된 covers80 데이터셋은 원본곡과 커버곡 쌍 80개로 이루어진 것으로 모두 160곡으로 구성되어있다.[12,13] 자체적으로 수집한 covers330 데이터셋은 1000곡으로 이루어져 있으며, 330곡은 커버곡 실험용도로 30개의 원곡과 각 원곡당 10가지 커버곡 버전으로 구성된다. 나머지 670곡은 검색 성능을 평가하기 위해서 사칭자(imposter)로 삽입되었다. 커버곡 검색 성능 지표로 covers80 데이터셋에서는 각 커버곡을 80곡의 원곡 데이터셋과 비교하여 가장 거리가 가까운 것이 입력 커버곡의 원곡이 맞을 경우의 확률인 P@1과 MAP(Mean of Average Precision)를 구하였다. covers330 데이터셋에 대해서는 커버곡 실험용 330곡 각각을 1000곡 데이터셋 전체에 대해서 비교를 수행하여 자신을 제외하고 거리가 가까운 10곡 중 커버곡이 맞는 곡의 개수인 $MNCI_{10}$과 MAP를 구하였다.

실험대상 음악들로부터 크로마 수열을 얻기위해서 메트랩 기반의 Chroma Toolbox[6]를 사용하였으며, 크로마를 얻는 방법 중 음색에 대한 불변성을 개선한 CRP(Chroma DCT-Reduced log Pitch)를 크로마그램으로 사용하였다.[14] 실험에 사용되는 음악 파일들을 모노로 바꾸고 22050 Hz로 샘플링 주파수를 맞춘 후, 4410길이의 윈도우를 50 %씩 겹쳐 가면서 프레임의 피치를 구해서 옥타브 단위로 나누고 각 옥타브에서 12개의 크로마에 해당하는 값들을 구한다. 옥타브별로 얻어진 크로마 값을 다 더하면 최종적으로 12차수의 크로마그램 벡터가 얻어진다. 얻어진 크로마그램 수열을 리샘플링해서 0.5 s당 1개씩의 12차 크로마그램벡터가 나오도록 했다. 코드북 학습 시 *k*-means 방법의 특성으로 초기값에 따라 다른 코드가 얻어질 수 있으므로, 제안된 코드 기반 방법의 커버곡 검색 성능은 20번 반복 수행한 평균치이다.

음원데이터셋 covers80과 covers330에서 코드북 사이즈 K를 가변하면서 실험을 수행하여 구한 커버곡 검색 성능을 Tables 1과 2에 정리하였다. 검색 성능을 비교하기 위해서 상대 정확도(Relative Precision, RP)를 사용하였다. 상대 정확도는 제안된 방법의 MAP를 기존 OTI-SW 방법의 MAP로 나눈 비로 주어진다. 상대 정확도를 보면 코드북 사이즈가 커짐에 따라서 제안된 코드 기반 방법의 성능이 기존 OTI-SW 성능에 근접함을 알 수 있다. 특히 코드북 사이즈 K값이 48보다 크게 되면 두 데이터셋 모두에서 상대 정확도가 0.98을 넘는 것을 확인할 수 있었다. 기존 OTI-SW 방법의 문제인 계산량과 저장공간이 줄어드는 정도를 각각 상대 계산량(Relative Computational cost, RC)과 상대 저장공간(Relative Storage cost, RS)를 사용하여 확인하였다. 상대 계산량은 유사도 행렬 계산에 있어서 제안된 방법에서 소요되는 곱하기의 개수를 기존 OTI-SW 방법에서 소요되는 곱하기의 개수로 나눈 비로 주어진다. 코드북 사이즈 K값이 48인 경우에도

Table 1. Identification performance of the covers80 dataset. Accuracy measures are precision at one, P@1, and the mean of average precision, MAP. RP, RC, and RS refer to relative precision, relative computational cost, and relative storage cost respectively.

Method	K	P@1	MAP	RP	RC	RS
Code-based OTI-SW	16	0.538	0.613	0.971	0.001	0.042
	32	0.544	0.621	0.983	0.004	0.052
	48	0.545	0.624	0.987	0.009	0.058
	64	0.546	0.625	0.989	0.016	0.063
	80	0.552	0.628	0.994	0.026	0.066
OTI-SW[4]		0.550	0.632	1	1	1

Table 2. Identification performance of the covers330 dataset. Accuracy measures are the mean number of covers identified within the ten first answers, $MNCI_{10}$, and the mean of average precision, MAP. RP, RC, and RS refer to relative precision, relative computational cost, and relative storage cost respectively.

