(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년03월30일
(11) 등록번호 10-2095091
(24) 등록일자 2020년03월24일

(51) 국제특허분류(Int. Cl.)
G10L 19/038 (2013.01) *G10L 19/002* (2013.01)
G10L 19/008 (2014.01) *G10L 19/08* (2006.01)
H04S 3/00 (2006.01) *H04S 7/00* (2006.01)
(52) CPC특허분류
G10L 19/038 (2013.01)
G10L 19/002 (2013.01)
(21) 출원번호 10-2017-7018248(분할)
(22) 출원일자(국제) 2015년01월30일
심사청구일자 2020년01월15일
(85) 번역문제출일자 2017년06월30일
(65) 공개번호 10-2017-0081296
(43) 공개일자 2017년07월11일
(62) 원출원 특허 10-2016-7023092
원출원일자(국제) 2015년01월30일
심사청구일자 2017년03월06일
(86) 국제출원번호 PCT/US2015/013818
(87) 국제공개번호 WO 2015/116952
국제공개일자 2015년08월06일
(30) 우선권주장
61/933,706 2014년01월30일 미국(US)
(뒷면에 계속)

(56) 선행기술조사문헌
Andrew Wabnitz, et al. Time domain reconstruction of spatial sound fields using compressed sensing. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2011. pp.465-468.
US20130028427 A1
KR1020120070521 A
KR1020140000240 A

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
페터스 닐스 귄터
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
센 디판잔
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리아나

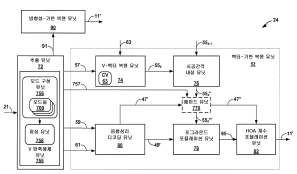
전체 청구항 수 : 총 30 항 심사관 : 이남숙

(54) 발명의 명칭 **벡터들을 코딩하기 위한 프레임 파라미터 재사용성의 표시**

(57) 요 약

일반적으로, 벡터들을 디코딩하기 위한 프레임 파라미터 재사용성을 표시하는 기법들이 설명된다. 프로세서 및 메모리를 포함하는 디바이스가 본 기법들을 수행할 수도 있다. 프로세서는 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터를 포함하는 비트스트림을 획득하도록 구성될 수도 있다. 비트스트림은 벡터를 압축할 때 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자를 더 포함할 수도 있다. 메모리는 비트스트림을 저장하도록 구성될 수도 있다.

대 표 도



(52) CPC특허분류
G10L 19/008 (2013.01)
G10L 19/08 (2013.01)
H04S 3/002 (2013.01)
H04S 7/30 (2013.01)
H04S 2420/11 (2013.01)

(30) 우선권주장
61/933,714 2014년01월30일 미국(US)
61/933,731 2014년01월30일 미국(US)
61/949,591 2014년03월07일 미국(US)
61/949,583 2014년03월07일 미국(US)
61/994,794 2014년05월16일 미국(US)
62/004,147 2014년05월28일 미국(US)
62/004,067 2014년05월28일 미국(US)
62/004,128 2014년05월28일 미국(US)
62/019,663 2014년07월01일 미국(US)
62/027,702 2014년07월22일 미국(US)
62/028,282 2014년07월23일 미국(US)
62/029,173 2014년07월25일 미국(US)
62/032,440 2014년08월01일 미국(US)
62/056,248 2014년09월26일 미국(US)
62/056,286 2014년09월26일 미국(US)
62/102,243 2015년01월12일 미국(US)
14/609,190 2015년01월29일 미국(US)

명 세 서

청구범위

청구항 1

비트스트림을 프로세싱하기 위한 디바이스로서,

상기 디바이스는,

음장의 공간 구성요소의 압축된 버전을 포함하는 상기 비트스트림을 획득하는 것으로서, 상기 음장의 상기 공간 구성요소는 구면 고조파들 도메인에서의 벡터에 의해 표현되고, 현재 프레임에 대한 신택스 엘리먼트의 값은 특정 허프만 코드북을 결정하는 인덱스를 나타내며, 상기 비트스트림은 표시자를 더 포함하고, 특정 값을 갖는 상기 표시자는, 상기 비트스트림이 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값은 이전 프레임에 대한 상기 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 상기 비트스트림을 획득하는 것을 행하고;

상기 벡터와 연관된 데이터를 코딩하기 위해 상기 특정 허프만 코드북을 사용하도록

구성된 하나 이상의 프로세서들; 및

상기 하나 이상의 프로세서들에 커플링되고 상기 비트스트림을 저장하도록 구성된 메모리를 포함하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 2

제 1 항에 있어서,

상기 표시자는 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값의 하나 이상의 비트들을 포함하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 3

제 2 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 표시자는 상기 현재 프레임에 대한 제 2 신택스 엘리먼트의 값 및 상기 현재 프레임에 대한 제 3 신택스 엘리먼트의 값을 포함하며, 그리고

상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값 플러스 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값이 제로와 동일한 것은, 상기 비트스트림이 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 1 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 4

제 2 항에 있어서,

상기 표시자는 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 최상위 비트 및 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 제 2 최상위 비트를 포함하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 5

제 1 항에 있어서,

상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값은, 5 보다 더 큰 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값에 기초하여 상기 특정 허프만 코드북을 나타내는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 6

제 5 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고, 그리고

6 에서부터 15 까지의 상기 제 1 신택스 엘리먼트의 각각의 허용가능한 값은 5 개의 허프만 코드북들의 각각의 세트와 연관되며;

상기 특정 값을 갖는 상기 표시자는, 상기 비트스트림이 상기 현재 프레임에 대한 제 2 신택스 엘리먼트의 값을 포함하지 않고, 상기 비트스트림이 상기 현재 프레임에 대한 제 3 신택스 엘리먼트의 값을 포함하지 않으며, 상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 2 신택스 엘리먼트의 값과 동일하고, 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 3 신택스 엘리먼트의 값과 동일하다는 것을 나타내고,

상기 제 2 신택스 엘리먼트는, 상기 벡터에 대해 예측이 수행되었는지 여부를 나타내고,

상기 제 3 신택스 엘리먼트는, 상기 비트스트림에서 시그널링된 상기 제 1 신택스 엘리먼트의 값과 연관된 5 개의 허프만 코드북들의 상기 세트로부터 특정 허프만 코드북을 선택하기 위해 사용되는 추가적인 허프만 코드북 정보를 나타내며, 그리고,

상기 하나 이상의 프로세서들은, 상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값 및 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값에 기초하여, 상기 비트스트림에서 시그널링된 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값과 연관된 5 개의 허프만 코드북들의 상기 세트 중에서 상기 특정 허프만 코드북을 결정하도록 더 구성되고; 그리고

상기 하나 이상의 프로세서들은, 상기 벡터와 연관된 상기 데이터를 코딩하기 위해 상기 특정 허프만 코드북을 이용하는 것의 부분으로서, 상기 하나 이상의 프로세서들이 상기 벡터의 적어도 하나의 벡터 엘리먼트를 코딩하기 위해 상기 특정 허프만 코드북을 이용하도록, 구성되는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 7

제 1 항에 있어서,

상기 하나 이상의 프로세서들은,

상기 벡터를 획득하기 위해 고차 앰비소닉 오디오 데이터를 분해하고; 그리고

상기 비트스트림을 획득하기 위해 상기 비트스트림에서 상기 벡터를 규정하도록 더 구성되는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 8

제 1 항에 있어서,

상기 하나 이상의 프로세서들은,

상기 비트스트림으로부터, 상기 벡터에 대응하는 오디오 오브젝트를 획득하고; 그리고

상기 오디오 오브젝트를 상기 벡터와 결합하여 고차 앰비소닉 (HOA) 오디오 데이터를 복원하도록 더 구성되는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 9

제 8 항에 있어서,

상기 하나 이상의 프로세서들은, 하나 이상의 라우드스피커 피드들을 출력하기 위해 HOA 계수들을 렌더링하도록 구성되고,

상기 디바이스는 하나 이상의 라우드스피커들에 커플링되고, 상기 하나 이상의 라우드스피커 피드들은 상기 하나 이상의 라우드스피커들을 구동하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 10

제 1 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 하나 이상의 프로세서들은,

상기 특정 값을 갖지 않는 상기 표시자에 기초하여, 상기 비트스트림으로부터 제 2 신택스 엘리먼트를 획득하도록 더 구성되고, 상기 제 2 신택스 엘리먼트는 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 최하위 비트들을 나타내는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 11

비트스트림을 프로세싱하는 방법으로서,

음장의 공간 구성요소의 압축된 버전을 포함하는 상기 비트스트림을 획득하는 단계로서, 상기 음장의 상기 공간 구성요소는 구면 고조파들 도메인에서의 벡터에 의해 표현되고, 현재 프레임에 대한 신택스 엘리먼트의 값은 특정 허프만 코드북을 결정하는 인덱스를 나타내며, 상기 비트스트림은 표시자를 더 포함하고, 특정 값을 갖는 상기 표시자는, 상기 비트스트림이 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값은 이전 프레임에 대한 상기 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 상기 비트스트림을 획득하는 단계;

상기 벡터와 연관된 데이터를 코딩하기 위해 상기 특정 허프만 코드북을 사용하는 단계; 및

상기 비트스트림을 저장하는 단계를 포함하는, 비트스트림을 프로세싱하는 방법.

청구항 12

제 11 항에 있어서,

상기 표시자는 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값의 하나 이상의 비트들을 포함하는, 비트스트림을 프로세싱하는 방법.

청구항 13

제 12 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 표시자는 상기 현재 프레임에 대한 제 2 신택스 엘리먼트의 값 및 상기 현재 프레임에 대한 제 3 신택스 엘리먼트의 값을 포함하며, 그리고

상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값 플러스 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값이 제로와 동일한 것은, 상기 비트스트림이 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 1 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 비트스트림을 프로세싱하는 방법.

청구항 14

제 12 항에 있어서,

상기 표시자는 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 최상위 비트 및 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 제 2 최상위 비트를 포함하는, 비트스트림을 프로세싱하는 방법.

청구항 15

제 11 항에 있어서,

상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값은, 5 보다 더 큰 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값에 기초하여 상기 특정 허프만 코드북을 나타내는, 비트스트림을 프로세싱하는 방법.

청구항 16

제 15 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고, 그리고

6 에서부터 15 까지의 상기 제 1 신택스 엘리먼트의 각각의 허용가능한 값은 5 개의 허프만 코드북들의 각각의 세트와 연관되며;

상기 특정 값을 갖는 상기 표시자는, 상기 비트스트림이 상기 현재 프레임에 대한 제 2 신택스 엘리먼트의 값을 포함하지 않고, 상기 비트스트림이 상기 현재 프레임에 대한 제 3 신택스 엘리먼트의 값을 포함하지 않으며, 상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 2 신택스 엘리먼트의 값과 동일하고, 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 3 신택스 엘리먼트의 값과 동일하다는 것을 나타내고,

상기 제 2 신택스 엘리먼트는, 상기 벡터에 대해 예측이 수행되었는지 여부를 나타내고,

상기 제 3 신택스 엘리먼트는, 상기 비트스트림에서 시그널링된 상기 제 1 신택스 엘리먼트의 값과 연관된 5 개의 허프만 코드북들의 상기 세트로부터 특정 허프만 코드북을 선택하기 위해 사용되는 추가적인 허프만 코드북 정보를 나타내며, 그리고,

상기 방법은, 상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값 및 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값에 기초하여, 상기 비트스트림에서 시그널링된 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값과 연관된 5 개의 허프만 코드북들의 상기 세트 중에서 상기 특정 허프만 코드북을 결정하는 단계를 더 포함하고; 그리고

상기 벡터와 연관된 상기 데이터를 코딩하기 위해 상기 특정 허프만 코드북을 이용하는 것은, 상기 벡터의 적어도 하나의 벡터 엘리먼트를 코딩하기 위해 상기 특정 허프만 코드북을 이용하는 것을 포함하는, 비트스트림을 프로세싱하는 방법.

청구항 17

제 11 항에 있어서,

상기 벡터를 획득하기 위해 고차 앰비소닉 오디오 데이터를 분해하는 단계; 및

상기 비트스트림을 획득하기 위해 상기 비트스트림에서 상기 벡터를 규정하는 단계를 더 포함하는, 비트스트림을 프로세싱하는 방법.

청구항 18

제 11 항에 있어서,

상기 비트스트림으로부터, 상기 벡터에 대응하는 오디오 오브젝트를 획득하는 단계; 및

상기 오디오 오브젝트를 상기 벡터와 결합하여 고차 앰비소닉 (HOA) 오디오 데이터를 복원하는 단계를 더 포함하는, 비트스트림을 프로세싱하는 방법.

청구항 19

제 18 항에 있어서,

하나 이상의 라우드스피커 피드들을 출력하기 위해 HOA 계수들을 렌더링하는 단계를 더 포함하고,

상기 하나 이상의 라우드스피커 피드들을 출력하기 위해 상기 HOA 계수들을 렌더링하는 디바이스가 하나 이상의 라우드스피커들에 커플링되고, 상기 하나 이상의 라우드스피커 피드들은 상기 하나 이상의 라우드스피커들을 구동하는, 비트스트림을 프로세싱하는 방법.

