



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0036851
(43) 공개일자 2017년04월03일

(51) 국제특허분류(Int. Cl.)
G06Q 50/10 (2012.01) G06F 17/30 (2006.01)
(52) CPC특허분류
G06Q 50/10 (2015.01)
G06F 17/30 (2013.01)
(21) 출원번호 10-2015-0132272
(22) 출원일자 2015년09월18일
심사청구일자 2015년09월18일

(71) 출원인
이화여자대학교 산학협력단
서울특별시 서대문구 이화여대길 52 (대현동, 이화여자대학교)
(72) 발명자
채현경
서울특별시 서대문구 연희로15안길 61-6 (연희동)
이형준
서울특별시 성동구 금호로 17, 209동 703호 (금호동4가, 서울숲2차푸르지오)
(뒷면에 계속)
(74) 대리인
특허법인(유한)아이시스

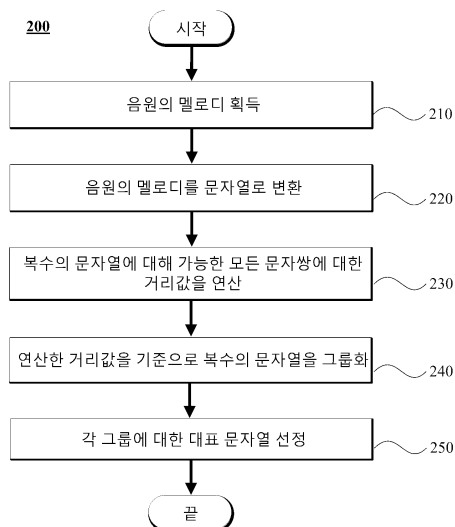
전체 청구항 수 : 총 11 항

(54) 발명의 명칭 멜로디에 기반한 음악 분류 방법 및 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법

(57) 요약

멜로디에 기반한 음악 분류 방법은 컴퓨터 장치가 복수의 음원에 대해 멜로디에 해당하는 복수의 문자열을 획득하는 단계, 상기 컴퓨터 장치가 상기 복수의 문자열 중 문자열쌍 사이의 유사도를 나타내는 거리값을 연산하는 단계 및 상기 컴퓨터 장치가 상기 문자열쌍에 대한 상기 거리값을 사용하여 상기 문자열을 복수의 그룹으로 분류하는 단계를 포함한다.

대표도 - 도2



(72) 발명자

김은하

서울특별시 서초구 헌릉로8길 22, 304동 201호 (신원동, 서초포레스타3단지)

이경자

서울특별시 구로구 디지털로31길 90, 114동 1902호(구로동, 삼성래미안아파트)

이 발명을 지원한 국가연구개발사업

과제고유번호 2014S1A5B8066410

부처명 교육부

연구관리전문기관 한국연구재단

연구사업명 대학중점연구소지원사업

연구과제명 '협업'과 '개인맞춤'을 통한 새로운 음악문화 콘텐츠 창출

기 여 율 1/1

주관기관 이화여자대학교 산학협력단

연구기간 2014.12.01 ~ 2017.11.30

명세서

청구범위

청구항 1

컴퓨터 장치가 복수의 음원에 대해 멜로디에 해당하는 복수의 문자열을 획득하는 단계;

상기 컴퓨터 장치가 상기 복수의 문자열 중 문자열쌍 사이의 유사도를 나타내는 거리값을 연산하는 단계; 및

상기 컴퓨터 장치가 상기 문자열쌍에 대한 상기 거리값을 사용하여 상기 문자열을 복수의 그룹으로 분류하는 단계를 포함하는 멜로디에 기반한 음악 분류 방법.

청구항 2

제1항에 있어서,

상기 문자열은 음의 높이 또는 음의 길이 중 적어도 하나에 해당하는 문자로 구성되는 멜로디에 기반한 음악 분류 방법.

청구항 3

제2항에 있어서,

상기 문자가 음의 높이를 나타내는 경우 동일한 문자라도 서로 다른 옥타브인 경우 서로 다른 값으로 식별되는 멜로디에 기반한 음악 분류 방법.

청구항 4

제1항에 있어서,

상기 문자열쌍은 제1 문자열 및 제2 문자열을 포함하고,

상기 거리값은 상기 문자열쌍 중 동일한 부분의 길이, 상기 제1 문자열을 기준으로 상기 제2 문자열에 삽입된 문자의 개수, 상기 제1 문자열을 기준으로 상기 제2 문자열에 삭제된 문자의 개수 및 상기 제1 문자열을 기준으로 상기 제2 문자열에서 위치가 변경된 문자의 개수를 기준으로 연산되는 멜로디에 기반한 음악 분류 방법.

청구항 5

제1항에 있어서,

상기 거리값(D)은 아래의 수식으로 연산되는 멜로디에 기반한 음악 분류 방법.

$$D(i, j) = \begin{cases} D(i-1, j-1) & \text{if } X_i = Y_j \\ \min \begin{cases} D(i-1, j) + w_{ins} \\ D(i, j-1) + w_{del} \\ D(i-1, j-1) + w_{sub}(X_i, Y_j) \end{cases} & \text{otherwise} \end{cases}$$

(여기서, X 및 Y는 문자열쌍을 구성하는 문자열이고, X_i 는 문자열 X에서 i번째 문자, Y_j 는 문자열 Y에서 j번째 문자, w_{ins} 는 문자 삽입에 부여되는 가중치, w_{del} 은 문자 삭제에 부여되는 가중치, $w_{sub}(X_i, Y_j)$ 는 문자의 순서 치환에 부여되는 가중치임)

청구항 6

제5항에 있어서,

상기 컴퓨터 장치는 상기 복수의 문자열 중 길이가 다른 문자열이 있는 경우 상기 문자열쌍에 대한 거리값을 상기 문자열쌍을 구성하는 어느 하나의 문자열의 길이를 기준으로 상기 거리값을 정규화하는 멜로디에 기반한 음악 분류 방법.

청구항 7

제1항에 있어서,

상기 컴퓨터 장치는 상기 거리값을 기준으로 상기 거리값이 기준 범위에 있는 문자열들을 하나의 그룹으로 분류하는 멜로디에 기반한 음악 분류 방법.