Method	K	$MNCI_{10}$	MAP	RP	RC	RS
Code-based OTI-SW	16	6.035	0.630	0.916	0.001	0.042
	32	6.368	0.663	0.964	0.004	0.052
	48	6.482	0.674	0.980	0.009	0.058
	64	6.528	0.679	0.988	0.016	0.063
	80	6.557	0.682	0.992	0.026	0.066
OTI-SW[4]		6.609	0.688	1	1	1

상대 계산량이 1 %도 되지 않음을 알 수 있다. 상대 저장공간은 2.2절에서 언급했듯이 $\log_2(K)/(L*8$비트) 형태로 계산했다. 각 프레임의 크로마그램이 L차원이고 각 원소는 8 bit로 표현한다고 가정하였다. Tables 1과 2에서 크로마그램 벡터 수열을 코드 비트 수열로 변환함으로써 저장 공간도 크게 줄인다. 실험 결과로부터 제안된 방법이 기존 방법과 유사한 커버곡 검색 성능을 유지하면서 계산량과 특징 저장 공간을 크게 줄일 수 있음을 알 수 있다. 실제 대용량 음악 아카이브 상에서 커버곡 검색 서비스를 구현할 때 계산량 및 저장 공간 등의 시스템 비용과 커버곡 검색 정확도 간의 균형을 조정할 필요가 있으며, 제안된 방법에서는 코드북 사이즈를 가변하여 시스템 비용과 검색 정확도를 조정할 수 있다.

IV. 결 론

코드북을 학습하여 크로마그램을 코드 비트로 변환하는 방법을 제안하였다. 코드로 변환함으로써 실제 검색 서비스 적용 시에 필요한 특징 저장 공간을 줄일 수 있다. 또한 코드북간의 OTI 거리를 미리 계산하여 LUT 형태로 저장하여, 수열간 유사도 행렬을 만드는 과정의 계산량을 크게 줄였다. 실험을 통해서 코드북 사이즈를 줄이더라도 커버곡 검색 성능이 크게 열화되지 않음을 확인하였다.

감사의 글

이 논문은 2018년도 강릉원주대학교 학술연구조성비 지원에 의하여 수행되었음.

References

1. M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, “Content-based music information retrieval: Current directions and future challenges,” Proc. the IEEE **96**, 668-696 (2008).
2. J. -Y. Lee and H. -G. Kim, “Audio fingerprinting using a robust hash function based on the MCLT peak-pair” (in Korean), J. Acoust. Soc. Kr. **34**, 157-162 (2015).
3. J. S. Seo, J. Kim, and J. Park, “Centroid-model based music similarity with alpha divergence” (in Korean), J. Acoust. Soc. Kr. **35**, 83-91 (2016).
4. J. Serra, E. Gomez, P. Herrera, and X. Serra, “Chroma binary similarity and local alignment applied to cover song identification,” IEEE Trans. Audio Speech Lang Process. **16**, 1138-1151 (2008).
5. J. S. Seo, “Cover song search based on magnitude and phase of the 2D Fourier transform” (in Korean), J. Acoust. Soc. Kr. **37**, 518-524 (2018).
6. M. Muller and S. Ewert, “Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features,” Proc. ISMIR-2011, 215-220 (2011).
7. P. Foster, S. Dixon, and A. Klapuri, “Identifying cover songs using information-theoretic measures of similarity,” IEEE Trans. Audio Speech Lang. Process. **23**, 993-1005 (2015).
8. D. F. Silva, C. -C. Yeh, G. E. A. P. A. Batista, and E. Keogh, “SIMPle: Assessing music similarity using subsequences joins,” Proc. ISMIR-2016, 23-29 (2016).
9. T. F. Smith and M. S. Waterman, “Identification of common molecular subsequences,” J. Molecular Biology **147**, 195-197 (1981).
10. E. Nowak, F. Jurie, and B. Triggs, “Sampling strategies for bag-of-features image classification,” Proc. ECCV-2006, 490-503 (2006).
11. L. Wu, S. C. H. Hoi, and N. Yu, “Semantics-preserving bag-of-words models and applications,” IEEE Trans. Image Process. **19**, 1908-1920 (2010).
12. D. P. W. Ellis and G. E. Poliner, “Identifying cover songs with chroma features and dynamic programming beat tracking,” Proc. ICASSP-2007, 1429-1432 (2007).
13. *Covers80 Cover Song Data Set, Available*, https://labrosa.ee.columbia.edu/projects/coversongs/covers80/, 2007.
14. M. Muller and S. Ewert, “Towards timbre-invariant audio features for harmony-based music,” IEEE Trans. Audio Speech Lang. Process. **18**, 649-662 (2010).

저자 약력

▸ 서 진 수 (Jin Soo Seo)

1998년 2월: KAIST 전기 및 전자공학과 공학사
2000년 2월: KAIST 전기 및 전자공학과 공학석사
2005년 2월: KAIST 전기 및 전자공학과 공학박사
2006년 3월 ~ 2008년 2월: 한국전자통신연구원 선임연구원
2008년 3월 ~ 현재: 강릉원주대학교 전자공학과 교수