청구항 20

제 11 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 방법은,

상기 특정 값을 갖지 않는 상기 표시자에 기초하여, 상기 비트스트림으로부터 제 2 신택스 엘리먼트를 획득하는 단계를 더 포함하고, 상기 제 2 신택스 엘리먼트는 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 최하위 비트들을 나타내는, 비트스트림을 프로세싱하는 방법.

청구항 21

비트스트림을 프로세싱하기 위한 디바이스로서,

음장의 공간 구성요소의 압축된 버전을 포함하는 상기 비트스트림을 획득하는 수단으로서, 상기 음장의 상기 공간 구성요소는 구면 고조파들 도메인에서의 벡터에 의해 표현되고, 현재 프레임에 대한 신택스 엘리먼트의 값은 특정 허프만 코드북을 결정하는 인덱스를 나타내며, 상기 비트스트림은 표시자를 더 포함하고, 특정 값을 갖는 상기 표시자는, 상기 비트스트림이 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값은 이전 프레임에 대한 상기 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 상기 비트스트림을 획득하는 수단;

상기 벡터와 연관된 데이터를 코딩하기 위해 상기 특정 허프만 코드북을 사용하는 수단; 및

상기 비트스트림을 저장하는 수단을 포함하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 22

제 21 항에 있어서,

상기 표시자는 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값의 하나 이상의 비트들을 포함하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 23

제 21 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 표시자는 상기 현재 프레임에 대한 제 2 신택스 엘리먼트의 값 및 상기 현재 프레임에 대한 제 3 신택스 엘리먼트의 값을 포함하며, 그리고

상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값 플러스 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값이 제로와 동일한 것은, 상기 비트스트림이 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 1 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 24

제 21 항에 있어서,

상기 벡터를 획득하기 위해 고차 앰비소닉 오디오 데이터를 분해하는 수단; 및

상기 비트스트림을 획득하기 위해 상기 비트스트림에서 상기 벡터를 규정하는 수단을 더 포함하는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 25

제 21 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 디바이스는,

상기 특정 값을 갖지 않는 상기 표시자에 기초하여, 상기 비트스트림으로부터 제 2 신택스 엘리먼트를 획득하는 수단을 더 포함하고, 상기 제 2 신택스 엘리먼트는 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의

최하위 비트들을 나타내는, 비트스트림을 프로세싱하기 위한 디바이스.

청구항 26

명령들을 저장한 비일시성 컴퓨터-판독가능 저장 매체로서,

상기 명령들은, 실행될 때, 디바이스를,

음장의 공간 구성요소의 압축된 버전을 포함하는 비트스트림을 획득하는 것으로서, 상기 음장의 상기 공간 구성요소는 구면 고조파들 도메인에서의 벡터에 의해 표현되고, 현재 프레임에 대한 신택스 엘리먼트의 값은 특정 허프만 코드북을 결정하는 인덱스를 나타내며, 상기 비트스트림은 표시자를 더 포함하고, 특정 값을 갖는 상기 표시자는, 상기 비트스트림이 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값은 이전 프레임에 대한 상기 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 상기 비트스트림을 획득하는 것을 행하고;

상기 벡터와 연관된 데이터를 코딩하기 위해 상기 특정 허프만 코드북을 사용하며; 그리고

상기 비트스트림을 저장하도록

구성하는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 27

제 26 항에 있어서,

상기 표시자는 상기 현재 프레임에 대한 상기 신택스 엘리먼트의 값의 하나 이상의 비트들을 포함하는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 28

제 26 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 표시자는 상기 현재 프레임에 대한 제 2 신택스 엘리먼트의 값 및 상기 현재 프레임에 대한 제 3 신택스 엘리먼트의 값을 포함하며, 그리고

상기 현재 프레임에 대한 상기 제 2 신택스 엘리먼트의 값 플러스 상기 현재 프레임에 대한 상기 제 3 신택스 엘리먼트의 값이 제로와 동일한 것은, 상기 비트스트림이 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값을 포함하지 않고 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값은 상기 이전 프레임에 대한 상기 제 1 신택스 엘리먼트의 값과 동일하다는 것을 나타내는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 29

제 26 항에 있어서,

상기 명령들은, 실행될 때, 상기 디바이스를,

상기 벡터를 획득하기 위해 고차 앰비소닉 오디오 데이터를 분해하고; 그리고

상기 비트스트림을 획득하기 위해 상기 비트스트림에서 상기 벡터를 규정하도록

더 구성하는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 30

제 26 항에 있어서,

상기 신택스 엘리먼트는 제 1 신택스 엘리먼트이고,

상기 명령들은, 실행될 때, 상기 디바이스를,

상기 특정 값을 갖지 않는 상기 표시자에 기초하여, 상기 비트스트림으로부터 제 2 신택스 엘리먼트를 획득하도록 더 구성하고, 상기 제 2 신택스 엘리먼트는 상기 현재 프레임에 대한 상기 제 1 신택스 엘리먼트의 값의 최

하위 비트들을 나타내는, 비일시성 컴퓨터-판독가능 저장 매체.

발명의 설명

기 술 분 야

[0001] 본 출원은 다음 미국 가출원들: "COMPRESSION OF DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로 2014년 1월 30일에 출원된 미국 가출원 번호 제 61/933,706호; "COMPRESSION OF DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로 2014년 1월 30일에 출원된 미국 가출원 번호 제 61/933,714호; "INDICATING FRAME PARAMETER REUSABILITY FOR DECODING SPATIAL VECTORS"란 발명의 명칭으로 2014년 1월 30일에 출원된 미국 가출원 번호 제 61/933,731호; "IMMEDIATE PLAY-OUT FRAME FOR SPHERICAL HARMONIC COEFFICIENTS"란 발명의 명칭으로 2014년 3월 7일에 출원된 미국 가출원 번호 제 61/949,591호; "FADE-IN/FADE-OUT OF DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로 2014년 3월 7일에 출원된 미국 가출원 번호 제 61/949,583호; "CODING V-VECTORS OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 5월 16일에 출원된 미국 가출원 번호 제 61/994,794호; "INDICATING FRAME PARAMETER REUSABILITY FOR DECODING SPATIAL VECTORS"란 발명의 명칭으로 2014년 5월 28일에 출원된 미국 가출원 번호 제 62/004,147호; "IMMEDIATE PLAY-OUT FRAME FOR SPHERICAL HARMONIC COEFFICIENTS AND FADE-IN/FADE-OUT OF DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로 2014년 5월 28일에 출원된 미국 가출원 번호 제 62/004,067호; "CODING V-VECTORS OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 5월 28일에 출원된, 미국 가출원 번호 제 62/004,128호; "CODING V-VECTORS OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 7월 1일에 출원된, 미국 가출원 번호 제 62/019,663호; "CODING V-VECTORS OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 7월 22일에 출원된, 미국 가출원 번호 제 62/027,702호; "CODING V-VECTORS OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 7월 23일에 출원된, 미국 가출원 번호 제 62/028,282호; "IMMEDIATE PLAY-OUT FRAME FOR SPHERICAL HARMONIC COEFFICIENTS AND FADE-IN/FADE-OUT OF DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로 2014년 7월 25일에 출원된 미국 가출원 번호 제 62/029,173호; "CODING V-VECTORS OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 8월 1일에 출원된, 미국 가출원 번호 제 62/032,440호; "SWITCHED V-VECTOR QUANTIZATION OF A HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 9월 26일에 출원된 미국 가출원 번호 제 62/056,248호; "PREDICTIVE VECTOR QUANTIZATION OF A DECOMPOSED HIGHER ORDER AMBISONICS (HOA) AUDIO SIGNAL"란 발명의 명칭으로 2014년 9월 26일에 출원된 미국 가출원 번호 제 62/056,286호; 및 "TRANSITIONING OF AMBIENT HIGHER-ORDER AMBISONIC COEFFICIENTS"란 발명의 명칭으로 2015년 1월 12일에 출원된 미국 가출원 번호 제 62/102,243호의 이익을 주장하며, 전술한 리스트된 미국 가출원들의 각각이 그들 각각의 전체로 본원에서 개시된 것처럼 참조로 포함된다.

[0002] 기술 분야

[0003] 본 개시물은 오디오 데이터, 좀더 구체적으로는, 고-차수 앰비소닉 오디오 데이터의 코딩에 관한 것이다.

배 경 기 술

[0004] 고차 앰비소닉스 (higher-order ambisonics; HOA) 신호 (종종 복수의 구면 고조파 계수들 (spherical harmonic coefficients; SHC) 또는 다른 계층적 엘리먼트들에 의해 표현됨) 는 음장의 3차원의 표현이다. HOA 또는 SHC 표현은 음장을, SHC 신호로부터 렌더링되는 멀티-채널 오디오 신호를 플레이백하는데 사용되는 로컬 스피커 기하학적 구조 (local speaker geometry) 와 독립적 방법으로 표현할 수도 있다. SHC 신호는 또한 SHC 신호가 5.1 오디오 채널 포맷 또는 7.1 오디오 채널 포맷과 같은, 널리 공지된 그리고 많이 채택된 멀티-채널 포맷들로 렌더링될 수도 있기 때문에, 역방향 호환성 (backwards compatibility) 을 용이하게 할 수도 있다. 따라서 SHC 표현은 역방향 호환성을 또한 수용하는 더 나은 음장의 표현을 가능하게 할 수도 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0005] 일반적으로, 고-차수 앰비소닉스 오디오 데이터의 코딩을 위한 기법들이 설명된다. 고-차수 앰비소닉스 오디오 데이터는 1보다 큰 차수를 가지는 구면 고조파 기저 함수에 대응하는 적어도 하나의 구면 고조파 계수를 포함할 수도 있다.

[0006] 일 양태에서, 효율적인 비트 사용의 방법은 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터를 포함하는 비트스트림을 획득하는 단계를 포함한다. 비트스트림은 벡터를 압축하는데 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자를 더 포함한다.

[0007] 다른 양태에서, 효율적인 비트 사용을 수행하도록 구성된 디바이스는 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터를 포함하는 비트스트림을 획득하도록 구성된 하나 이상의 프로세서들을 포함한다. 비트스트림은 벡터를 압축하는데 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자를 더 포함한다. 본 디바이스는 비트스트림을 저장하도록 구성된 메모리를 포함한다.

[0008] 다른 양태에서, 효율적인 비트 사용을 수행하도록 구성된 디바이스는 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터를 포함하는 비트스트림을 획득하는 수단을 포함한다. 비트스트림은 벡터를 압축하는데 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자를 더 포함한다. 디바이스는 또한 표시자를 저장하는 수단을 포함한다.

[0009] 다른 양태에서, 비일시성 컴퓨터-판독가능 저장 매체는, 실행될 때, 하나 이상의 프로세서들로 하여금, 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터를 포함하는 비트스트림을 획득하도록 하는 명령들을 저장하고 있으며, 상기 비트스트림은 벡터를 압축하는데 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자를 더 포함한다.

[0010] 본 기법들의 하나 이상의 양태들의 세부 사항들은 첨부도면 및 아래의 상세한 설명에서 개시된다. 본 기법들의 다른 특징들, 목적들, 및 이점들은 설명 및 도면들로부터, 그리고 청구범위로부터 명백히 알 수 있을 것이다.

도면의 간단한 설명

[0011] 도 1 은 여러 차수들 및 하위-차수들의 구면 고조파 (spherical harmonic) 기저 함수들을 예시하는 다이어그램이다.

도 2 는 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 수도 있는 시스템을 예시하는 다이어그램이다.

도 3 은 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 수도 있는 도 2 의 예에 나타낸 오디오 인코딩 디바이스의 일 예를 좀더 자세하게 예시하는 블록도이다.

도 4 는 도 2 의 오디오 디코딩 디바이스를 좀더 자세하게 예시하는 블록도이다.

도 5a 는 본 개시물에서 설명되는 벡터-기반 합성 기법들의 여러 양태들을 수행할 때에 오디오 인코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다.

도 5b 는 본 개시물에서 설명되는 코딩 기법들의 여러 양태들을 수행할 때에 오디오 인코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다.

도 6a 는 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 때에 오디오 디코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다.

도 6b 는 본 개시물에서 설명되는 코딩 기법들의 여러 양태들을 수행할 때에 오디오 디코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다.

도 7 은 압축된 공간 구성요소들을 규정하는 비트스트림의 프레임들을 좀더 자세하게 예시하는 다이어그램이다.

도 8 은 압축된 공간 구성요소들을 규정할 수도 있는 비트스트림 또는 부 채널 정보의 부분을 좀더 자세하게 예시하는 다이어그램이다.