청구항 8

제1항에 있어서,

상기 컴퓨터 장치가 상기 복수의 그룹 각각에 대해 가장 긴 공통 문자열을 각 그룹에 대한 대표 문자열로 설정하는 단계;

상기 컴퓨터 장치가 새로운 음원의 멜로디에 해당하는 입력 문자열을 획득하는 단계; 및

상기 컴퓨터 장치가 상기 대표 문자열과 상기 입력 문자열의 유사도를 연산하여 상기 입력 문자열이 소속되는 그룹을 결정하는 단계를 더 포함하는 멜로디에 기반한 음악 분류 방법.

청구항 9

컴퓨터 장치가 복수의 음원에 대해 멜로디에 해당하는 복수의 문자열을 획득하는 단계;

상기 컴퓨터 장치가 상기 복수의 문자열 중 문자열쌍 사이의 유사도를 나타내는 거리값을 연산하는 단계;

상기 컴퓨터 장치가 상기 문자열쌍에 대한 상기 거리값을 사용하여 상기 문자열을 복수의 그룹으로 분류하는 단계;

상기 컴퓨터 장치가 상기 복수의 그룹 각각에 대해 가장 긴 공통 문자열을 각 그룹에 대한 대표 문자열로 설정하는 단계;

상기 컴퓨터 장치가 입력 문자열을 획득하는 단계;

상기 컴퓨터 장치가 상기 대표 문자열과 상기 입력 문자열의 유사도를 기준으로 상기 복수의 그룹 중 적어도 복수의 후보 그룹을 결정하는 단계; 및

상기 컴퓨터 장치가 상기 복수의 후보 그룹의 대표 문자열에서 문자 출현 빈도를 기준으로 순서에 따라 멜로디에 해당하는 문자열을 추천하는 단계를 포함하는 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법.

청구항 10

제9항에 있어서,

상기 거리값(D)은 아래의 수식으로 연산되는 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법.

$$D(i, j) = \begin{cases} D(i-1, j-1) & \text{if } X_i = Y_j \\ \min \begin{cases} D(i-1, j) + w_{ins} & \text{otherwise} \\ D(i, j-1) + w_{del} \\ D(i-1, j-1) + w_{sub}(X_i, Y_j) \end{cases} & \end{cases}$$

(여기서, X 및 Y는 문자열쌍을 구성하는 문자열이고, X_i 는 문자열 X에서 i번째 문자, Y_j 는 문자열 Y에서 j번째 문자, w_{ins} 는 문자 삽입에 부여되는 가중치, w_{del} 은 문자 삭제에 부여되는 가중치, $w_{sub}(X_i, Y_j)$ 는 문자의 순서 치환에 부여되는 가중치임)

청구항 11

제9항에 있어서,

상기 컴퓨터 장치는 상기 입력 문자열과 상기 대표 문자열의 유사도를 아래의 수식에서 M으로 연산하고, 연산한 유사도가 높은 순서에 따라 기준 개수의 후보 그룹을 결정하는 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법.

$$M(i, j) = \begin{cases} M(i-1, j-1) + w_{mat} & \text{if } S_i = X'_j \\ \max \begin{cases} M(i-1, j) \\ M(i, j-1) \\ M(i-1, j-1) + \alpha/w_{sub}(S_i, X'_j) \end{cases} & \text{otherwise} \end{cases}$$

(S는 대표 문자열, X'는 입력 문자열, S_i는 문자열 S에서 i번째 문자, X'_j는 문자열 X'에서 j번째 문자, w_{mat}는 문자가 동일한 경우에 부여되는 값, α는 상수값, w_{sub}(S_i, X'_j)는 문자의 순서 치환에 부여되는 가중치임)

발명의 설명

기술 분야

[0001] 이하 설명하는 기술은 멜로디에 기반하여 음악을 분류하는 기법에 관한 것이다.

배경 기술

[0002] 최근 인터넷 포털 업체 등이 음악 서비스를 제공하면서, 스마트폰과 같은 기기에서 자신이 원하는 음악을 검색하는 서비스가 제공되고 있다. 스마트폰을 이용한 음악 검색은 일반적으로 핑거 프린팅(finger printing) 기법 등을 이용하여 음악이 재생되는 소리를 분석하는 방식이다.

선행기술문헌

특허문헌

[0003] (특허문헌 0001) 한국공개특허 10-2006-0091063호

발명의 내용

해결하려는 과제

[0004] 음악을 분류하거나 검색하기 위해서는 기본적으로 DB(데이터베이스)가 필요하다. 종래 음악을 검색하는 기법은 대부분 소리(sound)를 이용하여 음악을 검색한다. 이 경우 DB는 소리라는 데이터를 저장한다. 일반적으로 소리라는 데이터는 비교적 고량의 데이터이므로 DB의 용량이 크다.

[0005] 이하 설명하는 기술은 멜로디를 나타내는 문자(character) 정보를 기반으로 음악을 분류하는 기법을 제공하고자 한다.

과제의 해결 수단

[0006] 멜로디에 기반한 음악 분류 방법은 컴퓨터 장치가 복수의 음원에 대해 멜로디에 해당하는 복수의 문자열을 획득하는 단계, 상기 컴퓨터 장치가 상기 복수의 문자열 중 문자열쌍 사이의 유사도를 나타내는 거리값을 연산하는 단계 및 상기 컴퓨터 장치가 상기 문자열쌍에 대한 상기 거리값을 사용하여 상기 문자열을 복수의 그룹으로 분류하는 단계를 포함한다.

[0007] 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법은 컴퓨터 장치가 복수의 음원에 대해 멜로디에 해당하는 복수의 문자열을 획득하는 단계, 상기 컴퓨터 장치가 상기 복수의 문자열 중 문자열쌍 사이의 유사도를 나타내는 거리값을 연산하는 단계, 상기 컴퓨터 장치가 상기 문자열쌍에 대한 상기 거리값을 사용하여 상기 문자열을 복수의 그룹으로 분류하는 단계, 상기 컴퓨터 장치가 상기 복수의 그룹 각각에 대해 가장 긴 공통 문자열을 각 그룹에 대한 대표 문자열로 설정하는 단계, 상기 컴퓨터 장치가 입력 문자열을 획득하는 단계, 상기 컴퓨터 장치가 상기 대표 문자열과 상기 입력 문자열의 유사도를 기준으로 상기 복수의 그룹 중 적어도 복수의 후보 그룹을 결정하는 단계 및 상기 컴퓨터 장치가 상기 복수의 후보 그룹의 대표 문자열에서 문자 출현 빈도를 기준으로 순서에 따라 멜로디에 해당하는 문자열을 추천하는 단계를 포함한다.