발명을 실시하기 위한 구체적인 내용

[0012] 오늘날 서라운드 사운드의 발전은 엔터테인먼트에 대한 많은 출력 포맷들을 이용가능하게 하였다. 이러한 소비자 서라운드 사운드 포맷들의 예들은 그들이 라우드스피커들에의 공급들을 어떤 기하학적인 좌표들로 암시적으로 규정한다는 점에서 주로 '채널' 기반이다. 소비자 서라운드 사운드 포맷들은 (다음 6개의 채널들: 전면 좌측 (FL), 전면 우측 (FR), 중심 또는 전면 중앙, 후면 좌측 또는 서라운드 좌측, 후면 우측 또는 서라운드 우측, 및 저주파수 효과들 (LFE) 을 포함하는) 인기 있는 5.1 포맷, 성장하는 7.1 포맷, 및 (예컨대, 초고화질 텔레비전 표준 (Ultra High Definition Television standard) 과 함께 사용하기 위한) 22.2 포맷 및 7.1.4 포맷과 같은, 높이 스피커들을 포함하는 다양한 포맷들을 포함한다. 비-소비자 포맷들은 '서라운드 어레이들' 로서 종종 불리는 임의 개수의 스피커들을 (대칭 및 비-대칭 기하학적 구조들로) 포괄할 수 있다. 이러한 어레이의 일 예는 트렁케이트된 (truncated) 20면체의 모서리들 상의 좌표들 상에 위치되는 32 개의 라우드스피커들을 포함한다.

[0013] 미래 MPEG 인코더에의 입력은 옵션적으로 다음 3개의 가능한 포맷들 중 하나이다: (i) (위에서 설명한 바와 같이) 사전-규정된 위치들에서 라우드스피커들을 통해서 플레이되어야 하는 전통적인 채널-기반의 오디오; (ii) (다른 정보 중에서) 그들의 로케이션 좌표들을 포함하는 연관된 메타데이터를 가진 단일 오디오 오브젝트들에 대한 이산 펄스-코드-변조 (PCM) 데이터를 수반하는 오브젝트-기반의 오디오; 및 (iii) 구면 고조파 기저 함수들의 계수들 (또한, "구면 고조파 계수들", 또는 SHC, "고-차수 앰비소닉스" 또는 HOA, 및 "HOA 계수들" 라 함) 을 이용하여 음장을 표현하는 것을 수반하는 장면-기반의 오디오. 미래 MPEG 인코더는 2013년 1월, 스위스, 제네바에서 배포되며, 그리고 http://mpeg.chiariglione.org/sites/default/files/files/standards/parts/docs/w13411.zip 에서 입수가능한, ISO/IEC (International Organization for Standardization/International Electrotechnical Commission) JTC1/SC29/WG11/N13411 에 의한, "Call for Proposals for 3D Audio" 란 제목으로 된 문서에 좀더 자세히 설명되었을 수도 있다.

[0014] 시장에서는 여러 '서라운드-사운드' 채널-기반 포맷들이 있다. 그들은 예를 들어, (스테레오를 넘어서 거실들로 잠식해 들어가는 관점에서 가장 성공적이었던) 5.1 홈 시어터 시스템으로부터, NHK (Nippon Hoso Kyokai 또는 일본 방송 협회 (Japan Broadcasting Corporation)) 에 의해 개발된 22.2 시스템에 이른다. 콘텐츠 생성자들 (예컨대, 할리우드 스튜디오들) 은 영화용 사운드트랙을 한번 제작하고, 각각의 스피커 구성을 위해 그것을 재믹싱하는데 노력을 들이지 않기를 원할 것이다. 최근, 표준들 개발 조직들은 표준화된 비트스트림으로의 인코딩, 및 스피커 기하학적 구조 (및 개수) 및 (렌더러를 포함한) 플레이백의 로케이션에서의 음향 조건들에 적응가능하고 독립적인 후속 디코딩을 제공할 방법들을 고려하고 있다.

[0015] 콘텐츠 생성자들에게 이러한 유연성을 제공하기 위해, 음장을 표현하는데 엘리먼트들의 계층적 세트가 사용될 수도 있다. 엘리먼트들의 계층적 세트는 낮은-차수의 엘리먼트들의 기본적인 세트가 모델링된 음장의 풀 표현을 제공하도록 엘리먼트들이 차수화된 엘리먼트들의 세트를 지칭할 수도 있다. 그 세트가 고-차수 엘리먼트들을 포함하도록 확장되므로, 그 표현이 좀더 상세해져, 해상도를 증가시킨다.

[0016] 엘리먼트들의 계층적 세트의 일 예는 구면 고조파 계수들의 세트 (SHC) 이다. 다음 수식은 음장의 설명 또는 표현을 SHC 를 이용하여 설명한다:

[0017] $$p_i(t, r_r, \theta_r, \varphi_r) = \sum_{\omega=0}^{\infty} \left[4\pi \sum_{n=0}^{\infty} j_n(kr_r) \sum_{m=-n}^{n} A_n^m(k)\, Y_n^m(\theta_r, \varphi_r) \right] e^{j\omega t},$$

[0018] 수식은 시간 t 에서 음장의 임의의 지점 $\{r_r, \theta_r, \varphi_r\}$ 에서의 압력 p_i 이, SHC, $A_n^m(k)$ 에 의해 고유하게 표현될 수 있다는 것을 나타낸다. 여기서, k=ω/c, c 는 사운드의 속도 (~343 m/s) 이고, $\{r_r, \theta_r, \varphi_r\}$ 는 참조의 지점 (또는, 관측 지점) 이고, $j_n(\cdot)$ 는 차수 n 의 구면 Bessel 함수이고, 그리고 $Y_n^m(\theta_r, \varphi_r)$ 는 차수 n 및 하위차수 m 의 구면 고조파 기저 함수들이다. 꺽쇠 괄호들 내 항은 이산 푸리에 변환 (DFT), 이산 코사인 변환 (DCT), 또는 웨이블릿 변환과 같은, 여러 시간-주파수 변환들에 의해 근사화될 수 있는 신호의 주파수-도메인 표현 (즉, $S(\omega, r_r, \theta_r, \varphi_r)$) 인 것을 알 수 있다. 계층적 세트들의 다른 예들은 웨이블릿 변환 계수들의 세트들 및 다중해상도 기저 함수들의 계수들의 다른 세트들을 포함한다.

[0019] 도 1 은 제로 차수 (n = 0) 로부터 제 4 차수 (n = 4) 까지의 구면 고조파 기저 함수들을 예시하는 다이어그램

이다. 볼 수 있는 바와 같이, 각각의 차수에 대해, 예시의 용이 목적을 위해 도 1 의 예에 나타내지만 명시적으로 표시되지 않은 하위차수들 m 의 확장이 존재한다.

[0020] SHC $A_n^m(k)$ 는 여러 마이크로폰 어레이 구성들에 의해 물리적으로 획득될 (예컨대, 기록될) 수 있거나, 또는 이의 대안으로, 그들은 음장의 채널-기반의 또는 오브젝트-기반의 설명들로부터 유도될 수 있다. SHC 는 장면-기반의 오디오를 나타내며, 여기서, SHC 는 좀더 효율적인 송신 또는 저장을 증진할 수도 있는 인코딩된 SHC 를 획득하기 위해 오디오 인코더에 입력될 수도 있다. 예를 들어, $(1+4)^2$ (25, 따라서, 제 4 차수) 계수들을 수반하는 제 4-차수 표현이 사용될 수도 있다.

[0021] 위에서 언급한 바와 같이, SHC 는 마이크로폰 어레이를 이용한 마이크로폰 리코딩으로부터 유도될 수도 있다. SHC 가 마이크로폰 어레이들로부터 유도될 수 있는 방법의 여러 예들은 2005년 11월, J. Audio Eng. Soc., Vol. 53, No. 11, pp. 1004-1025, Poletti, M., "Three-Dimensional Surround Sound Systems Based on Spherical Harmonics" 에 설명되어 있다.

[0022] SHC들이 어떻게 오브젝트-기반의 설명으로부터 유도될 수 있는지를 예시하기 위해, 다음 방정식을 고려한다. 개개의 오디오 오브젝트에 대응하는 음장에 대한 계수들 $A_n^m(k)$ 은 다음과 같이 표현될 수도 있다:

[0023] $$A_n^m(k) = g(\omega)(-4\pi ik)h_n^{(2)}(kr_s)Y_n^{m^*}(\theta_s,\varphi_s),$$

[0024] 여기서, i 는 $\sqrt{-1}$ 이고, $h_n^{(2)}(\cdot)$ 는 차수 n 의 (제 2 종의) 구면 Hankel 함수이고, $\{r_s,\theta_s,\varphi_s\}$ 는 오브젝트의 로케이션이다. (예컨대, PCM 스트림에 관해 고속 푸리에 변환을 수행하는 것과 같은, 시간-주파수 분석 기법들을 이용하여) 오브젝트 소스 에너지 g(ω) 를 주파수의 함수로서 아는 것은 우리가 각각의 PCM 오브젝트 및 그의 로케이션을 SHC $A_n^m(k)$ 로 변환가능하게 한다. 또, (상기가 선형 및 직교 분해이므로) 각각의 오브젝트에 대한 $A_n^m(k)$ 계수들이 누적되는 것으로 표시될 수 있다. 이러한 방법으로, 다수의 PCM 오브젝트들은 $A_n^m(k)$ 계수들에 의해 (예컨대, 개개의 오브젝트들에 대한 계수 벡터들의 합계로서) 표현될 수 있다. 본질적으로, 계수들은 음장에 관한 정보 (3D 좌표들의 함수로서의 압력) 을 포함하며, 상기는 관측 지점 $\{r_s,\theta_s,\varphi_s\}$ 근처에서, 개개의 오브젝트들로부터 전체 음장의 표현으로의 변환을 나타낸다. 나머지 도면들은 오브젝트-기반 및 SHC-기반 오디오 코딩의 상황에서 아래에서 설명된다.

[0025] 도 2 는 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 수도 있는 시스템 (10) 을 예시하는 다이어그램이다. 도 2 의 예에 나타낸 바와 같이, 시스템 (10) 은 콘텐츠 생성자 디바이스 (12) 및 콘텐츠 소비자 디바이스 (14) 를 포함한다. 콘텐츠 생성자 디바이스 (12) 및 콘텐츠 소비자 디바이스 (14) 의 상황에서 설명되지만, 이 기법들은 (HOA 계수들로서 또한 지칭될 수도 있는) SHC들 또는 음장의 임의의 다른 계층적 표현이 오디오 데이터를 나타내는 비트스트림을 형성하기 위해 인코딩되는 임의의 상황에서 구현될 수도 있다. 더욱이, 콘텐츠 생성자 디바이스 (12) 는 몇개의 예들을 제공하자면, 핸드셋 (또는, 셀룰러폰), 태블릿 컴퓨터, 스마트 폰, 또는 데스크탑 컴퓨터를 포함한, 본 개시물에서 설명하는 기법들을 구현하는 것이 가능한 임의 유형의 컴퓨팅 디바이스를 나타낼 수도 있다. 이와 유사하게, 콘텐츠 소비자 디바이스 (14) 는 몇개의 예들을 제공하자면 핸드셋 (또는, 셀룰러폰), 태블릿 컴퓨터, 스마트 폰, 셋-탑 박스, 또는 데스크탑 컴퓨터를 포함한, 본 개시물에서 설명하는 기법들을 구현하는 것이 가능한 임의 유형의 컴퓨팅 디바이스를 나타낼 수도 있다.

[0026] 콘텐츠 생성자 디바이스 (12) 는 콘텐츠 소비자 디바이스 (14) 와 같은 콘텐츠 소비자들의 조작자에 의한 소비를 위해 멀티-채널 오디오 콘텐츠를 발생할 수도 있는 영화 스튜디오 또는 다른 엔터티에 의해 동작될 수도 있다. 일부 예들에서, 콘텐츠 생성자 디바이스 (12) 는 HOA 계수들 (11) 을 압축하기를 원하는 개개의 사용자에 의해 동작될 수도 있다. 종종, 콘텐츠 생성자는 비디오 콘텐츠와 함께 오디오 콘텐츠를 발생시킨다. 콘텐츠 소비자 디바이스 (14) 는 개개인에 의해 동작될 수도 있다. 콘텐츠 소비자 디바이스 (14) 는 멀티-채널 오디오 콘텐츠로서 플레이백을 위한 SHC 를 렌더링하는 것이 가능한 임의 유형의 오디오 플레이백 시스템을 지칭할 수도 있는 오디오 플레이백 시스템 (16) 을 포함할 수도 있다.

[0027] 콘텐츠 생성자 디바이스 (12) 는 오디오 편집 시스템 (18) 을 포함한다. 콘텐츠 생성자 디바이스 (12) 는 라이브 리코딩들 (7) 을 (HOA 계수들로서 직접 포함하는) 여러 포맷들로, 그리고 콘텐츠 생성자 디바이스 (12) 가 오디오 편집 시스템 (18) 을 이용하여 편집할 수도 있는 오디오 오브젝트들 (9) 을 획득한다. 콘텐츠 생성자는 편집 프로세스 동안, 추가로 편집할 필요가 있는 음장의 여러 양태들을 식별하려는 시도로 렌더링된 스피커 피드들을 청취하는 오디오 오브젝트들 (9) 로부터 HOA 계수들 (11) 을 렌더링할 수도 있다. 콘텐츠 생성자 디바이스 (12) 는 그후 (잠재적으로는, 소스 HOA 계수들이 위에서 설명된 방법으로 유도될 수도 있는 오디오 오브젝트들 (9) 중 상이한 하나의 조작을 통해서 간접적으로) HOA 계수들 (11) 을 편집할 수도 있다. 콘텐츠 생성자 디바이스 (12) 는 HOA 계수들 (11) 을 발생시키기 위해 오디오 편집 시스템 (18) 을 채용할 수도 있다. 오디오 편집 시스템 (18) 은 오디오 데이터를 편집하여 오디오 데이터를 하나 이상의 소스 구면 고조파 계수들로서 출력하는 것이 가능한 임의의 시스템을 나타낸다.

[0028] 편집 프로세스가 완료될 때, 콘텐츠 생성자 디바이스 (12) 는 HOA 계수들 (11) 에 기초하여 비트스트림 (21) 을 발생시킬 수도 있다. 즉, 콘텐츠 생성자 디바이스 (12) 는 비트스트림 (21) 을 발생시키는 본 개시물에서 설명하는 기법들의 여러 양태들에 따라서 HOA 계수들 (11) 을 인코딩하거나 또는 아니면 압축하도록 구성된 디바이스를 나타내는 오디오 인코딩 디바이스 (20) 를 포함한다. 오디오 인코딩 디바이스 (20) 는 일 예로서, 유선 또는 무선 채널, 데이터 저장 디바이스, 또는 기타 등등일 수도 있는 송신 채널을 통한 송신을 위해 비트스트림 (21) 을 발생시킬 수도 있다. 비트스트림 (21) 은 HOA 계수들 (11) 의 인코딩된 버전을 나타낼 수도 있으며, 1차 비트스트림 및 부 채널 정보로서 지칭될 수도 있는 다른 부 비트스트림 (side bitstream) 을 포함할 수도 있다.

[0029] 아래에서 좀더 자세하게 설명되지만, 오디오 인코딩 디바이스 (20) 는 벡터-기반 합성 또는 방향-기반 합성에 기초하여 HOA 계수들 (11) 을 인코딩하도록 구성될 수도 있다. 벡터-기반 분해 방법론 또는 방향-기반 분해 방법론을 수행할지 여부를 결정하기 위해, 오디오 인코딩 디바이스 (20) 는 HOA 계수들 (11) 에 적어도 부분적으로 기초하여, HOA 계수들 (11) 이 음장의 자연스러운 리코딩 (예컨대, 라이브 리코딩 7) 을 통해서 발생되었는지 또는 일 예로서, PCM 오브젝트와 같은, 오디오 오브젝트들 (9) 로부터 인공적으로 (즉, 합성적으로) 발생되었는지 여부를 결정할 수도 있다. HOA 계수들 (11) 이 오디오 오브젝트들 (9) 로부터 발생되었을 때, 오디오 인코딩 디바이스 (20) 는 방향-기반 분해 방법론을 이용하여 HOA 계수들 (11) 을 인코딩할 수도 있다. HOA 계수들 (11) 이 예를 들어, eigenmike 를 이용하여 라이브로 캡쳐되었을 때, 오디오 인코딩 디바이스 (20) 는 벡터-기반 분해 방법론에 기초하여 HOA 계수들 (11) 을 인코딩할 수도 있다. 상기 차이 (distinction) 는 벡터-기반의 또는 방향-기반 분해 방법론이 채용될 수도 있는 경우의 일 예를 나타낸다. 자연스러운 리코딩들, 인공적으로 발생된 콘텐츠 또는 2개의 혼합물 (하이브리드 콘텐츠) 에 대해 어느 하나 또는 양쪽이 유용할 수도 있는 다른 경우들이 있을 수도 있다. 더욱이, HOA 계수들의 단일 시간-프레임을 코딩하는데 양쪽의 방법론들을 동시에 사용하는 것이 또한 가능하다.

[0030] 예시의 목적을 위해, HOA 계수들 (11) 이 라이브로 캡쳐되었다고 또는 아니면 라이브 리코딩 7 과 같은 라이브 리코딩들을 나타낸다고 오디오 인코딩 디바이스 (20) 가 결정한다고 가정하면, 오디오 인코딩 디바이스 (20) 는 선형 가역 변환 (linear invertible transform; LIT) 의 적용을 수반하는 벡터-기반 분해 방법론을 이용하여 HOA 계수들 (11) 을 인코딩하도록 구성될 수도 있다. 선형 가역 변환의 일 예는 "특이 값 분해" (또는, "SVD") 로서 지칭된다. 이 예에서, 오디오 인코딩 디바이스 (20) 는 HOA 계수들 (11) 의 분해된 버전을 결정하기 위해 SVD 를 HOA 계수들 (11) 에 적용할 수도 있다. 오디오 인코딩 디바이스 (20) 는 그후 HOA 계수들 (11) 의 분해된 버전의 재배열을 촉진할 수도 있는 여러 파라미터들을 식별하기 위해, HOA 계수들 (11) 의 분해된 버전을 분석할 수도 있다. 오디오 인코딩 디바이스 (20) 는 그후 식별된 파라미터들을 이용하여 HOA 계수들 (11) 의 분해된 버전을 재배열할 수도 있으며, 여기서, 이러한 재배열은, 이하에서 더 자세히 설명하는 바와 같이, 그 변환이 HOA 계수들의 프레임들 (여기서, 프레임은 HOA 계수들 (11) 의 M 개의 샘플들을 포함할 수도 있으며 M 은 일부 예들에서, 1024 로 설정된다) 을 가로질러 HOA 계수들을 재배열할 수도 있다는 점을 고려하면, 코딩 효율을 향상시킬 수도 있다. HOA 계수들 (11) 의 분해된 버전을 재배열한 후, 오디오 인코딩 디바이스 (20) 는 음장의 포그라운드 (또는, 즉, 독특한, 지배적인 또는 현저한) 구성요소들을 나타내는 HOA 계수들 (11) 의 분해된 버전을 선택할 수도 있다. 오디오 인코딩 디바이스 (20) 는 포그라운드 구성요소들을 나타내는 HOA 계수들 (11) 의 분해된 버전을 오디오 오브젝트 및 연관된 방향 정보로서 규정할 수도 있다.

[0031] 오디오 인코딩 디바이스 (20) 는 음장의 하나 이상의 백그라운드 (또는, 즉, 주변) 구성요소들을 나타내는 HOA 계수들 (11) 을 적어도 부분적으로 식별하기 위해, HOA 계수들 (11) 에 대해 음장 분석을 수행할 수도 있다. 오디오 인코딩 디바이스 (20) 는 일부 예들에서, 백그라운드 구성요소들이 단지 (예컨대, 제 2 또는 더 높은-

차수 구형 기저 함수들에 대응하는 HOA 계수들 (11) 이 아닌, 제로 및 1차 구형 기저 함수들에 대응하는 HOA 계수들 (11) 과 같은) HOA 계수들 (11) 의 임의의 주어진 샘플의 서브세트만을 포함할 수도 있다는 점을 고려하면, 백그라운드 구성요소들에 대해서, 에너지 보상을 수행할 수도 있다. 차수-감소가 수행될 때, 즉, 오디오 인코딩 디바이스 (20) 가 차수 감소를 수행하는 것에 기인하는 전체 에너지에서의 변화를 보상하기 위해 HOA 계수들 (11) 의 나머지 백그라운드 HOA 계수들을 증대시킬 수도 (예컨대, 그로부터/그에 에너지를 감산/가산할 수도) 있다.

[0032] 오디오 인코딩 디바이스 (20) 는 다음에, 백그라운드 구성요소들을 나타내는 HOA 계수들 (11) 의 각각 및 포그라운드 오디오 오브젝트들의 각각에 대해서 (MPEG 서라운드, MPEG-AAC, MPEG-USAC 또는 음향심리 인코딩의 다른 알려진 형태들과 같은) 음향심리 인코딩의 유형을 수행할 수도 있다. 오디오 인코딩 디바이스 (20) 는 포그라운드 방향 정보에 대해 내삽의 유형을 수행하고 그후 그 내삽된 포그라운드 방향 정보에 대해 차수 감소를 수행하여, 차수 감소된 포그라운드 방향 정보를 발생시킬 수도 있다. 오디오 인코딩 디바이스 (20) 는 일부 예들에서, 차수 감소된 포그라운드 방향 정보에 대해 양자화를 추가로 수행하여, 코딩된 포그라운드 방향 정보를 출력할 수도 있다. 일부의 경우, 양자화는 스칼라/엔트로피 양자화를 포함할 수도 있다. 오디오 인코딩 디바이스 (20) 는 그후 인코딩된 백그라운드 구성요소들, 인코딩된 포그라운드 오디오 오브젝트들, 및 양자화된 방향 정보를 포함시키기 위해 비트스트림 (21) 을 형성할 수도 있다. 오디오 인코딩 디바이스 (20) 는 그후 비트스트림 (21) 을 콘텐츠 소비자 디바이스 (14) 로 송신하거나 또는 아니면 출력할 수도 있다.

[0033] 도 2 에서 콘텐츠 소비자 디바이스 (14) 로 직접 송신되는 것으로 나타내지만, 콘텐츠 생성자 디바이스 (12) 는 비트스트림 (21) 을 콘텐츠 생성자 디바이스 (12) 와 콘텐츠 소비자 디바이스 (14) 사이에 위치된 중간 디바이스로 출력할 수도 있다. 중간 디바이스는 이 비트스트림을 요청할 수도 있는 콘텐츠 소비자 디바이스 (14) 에게의 추후 전달을 위해 비트스트림 (21) 을 저장할 수도 있다. 중간 디바이스는 파일 서버, 웹 서버, 데스크탑 컴퓨터, 랩탑 컴퓨터, 태블릿 컴퓨터, 모바일 폰, 스마트 폰, 또는 오디오 디코더에 의한 추후 취출을 위해 비트스트림 (21) 을 저장하는 것이 가능한 임의의 다른 디바이스를 포함할 수도 있다. 중간 디바이스는 비트스트림 (21) 을 (그리고, 어쩌면, 대응하는 비디오 데이터 비트스트림을 송신하는 것과 함께) 비트스트림 (21) 을 요청하는 콘텐츠 소비자 디바이스 (14) 와 같은, 가입자들에게 스트리밍하는 것이 가능한 콘텐츠 전달 네트워크에 상주할 수도 있다.

[0034] 이의 대안으로, 콘텐츠 생성자 디바이스 (12) 는 비트스트림 (21) 을, 대부분이 컴퓨터에 의해 판독가능하고 따라서 컴퓨터-판독가능 저장 매체들 또는 비일시성 컴퓨터-판독가능 저장 매체들로서 지칭될 수도 있는, 컴팩트 디스크, 디지털 비디오 디스크, 고화질 비디오 디스크 또는 다른 저장 매체들과 같은, 저장 매체에 저장할 수도 있다. 이 상황에서, 송신 채널은 매체들에 저장된 콘텐츠가 송신되는 채널들을 지칭할 수도 있다 (그리고, 소매점들 및 다른 저장-기반의 전달 메커니즘을 포함할 수도 있다). 어쨌든, 본 개시물의 기법들은 따라서 이 점에서 도 2 의 예에 한정되지 않아야 한다.

[0035] 도 2 의 예에서 추가로 나타낸 바와 같이, 콘텐츠 소비자 디바이스 (14) 는 오디오 플레이백 시스템 (16) 을 포함한다. 오디오 플레이백 시스템 (16) 은 멀티-채널 오디오 데이터를 플레이백하는 것이 가능한 임의의 오디오 플레이백 시스템을 나타낼 수도 있다. 오디오 플레이백 시스템 (16) 은 다수의 상이한 렌더러들 (22) 을 포함할 수도 있다. 렌더러들 (22) 은 상이한 유형의 렌더링을 각각 제공할 수도 있으며, 여기서, 상이한 유형들의 렌더링은 벡터-기반 진폭 패닝 (VBAP) 을 수행하는 여러 방법들 중 하나 이상, 및/또는 음장 합성을 수행하는 여러 방법들 중 하나 이상을 포함할 수도 있다. 본원에서 사용될 때, "A 및/또는 B" 는 "A 또는 B", 또는 "A 및 B" 양쪽을 의미한다.