발명의 효과

[0008] 이하 설명하는 기술은 음의 높이를 나타내는 문자 정보를 사용하여 음악을 분류한다. 따라서 이하 설명하는 기술은 문자라는 비교적 작은 용량의 데이터가 저장되는 DB를 구성할 수 있다. 나아가 이하 설명하는 기술은 DB를 이용하여 음악을 검색하는 경우 문자 정보에 기반하여 보다 빠르게 음악을 검색할 수 있다. 또한 이하 설명하는 기술은 멜로디 또는 리듬이 일부 상이한 경우를 고려하여 음악을 분류하고 검색한다.

도면의 간단한 설명

[0009] 도 1은 음악 데이터베이스를 생성하는 시스템의 구성을 도시한 블록도의 예이다.
 도 2는 음악 분류 방법에 대한 순서도의 예이다.
 도 3은 음악 데이터베이스에 저장되는 정보에 대한 예이다.
 도 4는 음악 검색 시스템의 구성을 도시한 블록도의 예이다.
 도 5는 멜로디를 추천하는 시스템의 구성을 도시한 블록도의 예이다.
 도 6은 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법에 대한 순서도의 예이다.

발명을 실시하기 위한 구체적인 내용

[0010] 이하 설명하는 기술은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다. 그러나, 이는 이하 설명하는 기술을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 이하 설명하는 기술의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0011] 제1, 제2, A, B 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 해당 구성요소들은 상기 용어들에 의해 한정되지는 않으며, 단지 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 이하 설명하는 기술의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.

[0012] 본 명세서에서 사용되는 용어에서 단수의 표현은 문맥상 명백하게 다르게 해석되지 않는 한 복수의 표현을 포함하는 것으로 이해되어야 하고, "포함한다" 등의 용어는 실시된 특징, 개수, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 의미하는 것이지, 하나 또는 그 이상의 다른 특징들이나 개수, 단계 동작 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 배제하지 않는 것으로 이해되어야 한다.

[0013] 도면에 대한 상세한 설명을 하기에 앞서, 본 명세서에서의 구성부들에 대한 구분은 각 구성부가 담당하는 주기능 별로 구분한 것에 불과함을 명확히 하고자 한다. 즉, 이하에서 설명할 2개 이상의 구성부가 하나의 구성부로 합쳐지거나 또는 하나의 구성부가 보다 세분화된 기능별로 2개 이상으로 분화되어 구비될 수도 있다. 그리고 이하에서 설명할 구성부 각각은 자신이 담당하는 주기능 이외에도 다른 구성부가 담당하는 기능 중 일부 또는 전부의 기능을 추가적으로 수행할 수도 있으며, 구성부 각각이 담당하는 주기능 중 일부 기능이 다른 구성부에 의해 전담되어 수행될 수도 있음은 물론이다.

[0014] 또, 방법 또는 동작 방법을 수행함에 있어서, 상기 방법을 이루는 각 과정들은 문맥상 명백하게 특정 순서를 기재하지 않은 이상 명기된 순서와 다르게 일어날 수 있다. 즉, 각 과정들은 명기된 순서와 동일하게 일어날 수도 있고 실질적으로 동시에 수행될 수도 있으며 반대의 순서대로 수행될 수도 있다.