[0036] 오디오 플레이백 시스템 (16) 은 오디오 디코딩 디바이스 (24) 를 더 포함할 수 있다. 오디오 디코딩 디바이스 (24) 는 비트스트림 (21) 으로부터 HOA 계수들 (11') 을 디코딩하도록 구성된 디바이스를 나타낼 수도 있으며, 여기서, HOA 계수들 (11') 은 HOA 계수들 (11) 과 유사하지만 손실있는 동작들 (예컨대, 양자화) 및/또는 송신 채널을 통한 송신으로 인해 상이할 수도 있다. 즉, 오디오 디코딩 디바이스 (24) 는 비트스트림 (21) 에 규정된 포그라운드 방향 정보를 역양자화할 수도 있지만, 또한 비트스트림 (21) 에 규정된 포그라운드 오디오 오브젝트들 및 백그라운드 구성요소들을 나타내는 인코딩된 HOA 계수들에 대해 음향심리 디코딩을 수행할 수도 있다. 오디오 디코딩 디바이스 (24) 는 추가로, 디코딩된 포그라운드 방향 정보에 대해 내삽을 수행하고 그후 그 디코딩된 포그라운드 오디오 오브젝트들 및 내삽된 포그라운드 방향 정보에 기초하여 포그라운드 구성요소들을 나타내는 HOA 계수들을 결정할 수도 있다. 오디오 디코딩 디바이스 (24) 는 그후 포그라운드 구성요소들을 나타내는 결정된 HOA 계수들 및 백그라운드 구성요소들을 나타내는 디코딩된 HOA 계수들에 기초하여

HOA 계수들 (11') 을 결정할 수도 있다.

[0037] 오디오 플레이백 시스템 (16) 은 HOA 계수들 (11') 을 얻기 위해 비트스트림 (21) 을 디코딩한 후, HOA 계수들 (11') 을 렌더링하여 라우드스피커 피드들 (25) 을 출력할 수도 있다. 라우드스피커 피드들 (25) 은 (용이한 예시의 목적을 위해 도 2 의 예에 도시되지 않은) 하나 이상의 라우드스피커들을 구동할 수도 있다.

[0038] 적합한 렌더러를 선택하기 위해, 또는, 일부 경우, 적합한 렌더러를 발생시키기 위해, 오디오 플레이백 시스템 (16) 은 다수의 라우드스피커들 및/또는 라우드스피커들의 공간 기하학적 구조를 나타내는 라우드스피커 정보 (13) 를 획득할 수도 있다. 일부의 경우, 오디오 플레이백 시스템 (16) 은 참조 마이크로폰을 이용하여 라우드스피커 정보 (13) 를 획득하고 라우드스피커 정보 (13) 를 동적으로 결정하는 방법으로 라우드스피커들을 구동할 수도 있다. 다른 경우들에서, 또는 라우드스피커 정보 (13) 의 동적 결정과 함께, 오디오 플레이백 시스템 (16) 은 오디오 플레이백 시스템 (16) 과 인터페이스하여 라우드스피커 정보 (13) 를 입력하도록 사용자에게 프롬프트할 수도 있다.

[0039] 오디오 플레이백 시스템 (16) 은 그후 라우드스피커 정보 (13) 에 기초하여 오디오 렌더러들 (22) 중 하나를 선택할 수도 있다. 일부의 경우, 오디오 플레이백 시스템 (16) 은 어떤 오디오 렌더러들 (22) 도 라우드스피커 정보 (13) 에 규정된 것에 대한 어떤 임계치 유사성 척도 (라우드스피커 기하학적 구조의 관점에서) 내에 있지 않을 때, 라우드스피커 정보 (13) 에 기초하여 오디오 렌더러들 (22) 중 하나를 발생시킬 수도 있다. 오디오 플레이백 시스템 (16) 은 일부 경우, 기존 오디오 렌더러들 (22) 중 하나를 선택하려고 먼저 시도함이 없이, 라우드스피커 정보 (13) 에 기초하여 오디오 렌더러들 (22) 중 하나를 발생시킬 수도 있다.

[0040] 도 3 은 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 수도 있는 도 2 의 예에 나타낸 오디오 인코딩 디바이스 (20) 의 일 예를 좀더 자세하게 예시하는 블록도이다. 오디오 인코딩 디바이스 (20) 는 콘텐츠 분석 유닛 (26), 벡터-기반 분해 유닛 (27) 및 방향-기반 분해 유닛 (28) 을 포함한다. 아래에서 간단히 설명되지만, 오디오 인코딩 디바이스 (20) 및 HOA 계수들을 압축하거나 또는 아니면 인코딩하는 여러 양태들에 관한 더 많은 정보는 "INTERPOLATION FOR DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로, 2014 년 5월 29일에 출원된, 국제 특허 출원 공개 번호 WO 2014/194099호에서 입수가능하다.

[0041] 콘텐츠 분석 유닛 (26) 은 HOA 계수들 (11) 의 콘텐츠를 분석하여 HOA 계수들 (11) 이 라이브 리코딩 또는 오디오 오브젝트로부터 발생된 콘텐츠를 나타내는지 여부를 식별하도록 구성된 유닛을 나타낸다. 콘텐츠 분석 유닛 (26) 은 HOA 계수들 (11) 이 실제 음장의 리코딩으로부터 또는 인공적인 오디오 오브젝트로부터 발생되었는지 여부를 결정할 수도 있다. 일부의 경우, 프레임으로 된 HOA 계수들 (11) 이 리코딩으로부터 발생되었을 때, 콘텐츠 분석 유닛 (26) 은 HOA 계수들 (11) 을 벡터-기반 분해 유닛 (27) 으로 전달한다. 일부의 경우, 프레임으로 된 HOA 계수들 (11) 이 합성 오디오 오브젝트로부터 발생되었을 때, 콘텐츠 분석 유닛 (26) 은 HOA 계수들 (11) 을 방향-기반 합성 유닛 (28) 으로 전달한다. 방향-기반 합성 유닛 (28) 은 HOA 계수들 (11) 의 방향-기반 합성을 수행하여 방향-기반 비트스트림 (21) 을 발생시키도록 구성된 유닛을 나타낼 수도 있다.

[0042] 도 3 의 예에 나타낸 바와 같이, 벡터-기반 분해 유닛 (27) 은 선형 가역 변환 (LIT) 유닛 (30), 파라미터 계산 유닛 (32), 재정리 유닛 (34), 포그라운드 선택 유닛 (36), 에너지 보상 유닛 (38), 음향심리 오디오 코더 유닛 (40), 비트스트림 발생 유닛 (42), 음장 분석 유닛 (44), 계수 감소 유닛 (46), 백그라운드 (BG) 선택 유닛 (48), 시공간적 내삽 유닛 (50), 및 양자화 유닛 (52) 을 포함할 수도 있다.

[0043] 선형 가역 변환 (LIT) 유닛 (30) 은 HOA 계수들 (11) 을 HOA 채널들의 유형으로 수신하며, 각각의 채널은 (HOA[k] 로서 표시될 수도 있으며, 여기서 k 는 샘플들의 현재의 프레임 또는 블록을 표시할 수도 있는) 구형 기저 함수들의 주어진 차수, 서브-차수와 연관된 계수의 블록 또는 프레임을 나타낸다. HOA 계수들 (11) 의 매트릭스는 치수들 D: M x $(N+1)^2$ 을 가질 수도 있다.

[0044] 즉, LIT 유닛 (30) 은 특이 값 분해로서 지칭되는 분석의 유형을 수행하도록 구성된 유닛을 나타낼 수도 있다. SVD 에 대해 설명되지만, 본 개시물에서 설명하는 기법들은 선형으로 비상관된, 에너지 압축된 출력의 세트들을 제공하는 임의의 유사한 변환 또는 분해에 대해서 수행될 수도 있다. 또한, 본 개시물에서 "세트들" 에 대한 참조는 구체적으로 반대로 언급되지 않는 한 비-제로 세트들을 지칭하는 것으로 일반적으로 의도되며, 소위 "빈 (empty) 세트" 를 포함하는 세트들의 고전 (classical) 수학적 정의를 지칭하는 것으로 의도되지 않는다.

[0045] 대안적인 변환은 "PCA" 로서 종종 지칭되는 주요 구성요소 분석을 포함할 수도 있다. PCA 는 어쩌면 상관된

변수들의 관측들의 세트를 주요 구성요소들로서 지칭되는 선형으로 비상관된 변수들의 세트로 변환시키기 위해 직교 변환을 채용하는 수학적 프로시저를 지칭한다. 선형으로 비상관된 변수들은 서로에 대해 선형 통계적 관계 (또는, 의존) 를 가지지 않는 변수들을 나타낸다. 주요 구성요소들은 서로에 대해 작은 정도의 통계적 상관을 갖는 것으로 설명될 수도 있다. 어쨌든, 소위 주요 구성요소들의 개수는 원래 변수들의 개수보다 적거나 또는 동일하다. 일부 예들에서, 변환은, 제 1 주요 구성요소가 최대 가능한 (또는, 즉, 가능한 한 많은 데이터에서의 변동성을 차지하는) 분산을 가지며 그 다음으로 각각의 다음 구성요소가, 다음에 이어지는 구성요소가 선행하는 구성요소들에 직교하다 (이것은 그와 비상관된 것으로 달리 말해질 수도 있다) 는 제약 하에서 가능한 최고 분산을 가지는 방식으로 정의된다. PCA 는 HOA 계수들 (11) 의 관점에서 HOA 계수들 (11) 의 압축을 초래할 수도 있는 차수-감소의 유형을 수행할 수도 있다. 상황에 따라서, PCA 는 몇 개의 예들을 들면, 이산 Karhunen-Loeve 변환, Hotelling 변환, 적합 직교 분해 (POD), 및 고유치 분해 (EVD) 와 같은, 다수의 상이한 이름들로 지칭될 수도 있다. 오디오 데이터를 압축하는 기본적인 목표에 도움이 되는 이러한 동작들의 성질들은 멀티채널 오디오 데이터의 ' 에너지 압축' 및 '비상관' 이다.

[0046] 어쨌든, 예의 목적을 위해 LIT 유닛 (30) 이 ("SVD" 로서 또한 지칭될 수도 있는) 특이 값 분해를 수행한다고 가정하면, LIT 유닛 (30) 은 HOA 계수들 (11) 을 변환된 HOA 계수들의 2개 이상의 세트들로 변환할 수도 있다.

변환된 HOA 계수들의 "세트들" 은 변환된 HOA 계수들의 벡터들을 포함할 수도 있다. 도 3 의 예에서, LIT 유닛 (30) 은 HOA 계수들 (11) 에 대해 SVD 를 수행하여, 소위 V 매트릭스, S 매트릭스, 및 U 매트릭스를 발생시킬 수도 있다. SVD 는, 선형 대수학에서, y 곱하기 z (y-by-z) 실수 또는 복소수 매트릭스 X (여기서, X 는 HOA 계수들 (11) 과 같은, 멀티-채널 오디오 데이터를 나타낼 수도 있다) 의 인수분해를 다음 형태로 나타낼 수도 있다:

[0047] X = USV*

[0048] U 는 y 곱하기 y 실수 또는 복소수 유니터리 매트릭스 (unitary matrix) 을 나타낼 수도 있으며, 여기서, U 의 y 칼럼들은 멀티-채널 오디오 데이터의 좌측-특이 벡터들로서 알려져 있다. S 는 대각선 상에 비-음의 실수들을 가지는 y 곱하기 z (y-by-z) 직사각형의 대각선 매트릭스를 나타낼 수도 있으며, 여기서, S 의 대각선 값들은 멀티-채널 오디오 데이터의 특이 값들로서 알려져 있다. (V 의 켤레 전치를 표시할 수도 있는) V* 는 z 곱하기 z 실수 또는 복소수 유니터리 매트릭스를 나타낼 수도 있으며, 여기서, V* 의 z 칼럼들은 멀티-채널 오디오 데이터의 우측-특이 벡터들로서 알려져 있다.

[0049] HOA 계수들 (11) 을 포함하는 멀티-채널 오디오 데이터에 적용되는 것으로 본 개시물에서 설명되지만, 이 기법들은 임의 유형의 멀티-채널 오디오 데이터에 적용될 수도 있다. 이러한 방법으로, 오디오 인코딩 디바이스 (20) 는 적어도 음장의 일부분을 나타내는 멀티-채널 오디오 데이터에 대해서 특이 값 분해를 수행하여, 멀티-채널 오디오 데이터의 좌측-특이 벡터들을 나타내는 U 매트릭스, 멀티-채널 오디오 데이터의 특이 값들을 나타내는 S 매트릭스 및 멀티-채널 오디오 데이터의 우측-특이 벡터들을 나타내는 V 매트릭스를 발생시킬 수도 있으며, 멀티-채널 오디오 데이터를 U 매트릭스, S 매트릭스 및 V 매트릭스의 하나 이상 중 적어도 일부분의 함수로서 나타낼 수도 있다.

[0050] 일부 예들에서, 아래에서 참조되는 SVD 수학적 수식에서 V* 매트릭스는 SVD 가 복소수들을 포함하는 매트릭스들에 적용될 수도 있다는 점을 반영하기 위해 V 매트릭스의 켤레 전치로서 표시된다. 단지 실수들만을 포함하는 매트릭스들에 적용될 때, V 매트릭스의 켤레 복소수 (또는, 즉, V* 매트릭스) 는 V 매트릭스의 전치인 것으로 간주될 수도 있다. 아래에서는, 용이한 예시 목적을 위해, V* 매트릭스보다는, V 매트릭스가 SVD 를 통해서 출력되는 결과로 HOA 계수들 (11) 이 실수들을 포함한다고 가정된다. 더욱이, 본 개시물에서 V 매트릭스로서 표시되지만, V 매트릭스에 대한 참조는 적당한 경우 V 매트릭스의 전치를 지칭하는 것으로 이해되어야 한다. V 매트릭스인 것으로 가정되지만, 이 기법들은 복소 계수들을 가지는 HOA 계수들 (11) 과 유사한 방식으로 적용될 수도 있으며, 여기서, SVD 의 출력은 V* 매트릭스이다. 따라서, 본 기법들은 이 점에서, 단지 V 매트릭스를 발생시키기 위한 SVD 의 적용을 허용하는데만 한정되지 않아야 하며, V* 매트릭스를 발생시키기 위한 복소수 구성요소들을 가지는 HOA 계수들 (11) 에의 SVD 의 적용을 포함할 수도 있다.