- [0016] 이하 설명하는 기술은 음악을 분류하고 검색하는 기법에 관련된다. 이하 설명하는 기술은 음악의 소리가 아닌 멜로디를 나타내는 문자정보를 기준으로 음악을 분류하고, 음악 데이터베이스를 생성한다. 문자정보는 멜로디는 음의 높이(pitch)를 나타내는 문자열(string)으로 표현될 수 있다. 예컨대, 서양 음계(7음계)는 "도 = C, 레 = D, 미 = E, 파 = F, 솔 = G, 라 = A, 시 = B"로 표현된다. 이 경우 멜로디는 "CGGADA...."와 같은 문자열로 표현될 수 있다. 음의 높이는 옥타브에 따라 달라지므로 각 옥타브마다 다른 문자 또는 별도의 식별자는 사용하여 구분할 수도 있다. 나아가 사전에 정의한다면 음의 높이는 알파벳과 같은 문자열 외에도 숫자 등과 같은 문자로 표현될 수도 있다.
- [0017] 또한 문자정보는 멜로디 외에 음의 길이를 나타내는 정보를 포함할 수 있다. 음악은 멜로디 외에도 음의 길이의 장단에 따라 발생하는 리듬도 포함한다. 따라서 문자정보가 리듬을 나타내는 정보를 포함할 필요가 있다. 예컨대, 음의 길이를 일정한 숫자로 표현할 수 있다. 물론 숫자 외에 다른 문자를 사용하여 표현할 수도 있을 것이다. 숫자를 사용한다고 가정하면 문자열은 "C1 G1 A2 E3..."와 같이 표현될 수 있다. 알파벳은 음의 높이를 나타내고, 알파벳 우측의 숫자는 음의 길이를 나타낸다. 또한 음이 발생하지 않고 쉬는 구간도 나타낼 수 있다. 음이 발생하지 않는 상태를 Z라고 표현하면, Z 다음에 숫자를 표시하여 음 발생 없이 쉬는 구간의 길이를 표현할 수 있다.
- [0019] 먼저 음악 데이터베이스를 생성하는 과정을 설명한다. 음악 데이터베이스 생성을 위해서는 사전에 입수가 가능한 다양한 음악 샘플이 필요하다. 음악 데이터베이스는 음악 샘플을 일정한 기준에 따라 분류하여 생성한다. 즉, 음악 샘플을 사용하여 음악 검색 등을 위한 데이터베이스를 미리 마련하는 것이다.
- [0020] 도 1은 음악 데이터베이스를 생성하는 시스템(100)의 구성을 도시한 블록도의 예이다. 사용자는 컴퓨터(50)를 통해 음악 샘플에 대한 멜로디를 서버(120)에 전달한다. 여기서 음악 샘플은 복수의 음원에 관한 것이고, 서버(120)는 음악 샘플을 멜로디에 해당하는 문자열을 기준으로 일정하게 분류한다. 서버(120)는 최종적으로 문자열을 기준으로 음악을 분류하는 과정을 수행한다.
- [0022] 음악을 나타내는 데이터 포맷이 무엇인지에 따라 다음과 같은 절차 흐름이 가능하다.
- [0023] 1. 음악을 나타내는 최초 데이터가 문자열 데이터일 수 있다. 이 경우 컴퓨터(50)가 문자열 형태로 멜로디를 서버(120)에 전달하면 된다.
- [0024] 2. 음악을 나타내는 최초 데이터가 소리(sound) 데이터일 수 있다. (1) 컴퓨터(50)가 음원 파일(mp3, wav 등의 포맷)을 신호 처리 과정을 통해 멜로디 라인(line)에 해당하는 문자열하고, 해당 문자열을 서버(120)에 전달할 수도 있다. (2) 컴퓨터(50)가 음원 파일을 서버(120)에 전달하고, 서버(120)가 음원 파일에서 멜로디 라인을 추출하여 각 음원의 멜로디에 해당하는 문자열을 생성할 수도 있다. 이후 서버(120)는 각 음악 샘플에 대한 문자열을 기준으로 분류 작업을 수행한다.
- [0025] 3. 음악을 나타내는 최초 데이터가 악보에 해당하는 이미지 데이터일 수 있다. (1) 컴퓨터(50)가 악보에 해당하는 이미지에서 멜로디를 결정하고, 멜로디에 해당하는 문자열을 생성하여 해당 문자열을 서버(120)에 전달할 수도 있다. (2) 컴퓨터(50)가 이미지 데이터를 서버(120)에 전달하고, 서버(120)가 이미지 처리를 통해 멜로디에 해당하는 문자열을 생성할 수도 있다. 이 경우 이미지 데이터를 이용한 멜로디 추출은 OMR(Optical Music Recognition)과 같은 기술을 이용할 수 있다.
- [0026] 4. 음악을 나타내는 최초 데이터가 컴퓨터 장치에서 악보를 나타내는 디지털 콘텐츠 일 수 있다. 예컨대, MIDI 파일과 같은 콘텐츠가 이에 해당한다. 이 경우 (1) 컴퓨터(50)가 디지털 콘텐츠로 음악을 재생하는 프로그램을 이용하여, 디지털 콘텐츠에서 멜로디를 추출하고, 해당 멜로디를 문자열로 변환할 수 있다. 컴퓨터는 변환한 문자열을 서버(120)에 전달할 수 있다. (2) 또는 서버(120)가 디지털 콘텐츠를 컴퓨터(50)로부터 수신하고, 디지털 콘텐츠로부터 문자열을 생성할 수도 있다.
- [0028] 한편 도 1의 절차 흐름과는 다른 경우도 있을 수 있다. 컴퓨터(50)가 음악 샘플에 대한 문자열을 직접 음악 데이터베이스(130)에 전달하고, 서버(120)가 음악 데이터베이스(130)에 접근하여 저장된 문자열을 기준으로 음악

분류 작업을 하고, 분류된 결과를 다시 음악 데이터베이스(130)에 저장할 수도 있다.

[0029] 음악 데이터베이스(130)는 최종적으로 서버(120)가 분류한 음악 정보를 저장한다. 도 1은 음악 데이터베이스(130)가 저장하는 데이터에 대한 하나의 예를 도시하였다. 음악 데이터베이스(130)는 기본적으로 멜로디를 표현하는 문자열을 저장하고, 부가적으로 각 음악에 대한 부가 정보인 음악 정보를 저장할 수 있다. 음악 정보는 음악의 제목, 작곡가, 작사자, 음악의 장르, 가사 등과 같은 정보를 포함할 수 있다. 음악 데이터베이스에 저장되는 구체적인 데이터에 대해서는 후술한다.

[0031] 도 2는 음악 분류 방법(200)에 대한 순서도의 예이다. 컴퓨터(50) 또는 서버(120)가 샘플 음악에 대한 음원 멜로디를 획득한다(210). 음원 멜로디는 도 1에서 설명한 바와 같이 문자열, 소리 데이터, 이미지 데이터, 디지털 콘텐츠 등과 같은 다양한 형태 일 수 있다. 이후 컴퓨터(50) 또는 서버(120)가 음원 멜로디를 문자열 형태로 변환한다(220). 이후 서버(120)는 문자열 형태의 샘플 음악을 분류하는 과정을 시작한다.

[0032] 서버(120)는 음악 샘플에 대한 복수의 문자열(하나의 문자열이 하나의 음악 샘플에 대응됨)에서 가능한 모든 문자쌍에 대해 거리값을 연산한다. 여기서 거리값을 각 문자쌍의 유사도와 관련된다. 서버(120)는 복수의 문자열을 이용하여 생성 가능한 모든 문자쌍에 대해 거리값을 연산하고, 연산된 거리값을 기준으로 복수의 문자열을 그룹화(clustering)한다.

[0033] 문자쌍의 거리값은 2개의 문자열을 구성하는 문자들이 얼마나 유사한지를 나타낸다. 하나의 문자쌍이 제1 문자열과 제2 문자열을 포함한다고 가정한다. 거리값은 제1 문자열을 기준으로 제2 문자열에 새롭게 추가된 문자가 있는지, 제1 문자열을 기준으로 제2 문자열에 삭제된 문자가 있는지 또는 제1 문자열을 기준으로 제2 문자열에서 순서가 바뀐 문자가 있는지 등을 고려하여 결정된다. 아래 수학적 식 1은 거리값을 연산하는 수식에 해당한다. 거리값 연산은 동적 프로그래밍(dynamic programming)을 사용한다.

수학적 식 1

$$D(i, j) = \begin{cases} D(i-1, j-1) & \text{if } X_i = Y_j \\ \min \begin{cases} D(i-1, j) + w_{ins} \\ D(i, j-1) + w_{del} \\ D(i-1, j-1) + w_{sub}(X_i, Y_j) \end{cases} & \text{otherwise} \end{cases}$$

[0036] 여기서, X 및 Y는 문자열쌍을 구성하는 문자열이고, X_i 는 문자열 X에서 i번째 문자, Y_j 는 문자열 Y에서 j번째 문자, w_{ins} 는 문자 삽입에 부여되는 가중치, w_{del} 은 문자 삭제에 부여되는 가중치, $w_{sub}(X_i, Y_j)$ 는 문자의 순서 치환에 부여되는 가중치이다.

[0037] $X[1, \dots, m]$ 이란 문자열을 $Y[1, \dots, n]$ 으로 변환한다고 가정하면, w_{ins} 는 변환되면서 문자가 삽입되는 경우에 대한 가중치이고, w_{del} 는 변환되면서 삭제되는 문자에 대한 가중치이고, $w_{sub}(X_i, Y_j)$ 는 변환되면서 순서가 치환되는 문자에 대한 가중치이다. X 문자열 및 Y 문자열의 거리를 $D(m, n)$ 으로 연산할 수 있다.

[0038] 한편 서버(120)는 메모리, 연산 장치 등을 갖고 일정한 작업을 수행하는 장치이다. 따라서 서버(120)도 일종의 컴퓨터 장치에 해당한다. 따라서 컴퓨터 장치가 전술한 음악 분류 방법(200)을 수행한다고 표현할 수도 있다.

[0039] 상기 수학적 식 1과 같은 동적 프로그래밍 기법을 사용하여 멜로디 또는/및 리듬이 약간 상이한 경우라도 유사한 음악으로 분류하고 검색할 수도 있다. 상기 수학적 식 1에 부여되는 가중치의 값을 조절하여 일정하게 변형된 멜로디 또는 리듬도 유사한 음악으로 분류할 수 있다. 예컨대, 동일한 멜로디의 중간에 새로운 멜로디 라인을 집어 넣는 경우에도 유사한 음악으로 분류할 수 있다.

[0041] 다만 각 문자열의 길이가 서로 다를 수 있다. 물론 음악 샘플에서 일정한 구역의 일정한 크기의 문자열을 사용할 수도 있다. 문자열의 길이가 다른 경우, 각 문자열쌍의 거리값을 정규화(normalization)하는 것이 바람직하다.

[0042] 복수의 문자열 중 길이가 다른 문자열이 있는 경우 문자열쌍에 대한 거리값을 문자열쌍을 구성하는 어느 하나의 문자열의 길이를 기준으로 거리값을 정규화할 수 있다. 아래 수학적 식 2는 문자열쌍 중 길이가 짧은 문자열로 삼은 경우이다.

수학적 식 2

$$\overline{D}(m, n) = \frac{D(m, n)}{\min(m, n)}$$

[0044]

[0045] $\overline{D}(m, n)$ 은 $D(m, n)$ 을 정규화한 결과값이다. 여기서 $D(m, n)$ 은 전술한대로 X 문자열 및 Y 문자열 사이의 거리값이고 $\min(m, n)$ 은 X 문자열과 Y 문자열 중 길이가 짧은 문자열의 길이값이다.

[0047] 서버(120)는 모든 문자열쌍에 대한 거리값 연산이 완료되면, 문자열쌍에 대한 거리값을 기준으로 복수의 문자열을 일정하게 그룹화한다. 이 과정에서 다양한 알고리즘이 사용될 수 있다. 예컨대, 트리와 같은 계층적 구조를 갖는 형태로 복수의 문자열을 그룹화할 수도 있다(소위 hierarchical clustering). 이 경우 전체 그룹의 개수를 적절하게 설정할 수 있다.

[0048] 마지막으로 서버(120)는 복수의 그룹에 대한 대표 문자열을 선정할 수 있다(250). 대표 문자열은 복수의 그룹을 대표하는 문자열을 의미한다. 음악 검색 등과 같은 후속 작업을 위해 대표 문자열 선정이 필요하다. 다양한 방법이 사용될 수 있다. 예컨대, 대표 문자열은 해당 그룹에 속한 문자열들이 공통으로 갖는 문자열 중 가장 긴 문자열(longest common subsequence)로 설정할 수 있다.

[0050] 도 3은 음악 데이터베이스(130)에 저장되는 정보에 대한 예이다. 도 3을 살펴보면, 음악 데이터베이스(130)는 예시적으로 3개의 그룹을 포함하고 있다(도 3에서 3개의 그룹만을 도시하였음). 음악 데이터베이스(130)는 각 그룹에 속한 문자열들을 포함하고, 경우에 따라서는 각 문자열인 음악에 대한 음악 정보를 포함할 수도 있다. 또한 음악 데이터베이스(130)는 각 그룹을 대표하는 대표 문자열을 별도로 저장하고 관리한다.

[0052] 도 4는 음악 검색 시스템(300)의 구성을 도시한 블록도의 예이다. 음악 데이터베이스(330)는 도 1에서 샘플 음악을 분류한 문자열을 저장하고 있는 음악 데이터베이스(130)와 동일한 구성이다.

[0053] 음악을 검색하고자 하는 사용자는 컴퓨터(50) 또는 스마트폰과 같은 휴대 단말기(60)를 통해 멜로디를 입력한다. 음악 검색에 사용되는 멜로디를 입력 멜로디라고 명명한다. (1) 컴퓨터(50) 또는 휴대 단말기(60)는 멜로디에 해당하는 문자열을 직접 서버(320)에 전달할 수 있다. (2) 컴퓨터(50) 또는 휴대 단말기(60)는 음원의 소리(sound)를 마이크로 수신하고, 이를 신호 처리 과정을 통해 멜로디 라인(line)에 해당하는 문자열로 변환하고, 변환한 문자열을 서버(320)에 전달할 수도 있다. (3) 컴퓨터(50) 또는 휴대 단말기(60)는 음원의 소리(sound)를 서버(320)에 전달하고, 서버(320)가 음원의 소리에서 멜로디를 추출하여 문자열을 생성할 수도 있다. 나아가 도 1에서 설명한 바와 같이 입력 멜로디는 악보 이미지일 수도 있고, 음악 재생에 사용하는 별도의 디지털 콘텐츠일 수도 있다. 최종적으로 서버(320)는 음의 높이를 나타내는 문자열 형태의 입력 멜로디를 획득한다. 이하 입력 멜로디의 문자열을 입력 문자열이라고 명명한다.