[0051] 어쨌든, LIT 유닛 (30) 은 고-차수 앰비소닉스 (HOA) 오디오 데이터 (여기서, 앰비소닉스 오디오 데이터는 HOA 계수들 (11) 의 블록들 또는 샘플들 또는 임의의 다른 유형의 멀티-채널 오디오 데이터를 포함한다) 의 각각의 블록 (프레임으로 지칭될 수도 있음) 에 대해 블록-방식 유형의 SVD 를 수행할 수도 있다. 위에서 언급한 바와 같이, 변수 M 은 샘플들에서 오디오 프레임의 길이를 표시하기 위해 사용될 수도 있다. 예를 들어, 오디오 프레임이 1024 개의 오디오 샘플들을 포함할 때, M 은 1024 와 동일하다. M 에 대한 전형적인 값에 대

해서 설명되지만, 본 개시물의 기법들은 M 에 대한 전형적인 값에 한정되지 않아야 한다. LIT 유닛 (30) 은 따라서 M 곱하기 $(N+1)^2$ HOA 계수들을 가지는 HOA 계수들 (11) 의 블록에 대해 블록-방식 SVD 를 수행할 수도 있으며, 여기서, N 은, 또한, HOA 오디오 데이터의 차수를 표시한다. LIT 유닛 (30) 은 SVD 를 통해서, V 매트릭스, S 매트릭스, 및 U 매트릭스를 발생시킬 수도 있으며, 여기서, 매트릭스들의 각각은 위에서 설명된 개개의 V, S 및 U 매트릭스들을 나타낼 수도 있다. 이러한 방법으로, 선형 가역 변환 유닛 (30) 은 HOA 계수들 (11) 에 대해 SVD 를 수행하여, 치수들 D: M x $(N+1)^2$ 를 가지는 (S 벡터들과 U 벡터들의 결합된 버전을 나타낼 수도 있는) US[k] 벡터들 (33) 및 치수들 D: $(N+1)^2$ x $(N+1)^2$ 를 가지는 V[k] 벡터들 (35) 을 출력할 수도 있다. US[k] 매트릭스에서의 개개의 벡터 엘리먼트들은 또한 $X_{PS}(k)$ 로서 지칭될 수도 있으며, 반면 V[k] 매트릭스의 개개의 벡터들은 또한 $v(k)$ 로서 지칭될 수도 있다.

[0052] U, S 및 V 매트릭스들의 분석은 매트릭스들이 X 로 위에서 나타낸 기본적인 음장의 공간 및 시간 특성들을 운반하거나 또는 나타낸다는 것을 보일 수도 있다. (길이 M 샘플들의) U 에서의 N 개의 벡터들의 각각은, 서로에 직교하며 (방향 정보로서 또한 지칭될 수도 있는) 임의의 공간 특성들로부터 분리되어 있는 정규화된 분리된 오디오 신호들을 (M 샘플들로 표현된 시간 기간에 대한) 시간의 함수로서 나타낼 수도 있다. 공간 형태 및 위치 (r, 쎄타(theta), 파이(phi)) 폭을 나타내는, 공간 특성들은 V 매트릭스 (길이 $(N+1)^2$ 각각) 에서, 개개의 i 번째 벡터들, $v^{(i)}(k)$ 로 대신 표시될 수도 있다. $v^{(i)}(k)$ 벡터들의 각각의 개개의 엘리먼트들은 연관된 오디오 오브젝트에 대한 음장의 형태 및 방향을 기술하는 HOA 계수를 나타낼 수도 있다. U 매트릭스 및 V 매트릭스의 벡터들 양쪽은 그들의 자승 평균 평방근 에너지들이 1 과 동일하도록 정규화된다. U 에서의 오디오 신호들의 에너지는 따라서 S 에서 대각선 엘리먼트들에 의해 표현된다. U 와 S 를 곱하여 (개개의 벡터 엘리먼트들 $X_{PS}(k)$ 을 가지는) US[k] 를 형성하는 것은, 따라서 실제 (true) 에너지들을 가지는 오디오 신호를 나타낸다. (U 에서) 오디오 시간-신호들, (S 에서) 그들의 에너지들 및 (V 에서) 그들의 공간 특성들을 분리시키는 SVD 분해의 능력은 본 개시물에서 설명하는 기법들의 여러 양태들을 지원할 수도 있다. 또, US[k] 와 V[k] 의 벡터 곱셈에 의해 기본적인 HOA[k] 계수들, X 를 합성하는 모델은, 이 문서 전반에 걸쳐서 사용되는 용어 "벡터-기반 분해" 를 야기시킨다.

[0053] HOA 계수들 (11) 에 대해 직접 수행되는 것으로 설명되지만, LIT 유닛 (30) 은 HOA 계수들 (11) 의 유도체들에 선형 가역 변환을 적용할 수도 있다. 예를 들어, LIT 유닛 (30) 은 HOA 계수들 (11) 로부터 유도된 전력 스펙트럼 밀도 매트릭스에 대해 SVD 를 적용할 수도 있다. 전력 스펙트럼 밀도 매트릭스는 PSD 로서 표시될 수도 있으며, 아래에 뒤따르는 의사-코드에서 약술한 바와 같은, hoaFrame 으로의 hoaFrame 의 전치의 매트릭스 곱셈을 통해서 획득될 수도 있다. hoaFrame 표기는 HOA 계수들 (11) 의 프레임을 지칭한다.

[0054] LIT 유닛 (30) 은 SVD (svd) 를 PSD 에 적용한 후, $S[k]^2$ 매트릭스 (S_squared) 및 V[k] 매트릭스를 획득할 수도 있다. $S[k]^2$ 매트릭스는 사각형으로 된 S[k] 매트릭스를 표시할 수도 있으며, 그 때문에, LIT 유닛 (30) 은 제곱근 동작을 $S[k]^2$ 매트릭스에 적용하여, S[k] 매트릭스를 얻을 수도 있다. LIT 유닛 (30) 은 일부 경우, V[k] 매트릭스에 대해 양자화를 수행하여, (V[k]' 매트릭스로서 표시될 수도 있는) 양자화된 V[k] 매트릭스를 획득할 수도 있다. LIT 유닛 (30) 은 S[k] 매트릭스에 양자화된 V[k]' 매트릭스를 먼저 곱함으로써 U[k] 매트릭스를 획득하여, SV[k]' 매트릭스를 획득할 수도 있다. LIT 유닛 (30) 은 다음으로 SV[k]' 매트릭스의 유사-역원 (pinv) 을 획득하고 그후 HOA 계수들 (11) 에 SV[k]' 매트릭스의 유사-역원을 곱하여, U[k] 매트릭스를 획득할 수도 있다. 전술한 것은 다음 의사-코드로 표현될 수도 있다:

[0055]
```
PSD = hoaFrame'*hoaFrame;
```

[0056]
```
[V, S_squared] = svd(PSD',econ');
```

[0057]
```
S = sqrt(S_squared);
```

[0058]
```
U = hoaFrame * pinv(S*V');
```

[0059] 계수들 자신보다는, HOA 계수들의 전력 스펙트럼 밀도 (PSD) 에 대해 SVD 를 수행함으로써, LIT 유닛 (30) 은 프로세서 사이클들 및 저장 공간 중 하나 이상의 관점에서 SVD 를 수행하는 계산 복잡성을 잠재적으로 감소시키

는 한편, SVD 가 HOA 계수들에 직접 적용된 것처럼 동일한 소스 오디오 인코딩 효율을 달성할 수도 있다. 즉, 상기 설명된 PSD-유형 SVD 는 SVD 가 F*F 매트릭스 (여기서, F 는 HOA 계수들의 개수) 상에서 이루어지기 때문에, M 이 프레임 길이, 즉, 1024 또는 더 이상의 샘플들인, M * F 매트릭스에 비해, 잠재적으로 더 적은 계산 요구적일 수도 있다. SVD 의 복잡성은 지금은, HOA 계수들 (11) 보다는 PSD 에의 적용을 통해서, HOA 계수들 (11) 에 적용될 때 $O(M*L^2)$ 와 비교하여 (여기서, O(*) 는 컴퓨터-과학 분야에 공통된 계산 복잡성의 big-O 표기를 나타낸다), 대략 $O(L^3)$ 일 수도 있다.

[0060] 이 점에서, LIT 유닛 (30) 은 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터 (예컨대, 위에서 V-벡터) 를 획득하기 위해 고-차수 앰비소닉 오디오 데이터 (11) 에 대해 분해를 수행하거나 또는 아니면 고-차수 앰비소닉 오디오 데이터 (11) 를 분해할 수도 있다. 분해는 SVD, EVD 또는 임의의 다른 유형의 분해를 포함할 수도 있다.

[0061] 파라미터 계산 유닛 (32) 은 상관 파라미터 (R), 방향 성질들 파라미터들 (Θ, ϕ, r), 및 에너지 성질 (e) 과 같은, 여러 파라미터들을 계산하도록 구성된 유닛을 나타낸다. 현재의 프레임에 대한 파라미터들의 각각은 R[k], Θ[k], ϕ[k], r[k] 및 e[k] 로서 표시될 수도 있다. 파라미터 계산 유닛 (32) 은 US[k] 벡터들 (33) 에 대해 에너지 분석 및/또는 상관 (또는, 소위 교차-상관) 을 수행하여, 파라미터들을 식별할 수도 있다.

파라미터 계산 유닛 (32) 은 또한 이전 프레임에 대한 파라미터들을 결정할 수도 있으며, 여기서 이전 프레임 파라미터들은 US[k-1] 벡터 및 V[k-1] 벡터들의 이전 프레임에 기초하여 R[k-1], Θ[k-1], ϕ[k-1], r[k-1] 및 e[k-1] 로 표시될 수도 있다. 파라미터 계산 유닛 (32) 은 현재의 파라미터들 (37) 및 이전 파라미터들 (39) 을 재정리 유닛 (34) 으로 출력할 수도 있다.

[0062] SVD 분해는, US[k-1][p] 벡터로서 (또는, 대안적으로는, $X_{PS}^{(p)}(k-1)$ 로서) 표시될 수도 있는, US[k-1] 벡터들 (33) 에서의 p-번째 벡터에 의해 표시되는 오디오 신호/오브젝트가 US[k][p] 벡터들 (33) 로서 (또는, 대안적으로는, $X_{PS}^{(p)}(k)$ 로서) 또한 표시될 수도 있는, US[k] 벡터들 (33) 에서의 p-번째 벡터에 의해 표시되는, (시간에 맞춰 진행된) 동일한 오디오 신호 /오브젝트일 것을 보장하지 않는다. 파라미터 계산 유닛 (32) 에 의해 계산된 파라미터들은 그들의 자연스러운 평가 또는 시간 경과에 따른 연속성을 표시하기 위해 오디오 오브젝트들을 재정리하는데 재정리 유닛 (34) 에 의해 사용될 수도 있다.

[0063] 즉, 재정리 유닛 (34) 은 파라미터들 (37) 의 각각을 제 1 US[k] 벡터들 (33) 과 비교하여, 제 2 US[k-1] 벡터들 (33) 에 대한 파라미터들 (39) 의 각각에 대해 턴-와이즈 (turn-wise) 할 수도 있다. 재정리 유닛 (34) 은 US[k] 매트릭스 (33) 및 V[k] 매트릭스 (35) 내 여러 벡터들을 현재의 파라미터들 (37) 및 이전 파라미터들 (39) 에 기초하여 (일 예로서, Hungarian 알고리즘을 이용하여) 재정리하여, (수학적으로 $\overline{US}[k]$ 로서 표시될 수도 있는) 재정리된 US[k] 매트릭스 (33') 및 (수학적으로 $\overline{V}[k]$ 로서 표시될 수도 있는) 재정리된 V[k] 매트릭스 (35') 를 포그라운드 사운드 (또는, 지배적인 사운드 - PS) 선택 유닛 (36) ("포그라운드 선택 유닛 (36)") 및 에너지 보상 유닛 (38) 으로 출력할 수도 있다.