[0054] 서버(320)는 입력 문자열과 음악 데이터베이스(330)에 저장된 각 그룹의 대표 문자열의 유사도를 비교한다. 입력 문자열과 대표 문자열의 유사도는 다양한 알고리즘을 통해 결정할 수 있다. 또한 전술한 수학적 식 1로 표현되는 거리값을 기준으로 입력 문자열과 대표 문자열의 유사도를 결정할 수도 있다. 서버(320)는 입력 문자열과 가

장 유사도가 높은 대표 문자열을 결정하고, 결정한 대표 문자열이 나타내는 그룹에 속한 문자열과 입력 문자열을 2차 비교할 수 있다. 2차 비교과정에서 서버(320)는 동일한 문자열을 검색하여, 동일한 문자열이 있는 경우 해당 음악에 대한 정보를 컴퓨터(50) 또는 휴대 단말기(60)에 전달한다.

[0055] 또는 문자열을 이용한 검색 기법을 종래의 핑거 프린팅 기법과 혼합하여 사용할 수도 있을 것이다. 종래의 핑거 프린팅 기법이 음원의 소리를 기준으로 음악을 검색하는 것으로, 음원의 종류가 기하급수적으로 많아지는 시전에서는 음악 검색에 다소 시간이 지체될 수 있다. 이 경우 전술한 문자열을 이용한 검색 기법은 입력 문자열을 기준으로 입력 문자열이 속하는 그룹만을 결정하고, 이후 핑거 프린팅 기법을 이용하여 검색대상을 그룹에 속한 음악만을 대상으로 한정할 수 있다.

[0056] 한편 음악 검색은 거리값을 기준으로 결정할 수 있고, 이 경우 완벽하게 동일하지 않아도 입력 문자열과 상당히 유사한 음악을 검색할 수 있다. 이를 통해 새로운 음악의 표절 여부 판단을 할 수도 있다.

[0058] 도 5는 멜로디를 추천하는 시스템(400)의 구성을 도시한 블록도의 예이다. 도 5(a)는 멜로디를 추천하는 시스템(400)의 구성을 도시하고, 도 5(b)는 서버(420)가 멜로디를 추천하는 과정에 대한 하나의 예를 도시한다. 도 5의 시스템(400)은 도 4의 시스템(300)과 구성이 동일하다. 다만 서버(420)가 입력 멜로디 이후에 사용될 수 있는 멜로디 라인을 추천하는 기능을 수행한다. 멜로디 추천은 음악 작곡 등에 이용할 수 있다.

[0059] 도 6은 음악 데이터베이스를 이용하여 멜로디 라인을 추천하는 방법에 대한 순서도의 예이다. 이하 도 5와 도 6을 기준으로 설명한다. 음악 데이터베이스(430)는 도 1의 과정을 통해 마련된 것을 전제로 한다(510). 도 5에서 음악 데이터베이스(430)는 도 1에서 샘플 음악을 분류한 문자열을 저장하고 있는 음악 데이터베이스(130)와 동일한 구성이다.

[0060] 사용자는 컴퓨터(50) 또는 휴대 단말기(60)를 통해 입력 멜로디를 서버(420)에 전달한다(520). 서버(420)가 입력 멜로디를 전달받는 과정은 도 4와 동일하다.

[0061] 서버(420)는 입력 문자열과 음악 데이터베이스(430)에 저장된 대표 문자열의 유사도를 연산한다(530). 입력 문자열과 대표 문자열의 유사도는 다양한 알고리즘을 통해 결정할 수 있다. 또한 전술한 수학적 식 1로 표현되는 거리값을 기준으로 입력 문자열과 대표 문자열의 유사도를 결정할 수도 있다.

[0062] 입력 문자열을 $X'[1, \dots, L]$ 이라고하면, 아래의 수학적 식 3을 이용하여 입력 문자열과 대표 문자열 S사이의 유사도를 연산할 수도 있다.

수학적 식 3

$$M(i, j) = \begin{cases} M(i-1, j-1) + w_{mat} & \text{if } S_i = X'_j \\ \max \begin{cases} M(i-1, j) \\ M(i, j-1) \\ M(i-1, j-1) + \alpha/w_{sub}(S_i, X'_j) \end{cases} & \text{otherwise} \end{cases}$$

[0064]

[0065] 여기서, S는 대표 문자열, X'는 입력 문자열, S_i 는 대표 문자열 S에서 i번째 문자, X'_j 는 문자열 X'에서 j번째 문자이다. w_{mat} 는 입력 문자열의 문자와 대표 문자열의 문자가 일치하는 경우 부여되는 값이다. $w_{sub}(S_i, X'_j)$ 는 문자의 순서 치환에 부여되는 가중치이고, α 는 상수값이다. 입력 문자열의 문자와 대표 문자열의 문자가 일치하는 경우 가장 높은 값을 부여하고, 입력 문자열과 대표 문자열이 일부 일치하는 경우 그 다음 높은 값을 부여한다.

[0067] 서버(420)는 입력 문자열과 가장 유사한 하나의 그룹을 결정할 수도 있겠지만, 유사도를 기준으로 유사도가 높

은 상위 몇 개의 그룹을 후보 그룹으로 결정할 수도 있다(540).

[0068] 도 5(b)에서는 입력 멜로디가 [19 20 21 22]인 예를 도시한다. 즉 도 5(b)는 알파벳이 아닌 숫자 형태로 멜로디를 표시한 경우이다. 서버(420)는 입력 문자열과 각 그룹의 대표 문자열의 유사도 연산을 통해 3개의 그룹(cluster)을 후보 그룹으로 결정하였다. 도 5(b)는 서버(420)가 그룹 2(cluster 2), 그룹 11(cluster 11) 및 그룹 1(cluster 1)을 후보 그룹으로 결정한 예를 도시하였다.

[0069] 대표 문자열 $S[1, \dots, n]$ 과 $X'[1, \dots, L]$ 이 매칭된다고 가정한다. 여기서 $n > L$ 이다. 서버(420)는 대표 문자열의 나머지 문자열을 이용하여 입력 문자열 이후의 멜로디를 추천할 수 있다. 이하 서버(420)가 입력 문자열에 대해 추천하는 멜로디를 후보 멜로디라고 명명한다. 서버(420)는 아래 수학식4의 조건을 만족하는 인덱스(index) k 를 결정할 수 있다.

수학식 4

$$M(k-1, L) < M(k, L) = M(k+1, L) = \dots = M(m, L)$$

[0073] 서버(420)는 후보 그룹의 대표 문자열에서 $S[k+1:n]$ 을 대상으로 입력 문자열에 대한 후보 멜로디를 추천할 수 있다(550). 예컨대, 도 5(b)에 도시한 후보 그룹을 살펴보면 입력 멜로디와 일치하는 구간($L=4$ 로 표시) 이후에 등장하는 구간에서 각 문자에 대한 빈도수를 연산할 수 있다. 도 5(b)에서 8은 3개의 그룹에서 모두 등장하였고, 6 및 7은 3개의 그룹 중 2개의 그룹(2/3)에서 등장하였다. 이와 같이 빈도수가 높은 문자를 후보 멜로디로 추천할 수 있다. 또한 서버(420)는 문자가 등장하는 순서를 고려하여 순서 및 빈도수를 기준으로 후보 멜로디를 추천할 수 있다.

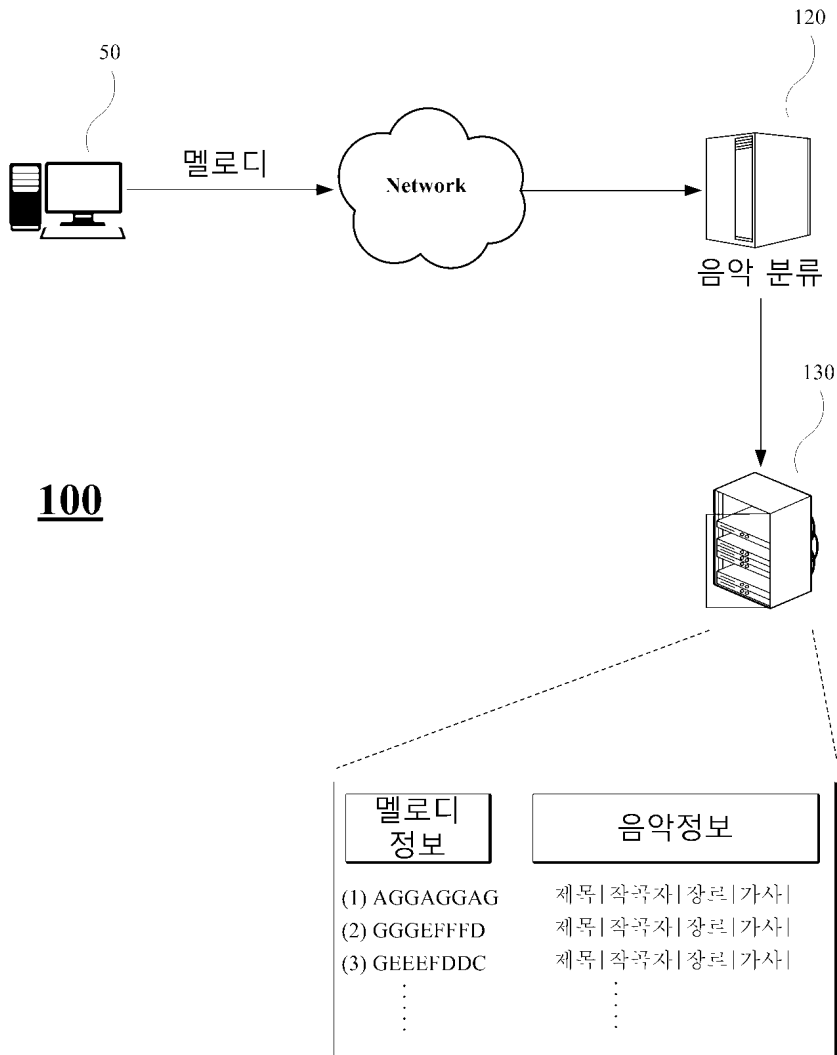
[0075] 본 실시예 및 본 명세서에 첨부된 도면은 전술한 기술에 포함되는 기술적 사상의 일부를 명확하게 나타내고 있는 것에 불과하며, 전술한 기술의 명세서 및 도면에 포함된 기술적 사상의 범위 내에서 당업자가 용이하게 유추할 수 있는 변형 예와 구체적인 실시예는 모두 전술한 기술의 권리범위에 포함되는 것이 자명하다고 할 것이다.

부호의 설명

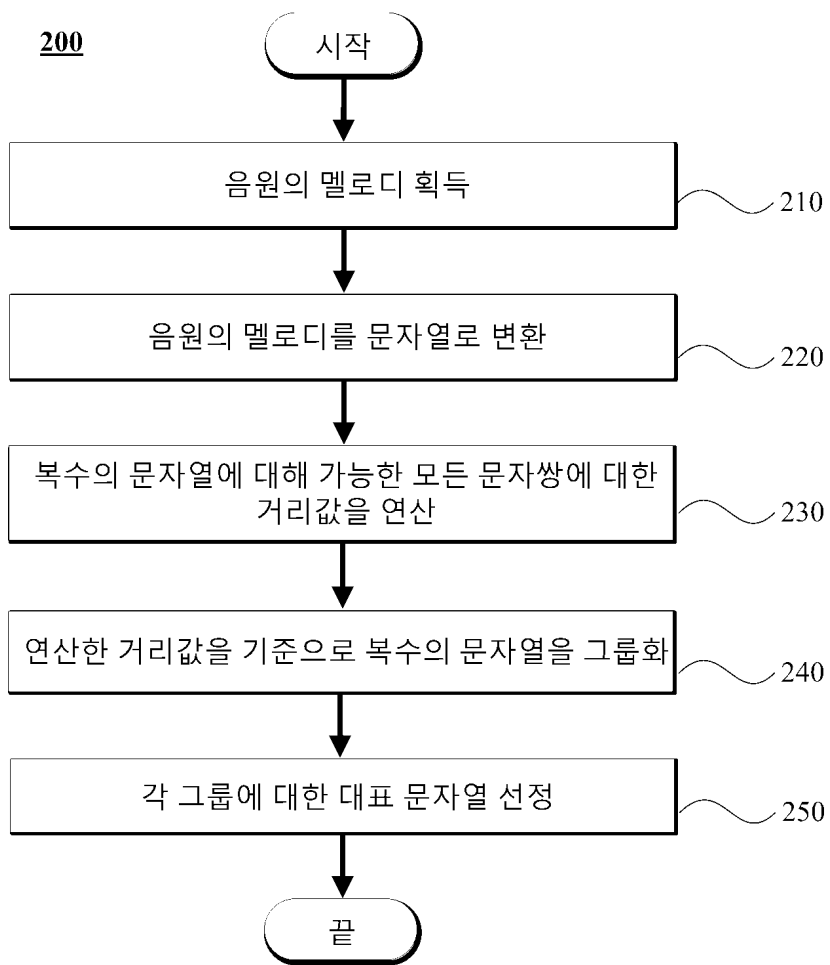
[0076] 50 : 컴퓨터
80 : 휴대용 단말기
100 : 음악 데이터베이스를 생성하는 시스템
120 : 서버
130 : 음악 데이터베이스
300 : 음악 검색 시스템
320 : 서버
330 : 음악 데이터베이스
400 : 음악 추천 시스템
420 : 서버
430 : 음악 데이터베이스