[0064] 음장 분석 유닛 (44) 은 목표 비트레이트 (41) 를 잠재적으로 달성하도록 HOA 계수들 (11) 에 대해 음장 분석을 수행하도록 구성된 유닛을 나타낼 수도 있다. 음장 분석 유닛 (44) 은 그 분석에, 및/또는 수신된 목표 비트레이트 (41) 에 기초하여, (주변 또는 백그라운드 채널들의 총 개수 (BG_{TOT}) 및 포그라운드 채널들 또는, 즉, 지배적인 채널들의 개수의 함수일 수도 있는) 음향심리 코더 인스턴스화들의 총 개수를 결정할 수도 있다. 음향심리 코더 인스턴스화들의 총 개수는 numHOATransportChannels 로서 표시될 수 있다.

[0065] 또한, 음장 분석 유닛 (44) 은 또한 목표 비트레이트 (41) 를 잠재적으로 달성하기 위해, 포그라운드 채널들의 총 개수 (nFG) (45), 백그라운드 (또는, 즉, 주변) 음장의 최소 차수 (N_{BG} 또는, 대안적으로, MinAmbHOAorder), 백그라운드 음장의 최소 차수를 나타내는 실수 채널들의 대응하는 개수 ($nBGa = (MinAmbHOAorder + 1)^2$), 및 (도 3 의 예에서 일괄하여 백그라운드 채널 정보 (43) 로서 표시될 수도 있는) 전송할 추가적인 BG HOA 채널들의 인덱스들 (i) 을 결정할 수도 있다. 백그라운드 채널 정보 (42) 는 또한 주변 채널 정보 (43) 로서 지칭될 수도 있다. numHOATransportChannels - nBGa 로부터 남은 채널들의 각각은, "추가적인 백그라운드/주변 채널", "활성 벡터-기반 지배적인 채널", "활성 방향 기반 지배적인 신호" 또는 "완전히 비활성적" 일 수도 있

다. 일 양태에서, 채널 유형들은 2 비트 (예컨대, 00: 방향 기반 신호; 01: 벡터-기반 지배적인 신호; 10: 추가적인 주변 신호; 11: 비활성 신호) 신택스 엘리먼트로서 ("ChannelType" 으로서) 표시될 수도 있다. 백그라운드 또는 주변 신호들의 총 개수, nBGa 는, $(MinAmbHOAorder +1)^2$ + 그 프레임에 대한 비트스트림에서 채널 유형으로 나타나는 (상기 예에서의) 인덱스 10 의 횟수로 주어질 수도 있다.

[0066] 어쨌든, 음장 분석 유닛 (44) 은 목표 비트레이트 (41) 에 기초하여, 백그라운드 (또는, 즉, 주변) 채널들의 개수 및 포그라운드 (또는, 즉, 지배적인) 채널들의 개수를 선택할 수도 있으며, 목표 비트레이트 (41) 가 상대적으로 더 높을 때 (예컨대, 목표 비트레이트 (41) 가 512 Kbps 와 동일하거나 또는 더 많을 때) 더 많은 백그라운드 및/또는 포그라운드 채널들을 선택할 수도 있다. 일 양태에서, numHOATransportChannels 는 8 로 설정될 수도 있으며, 한편 MinAmbHOAorder 는 비트스트림의 헤더 섹션에서 1 로 설정될 수도 있다. 이 시나리오에서, 매 프레임에서, 4개의 채널들이 음장의 백그라운드 또는 주변 부분을 표현하는데 담당될 수도 있지만, 다른 4 개의 채널들은 프레임 단위로, 채널의 유형에 따라서 변할 수 있다 - 예컨대, 추가적인 백그라운드/주변 채널 또는 포그라운드/지배적인 채널로서 사용될 수 있다. 포그라운드/지배적인 신호들은 위에서 설명한 바와 같이 벡터-기반 또는 방향 기반 신호들 중 하나일 수 있다.

[0067] 일부의 경우, 프레임에 대한 벡터-기반의 지배적인 신호들의 총 개수는 그 프레임의 비트스트림에서 ChannelType 인덱스가 01 인 횟수로 주어질 수도 있다. 상기 양태에서, (예컨대, 10 의 ChannelType 에 대응하는) 모든 추가적인 백그라운드/주변 채널에 대해, (처음 4개를 넘어서는) 가능한 HOA 계수들 중 어느 HOA 계수의 대응하는 정보가 그 채널에 표시될 수도 있다. 제 4 차수 HOA 콘텐츠에 대한, 정보는 HOA 계수들 5-25 를 표시하는 인덱스일 수도 있다. 처음 4개의 주변 HOA 계수들 1-4 는 minAmbHOAorder 가 1 로 설정될 때는 언제나 전송될 수도 있으며, 따라서 오디오 인코딩 디바이스는 단지 5-25 의 인덱스를 가지는 추가적인 주변 HOA 계수 중 하나만을 표시해야 할 수도 있다. 정보는 따라서 "CodedAmbCoeffIdx" 로서 표시될 수도 있는, (제 4 차수 콘텐츠에 대해) 5 비트 신택스 엘리먼트를 이용하여 전송될 수 있다.

[0068] 예시하기 위하여, 일 예로서 minAmbHOAorder 가 1 로 설정되고 6 의 인덱스를 가지는 추가적인 주변 HOA 계수가 비트스트림 (21) 을 통해서 전송된다고 가정한다. 이 예에서, 1 의 minAmbHOAorder 는 주변 HOA 계수들이 1, 2, 3 및 4 의 인덱스를 갖는다는 것을 나타낸다. 오디오 인코딩 디바이스 (20) 는 주변 HOA 계수들이 이 예에서 $(minAmbHOAorder + 1)^2$ 또는 4 보다 작거나 또는 동일한 인덱스를 가지기 때문에 주변 HOA 계수들을 선택할 수도 있다. 오디오 인코딩 디바이스 (20) 는 비트스트림 (21) 에서 1, 2, 3 및 4 의 인덱스들과 연관된 주변 HOA 계수들을 규정할 수도 있다. 오디오 인코딩 디바이스 (20) 는 또한 비트스트림에서 6 의 인덱스를 가지는 추가적인 주변 HOA 계수를 10 의 ChannelType 을 가지는 additionalAmbientHOAchannel 로서 규정할 수도 있다. 오디오 인코딩 디바이스 (20) 는 CodedAmbCoeffIdx 신택스 엘리먼트를 이용하여 인덱스를 규정할 수도 있다. 실제적인 이유로, CodedAmbCoeffIdx 엘리먼트는 1-25 중에서 인덱스들 모두를 규정할 수도 있다. 그러나, minAmbHOAorder 가 1 로 설정되기 때문에, 오디오 인코딩 디바이스 (20) 는 (처음 4개의 인덱스들이 minAmbHOAorder 신택스 엘리먼트를 통해서 비트스트림 (21) 에 규정된 것으로 알려져 있기 때문에) 처음 4개의 인덱스들 중 임의의 인덱스를 규정하지 않을 수도 있다. 어쨌든, 오디오 인코딩 디바이스 (20) 가 (처음 4개에 대한) minAmbHOAorder 및 (추가적인 주변 HOA 계수에 대한) CodedAmbCoeffIdx 를 통해서 5개의 주변 HOA 계수들을 규정하기 때문에, 오디오 인코딩 디바이스 (20) 는 1, 2, 3, 4 및 6 의 인덱스를 가지는 주변 HOA 계수들과 연관된 대응하는 V-벡터 엘리먼트들을 규정하지 않을 수도 있다. 그 결과, 오디오 인코딩 디바이스 (20) 는 엘리먼트들 [5, 7:25] 을 가지는 V-벡터를 규정할 수도 있다.

[0069] 제 2 양태에서, 포그라운드/지배적인 신호들의 모두는 벡터-기반의 신호들이다. 이 제 2 양태에서, 포그라운드/지배적인 신호들의 총 개수는 nFG = numHOATransportChannels - $[(MinAmbHOAorder +1)^2$ + additionalAmbientHOAchannel 의 각각] 으로 주어질 수도 있다.

[0070] 음장 분석 유닛 (44) 은 백그라운드 채널 정보 (43) 및 HOA 계수들 (11) 을 백그라운드 (BG) 선택 유닛 (36) 으로, 백그라운드 채널 정보 (43) 를 계수 감소 유닛 (46) 및 비트스트림 발생 유닛 (42) 으로, 그리고 nFG (45) 를 포그라운드 선택 유닛 (36) 으로 출력한다.

[0071] 백그라운드 선택 유닛 (48) 은 백그라운드 채널 정보 (예컨대, 백그라운드 음장 (N_{BG}) 및 개수 (nBGa) 및 전송할 추가적인 BG HOA 채널들의 인덱스들 (i)) 에 기초하여 백그라운드 또는 주변 HOA 계수들 (47) 을 결정하도록 구성된 유닛을 나타낼 수도 있다. 예를 들어, N_{BG} 가 1 과 동일할 때, 백그라운드 선택 유닛 (48) 은 1 과 동일하거나 또는 미만인 차수를 가지는 오디오 프레임의 각각의 샘플에 대해 HOA 계수들 (11) 을 선택할 수도

있다. 백그라운드 선택 유닛 (48) 은 이 예에서, 그후 인덱스들 (i) 중 하나를 추가적인 BG HOA 계수들로서 식별된 인덱스를 가지는 HOA 계수들 (11) 을 선택할 수도 있으며, nBGa 가 도 2 및 도 4 의 예에 나타낸 오디오 디코딩 디바이스 (24) 와 같은, 오디오 디코딩 디바이스로 하여금, 비트스트림 (21) 으로부터 백그라운드 HOA 계수들 (47) 을 파싱하도록 하기 위해서 비트스트림 (21) 에 규정되도록, 비트스트림 발생 유닛 (42) 에 제공된다. 백그라운드 선택 유닛 (48) 은 그후 주변 HOA 계수들 (47) 을 에너지 보상 유닛 (38) 으로 출력할 수도 있다. 주변 HOA 계수들 (47) 은 치수들 D: M x $[(N_{BG}+1)^2 + nBGa]$ 을 가질 수도 있다. 주변 HOA 계수들 (47) 은 또한 "주변 HOA 계수들 (47)" 로서 지칭될 수도 있으며, 여기서, 주변 HOA 계수들 (47) 의 각각은 음향심리 오디오 코더 유닛 (40) 에 의해 인코딩될 별개의 주변 HOA 채널 (47) 에 대응한다.

[0072] 포그라운드 선택 유닛 (36) 은 (포그라운드 벡터들을 식별하는 하나 이상의 인덱스들을 나타낼 수도 있는) nFG (45) 에 기초하여 음장의 포그라운드 또는 특유한 구성요소들을 나타내는 재정리된 US[k] 매트릭스 (33') 및 재정리된 V[k] 매트릭스 (35') 를 선택하도록 구성된 유닛을 나타낼 수도 있다. 포그라운드 선택 유닛 (36) 은 (재정리된 US[k]1, …, nFG (49), FG1, …, nfG[k] (49), 또는 $X_{PS}^{(1..nFG)}(k)$ (49) 로서 표시될 수도 있는) nFG 신호들 (49) 을 음향심리 오디오 코더 유닛 (40) 으로 출력할 수도 있으며, 여기서, nFG 신호들 (49) 은 치수들 D: M x nFG 을 가지며 각각 모노-오디오 오브젝트들을 나타낼 수도 있다. 포그라운드 선택 유닛 (36) 은 또한 음장의 포그라운드 구성요소들에 대응하는 재정리된 V[k] 매트릭스 (35') (또는, $v^{(1..nFG)}(k)$ (35')) 를 시공간적 내삽 유닛 (50) 으로 출력할 수도 있으며, 여기서, 포그라운드 구성요소들에 대응하는 재정리된 V[k] 매트릭스 (35') 의 서브세트는 치수들 D: $(N+1)^2$ x nFG 를 가지는 ($\overline{V}_{1,\ldots,nFG}[k]$ 로서 수학적으로 표시될 수도 있는) 포그라운드 V[k] 매트릭스 (51_k) 로서 표시될 수도 있다.

[0073] 에너지 보상 유닛 (38) 은 백그라운드 선택 유닛 (48) 에 의한 HOA 채널들 중 여러 HOA 채널의 제거로 인한 에너지 손실을 보상하기 위해 주변 HOA 계수들 (47) 에 대해 에너지 보상을 수행하도록 구성된 유닛을 나타낼 수도 있다. 에너지 보상 유닛 (38) 은 재정리된 US[k] 매트릭스 (33'), 재정리된 V[k] 매트릭스 (35'), nFG 신호들 (49), 포그라운드 V[k] 벡터들 (51_k) 및 주변 HOA 계수들 (47) 중 하나 이상에 대해 에너지 분석을 수행하고 그후 에너지 분석에 기초하여 에너지 보상을 수행하여 에너지 보상된 주변 HOA 계수들 (47') 을 발생시킬 수도 있다. 에너지 보상 유닛 (38) 은 에너지 보상된 주변 HOA 계수들 (47') 을 음향심리 오디오 코더 유닛 (40) 으로 출력할 수도 있다.