도면

도면1



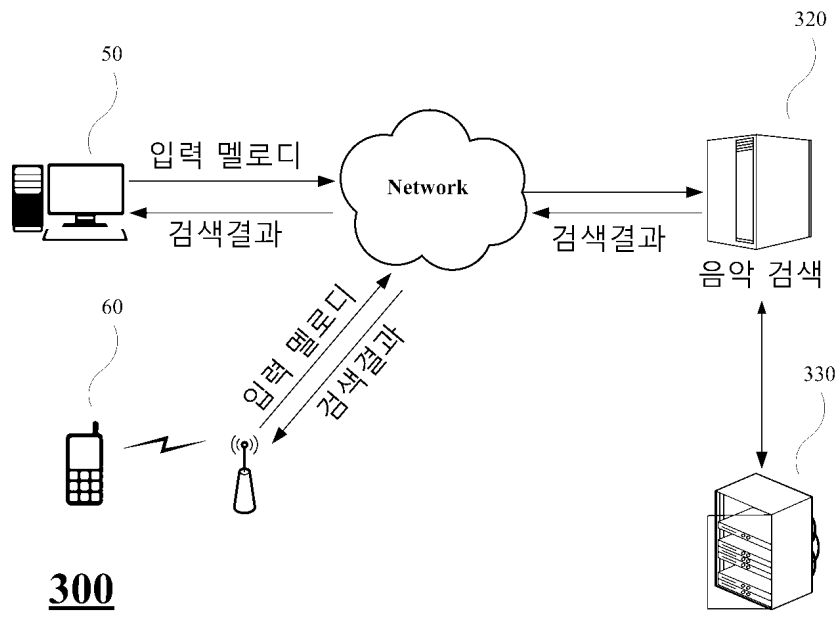
도면2



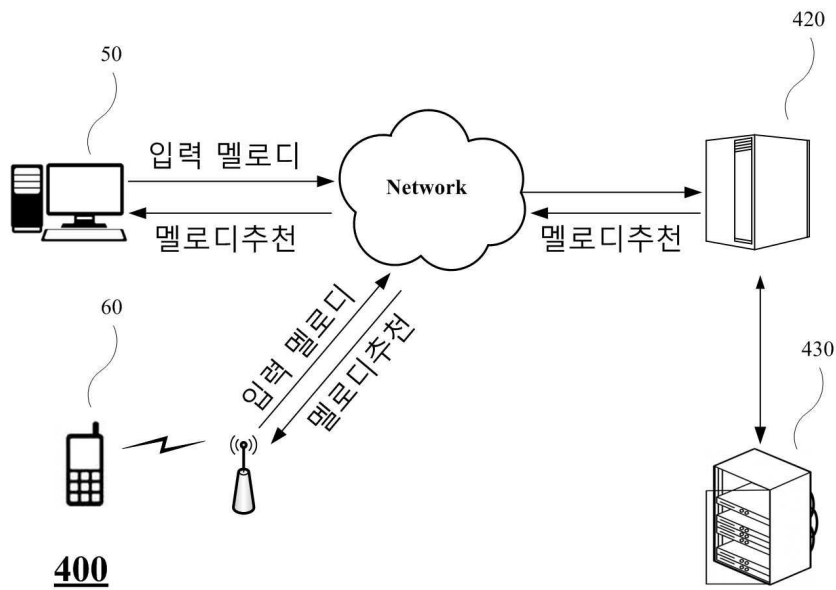
도면3

그룹	대표문자열	문자열	음악정보
그룹 1	(1) AGGAGAAG	(1-1) BDCGAGGAGGAGGCCG... (1-2) AGGAGGAGGCBBCGC... (1-3) BACGGEAAGGAGGAG... ...	(1-1) 제 목 작곡자 장르 가사 (1-2) 제 목 작곡자 장르 가사 (1-3) 제 목 작곡자 장르 가사 ...
그룹 2	(2) CGCCGEDG	(2-1) BDCGAGCGCCGEDGCG... (2-2) ACGCCGEDGGCBBCGC... (2-3) BACGGEACGCCGEDGG... ...	(2-1) 제 목 작곡자 장르 가사 (2-2) 제 목 작곡자 장르 가사 (2-3) 제 목 작곡자 장르 가사 ...
그룹 3	(3) EDAEADFD	(3-1) BDCGAGCGEDAEADFD... (3-2) EDAEADFDGGCBBCGC... (3-3) BACEDAEADFDGEDGG... ...	(3-1) 제 목 작곡자 장르 가사 (3-2) 제 목 작곡자 장르 가사 (3-3) 제 목 작곡자 장르 가사 ...
...	...	...	...

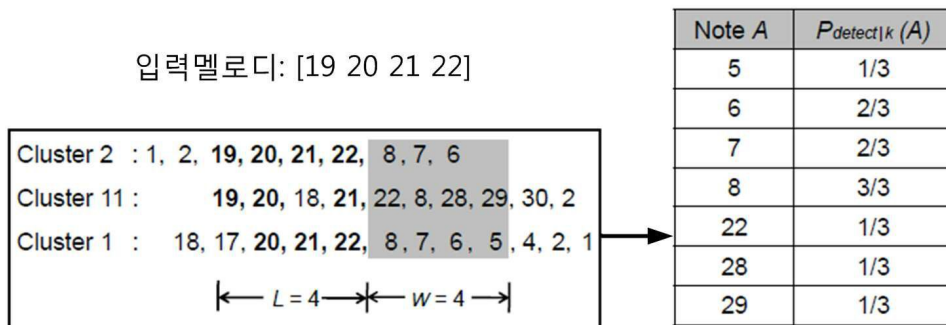
도면4



도면5



(a)



(b)

도면6