[0074] 시공간적 내삽 유닛 (50) 은 k 번째 프레임에 대한 포그라운드 V[k] 벡터들 (51_k) 및 포그라운드 이전 프레임 (따라서, k-1 표기) 에 대한 V[k-1] 벡터들 (51_{k-1}) 을 수신하고 시공간적 내삽을 수행하여 내삽된 포그라운드 V[k] 벡터들을 발생시키도록 구성된 유닛을 나타낼 수도 있다. 시공간적 내삽 유닛 (50) 은 nFG 신호들 (49) 을 포그라운드 V[k] 벡터들 (51_k) 과 재결합하여 재정리된 포그라운드 HOA 계수들을 복원할 수도 있다. 시공간적 내삽 유닛 (50) 은 그후 재정리된 포그라운드 HOA 계수들을 내삽된 V[k] 벡터들로 나눠서, 내삽된 nFG 신호들 (49') 을 발생시킬 수도 있다. 시공간적 내삽 유닛 (50) 은, 또한 오디오 디코딩 디바이스 (24) 와 같은, 오디오 디코딩 디바이스가 내삽된 포그라운드 V[k] 벡터들을 발생시켜 포그라운드 V[k] 벡터들 (51_k) 을 복원할 수 있도록 내삽된 포그라운드 V[k] 벡터들을 발생시키는데 사용된 포그라운드 V[k] 벡터들 (51_k) 을 출력할 수도 있다. 내삽된 포그라운드 V[k] 벡터들을 발생시키는데 사용되는 포그라운드 V[k] 벡터들 (51_k) 은 나머지 포그라운드 V[k] 벡터들 (53) 로서 표시된다. 동일한 V[k] 및 V[k-1] 이 (내삽된 벡터들 V[k] 을 생성하기 위해) 인코더 및 디코더에서 사용되도록 보장하기 위해, 벡터들의 양자화된/역양자화된 버전들이 인코더 및 디코더에서 사용될 수도 있다.

[0075] 동작 시, 시공간적 내삽 유닛 (50) 은 제 1 프레임에 포함된 제 1 복수의 HOA 계수들 (11) 의 부분의 제 1 분해, 예컨대, 포그라운드 V[k] 벡터들 (51_k) 및 제 2 프레임에 포함된 제 2 복수의 HOA 계수들 (11) 의 부분의 제 2 분해, 예컨대, 포그라운드 V[k] 벡터들 (51_{k-1}) 로부터 제 1 오디오 프레임의 하나 이상의 서브-프레임들을 내삽하여, 하나 이상의 서브-프레임들에 대해, 분해된 내삽된 구면 고조파 계수들을 발생시킬 수도 있다.

[0076] 일부 예들에서, 제 1 분해는 HOA 계수들 (11) 의 부분의 우측-특이 벡터들을 나타내는 제 1 포그라운드 V[k] 벡

터들 (51_k) 을 포함한다. 이와 유사하게, 일부 예들에서, 제 2 분해는 HOA 계수들 (11) 의 부분의 우측-특이 벡터들을 나타내는 제 2 포그라운드 V[k] 벡터들 (51_k) 을 포함한다.

[0077] 다시 말해서, 구면 고조파들-기반의 3D 오디오는 구 상의 직교 기저 함수들의 관점에서 3D 압력 장의 파라미터 표현일 수도 있다. 그 표현의 차수 N 이 더 높을 수록, 잠재적으로 공간 해상도가 더 높아지며 그리고 종종 (총 $(N+1)^2$ 계수들에 대한) 구면 고조파들 (SH) 계수들의 개수가 더 커진다. 많은 응용들에 있어, 계수들을 효율적으로 전송하고 저장할 수 있도록 하기 위해서 계수들의 대역폭 압축이 요구될 수도 있다. 본 개시물에서 알려주는 기법들은 특이 값 분해 (SVD) 를 이용한 프레임-기반, 차원수 감소 프로세스를 제공할 수도 있다. SVD 분석은 계수들의 각각의 프레임을 3개의 매트릭스들 U, S 및 V 로 분해할 수도 있다. 일부 예들에서, 이 기법들은 US[k] 매트릭스에서의 벡터들 중 일부를 기본적인 음장의 포그라운드 구성요소들로서 취급할 수도 있다. 그러나, 이와 같이 취급될 때, (US[k] 매트릭스에서의) 벡터들은, 설령 그들이 동일한 특유의 오디오 성분을 나타내더라도, 프레임들간에 불연속적이다. 불연속들은 구성요소들이 변환-오디오-코더들을 통해서 공급될 때 유의한 아티팩트들을 초래할 수도 있다.

[0078] 일부 양태들에서, 시공간적 내삽은 V 매트릭스가 구면 고조파들 도메인에서 직교 공간 축들로서 해석될 수 있다는 관측에 의존할 수도 있다. U[k] 매트릭스는 기저 함수들의 관점에서 구면 고조파들 (HOA) 데이터의 투영을 나타낼 수도 있으며, 여기서, 불연속성은 매 프레임 마다 변하는 직교 공간 축 (V[k]) 에 기인할 수 있으며 - 따라서 그들 스스로 불연속적이다. 이것은 푸리에 변환과 같은, 일부 다른 분해들과는 다르며, 여기서 기저 함수들이 일부 예들에서, 프레임들 간에 일정하다. 이들 용어들에서, SVD 는 매칭 추적 알고리즘으로서 간주될 수도 있다. 시공간적 내삽 유닛 (50) 은 프레임들간에, 그들간에 내삽함으로써, 기저 함수들 (V[k]) 사이에, 연속성을 잠재적으로 유지하기 위해, 내삽을 수행할 수도 있다.

[0079] 위에서 언급한 바와 같이, 내삽은 샘플들에 대해 수행될 수도 있다. 이 경우는 서브-프레임들이 샘플들의 단일 세트를 포함할 때에 상기 설명에서 일반화된다. 샘플들을 통한 그리고 서브-프레임들을 통한 내삽 양쪽의 경우, 내삽 동작은 다음 방정식의 유형을 취할 수도 있다:

[0080] $$\overline{v(l)} = w(l)v(k) + \big(1 - w(l)\big)v(k-1).$$

[0081] 상기 방정식에서, 내삽은 일 양태에서 인접 프레임들 k 및 k-1 로부터의 V-벡터들을 나타낼 수 있는 단일 V-벡터 v(k-1) 로부터 단일 V-벡터 v(k) 에 대해 수행될 수도 있다. 상기 방정식에서, l 는, 내삽이 수행중인 해상도를 나타내며, 여기서, l 는 정수 샘플을 나타낼 수도 있으며 l = 1, …, T 를 나타낼 수도 있다 (여기서, T 는 내삽이 수행중이며 출력된 내삽된 벡터들, $\overline{v(l)}$ 이 요구되는 샘플들의 길이이며 또한 그 프로세스의 출력이 벡터들의 l 를 발생시킨다는 것을 나타낸다). 대안적으로, l 는 다수의 샘플들로 이루어지는 서브-프레임들을 나타낼 수 있다. 예를 들어, 프레임이 4개의 서브-프레임들로 분할될 때, l 는 서브-프레임들의 각각의 하나에 대해 1, 2, 3 및 4 의 값들을 포함할 수도 있다. l 의 값은 내삽 동작이 디코더에서 복제될 수 있도록, 비트스트림을 통해서 "CodedSpatialInterpolationTime" 로 불리는 필드로서 시그널링될 수도 있다.

w(l) 는 내삽 가중치들의 값들을 포함할 수도 있다. 내삽이 선형일 때, w(l) 는 0 과 1 사이에서 선형적으로 그리고 단조적으로 (monotonically) l 의 함수로서 변할 수도 있다. 다른 경우, w(l) 는 0 과 1 사이에서 비선형적이지만 그러나 (올림 코사인 (raised cosine) 의 1/4 사이클과 같은) 단조 방식으로 l 의 함수로서 변할 수도 있다. 함수, w(l) 는, 함수들의 몇개의 상이한 가능성들 사이에 인덱싱될 수도 있으며, 동일한 내삽 동작이 디코더에서 복제될 수 있도록 "SpatialInterpolationMethod" 로 불리는 필드로서 비트스트림에서 시그널링될 수도 있다. w(l) 가 0 에 가까운 값을 가질 때, 출력, $\overline{v(l)}$ 은, v(k-1) 에 의해 크게 가중되거나 또는 영향을 받을 수도 있다. 반면 w(l) 가 1 에 가까운 값을 가질 때, 그것은 출력, $\overline{v(l)}$ 이, v(k-1) 에 의해 크게 가중되거나 또는 영향을 받도록 보장한다.

[0082] 계수 감소 유닛 (46) 은 백그라운드 채널 정보 (43) 에 기초하여 나머지 포그라운드 V[k] 벡터들 (53) 에 대해 계수 감소를 수행하여 감소된 포그라운드 V[k] 벡터들 (55) 을 양자화 유닛 (52) 으로 출력하도록 구성된 유닛을 나타낼 수도 있다. 감소된 포그라운드 V[k] 벡터들 (55) 은 치수들 D: $[(N+1)^2 - (N_{BG}+1)^2\text{-}BG_{TOT}] \times nFG$ 를 가질 수도 있다.

[0083] 계수 감소 유닛 (46) 은 이 점에서, 나머지 포그라운드 V[k] 벡터들 (53) 에서의 계수들의 개수를 감소시키도록

구성된 유닛을 나타낼 수도 있다. 다시 말해서, 계수 감소 유닛 (46) 은 거의 없거나 전혀 없는 방향 정보를 가지는 (나머지 포그라운드 V[k] 벡터들 (53) 을 형성하는) 포그라운드 V[k] 벡터들에서의 계수들을 제거하도록 구성된 유닛을 나타낼 수도 있다. 위에서 설명된 바와 같이, 일부 예들에서, (N_{BG} 로서 표시될 수도 있는) 제 1 및 제로 차수 기저 함수들에 대응하는 별개의, 또는, 즉, 포그라운드 V[k] 벡터들의 계수들은 적은 방향 정보를 제공하며, 따라서 ("계수 감소" 로서 지칭될 수도 있는 프로세스를 통해서) 포그라운드 V-벡터들로부터 제거될 수 있다. 이 예에서, $[(N_{BG}+1)^2+1, (N+1)^2]$ 의 세트로부터, N_{BG} 에 대응하는 계수들을 식별할 뿐만 아니라 (변수 TotalOfAddAmbHOAChan 에 의해 표시될 수도 있는) 추가적인 HOA 채널들을 식별하기 위해 더 큰 유연성이 제공될 수도 있다. 음장 분석 유닛 (44) 은 HOA 계수들 (11) 을 분석하여, $(N_{BG}+1)^2$ 뿐만 아니라 백그라운드 채널 정보 (43) 로서 일괄하여 지칭될 수도 있는 TotalOfAddAmbHOAChan 을 식별하는, BG_{TOT}, 을 결정할 수도 있다. 계수 감소 유닛 (46) 은 그후 나머지 포그라운드 V[k] 벡터들 (53) 로부터 $(N_{BG}+1)^2$ 및 TotalOfAddAmbHOAChan 에 대응하는 계수들을 제거하여, 감소된 포그라운드 V[k] 벡터들 (55) 로서 또한 지칭될 수도 있는 사이즈 $((N+1)^2 - (BG_{TOT}) \times nFG$ 의 더 작은 차원 V[k] 매트릭스 (55) 를 발생시킬 수도 있다.

[0084] 즉, 공개번호 제 WO 2014/194099호에 언급한 바와 같이, 계수 감소 유닛 (46) 은 부 채널 정보 (57) 에 대한 신택스 엘리먼트들을 발생시킬 수도 있다. 예를 들어, 계수 감소 유닛 (46) 은 복수의 구성 모드들 중 어느 모드가 선택되었는지를 표시하는 신택스 엘리먼트를 (하나 이상의 프레임들을 포함할 수도 있는) 액세스 유닛의 헤드에 규정할 수도 있다. 액세스 유닛 단위로 규정되는 것으로 설명되지만, 계수 감소 유닛 (46) 은 프레임 단위 또는 (전체 비트스트림에 대해 한번과 같은) 임의의 다른 주기적인 단위 또는 비-주기적인 단위로 신택스 엘리먼트를 규정할 수도 있다. 어쨌든, 신택스 엘리먼트는 특유한 구성요소의 방향 양태들을 나타내기 위해 그 감소된 포그라운드 V[k] 벡터들 (55) 의 계수들의 비-제로 세트를 규정하는데 3개의 구성 모드들 중 어느 구성 모드가 선택되는지를 표시하는 2 비트들을 포함할 수도 있다. 신택스 엘리먼트는 "CodedVVecLength" 로서 표시될 수도 있다. 이와 같이, 계수 감소 유닛 (46) 은 비트스트림 (21) 에 그 감소된 포그라운드 V[k] 벡터들 (55) 을 규정하는데 3개의 구성 모드들 중 어느 구성 모드가 사용되었는지를 비트스트림에서 시그널링하거나 또는 아니면 규정할 수도 있다.

[0085] 예를 들어, 3개의 구성 모드들이 (이 문서에서 추후에 참조되는) VVecData 에 대한 신택스 테이블에 제시될 수도 있다. 그 예에서, 구성 모드들은 다음과 같다: (모드 0), 완전한 V-벡터 길이가 VVecData 필드에서 송신된다; (모드 1), 주변 HOA 계수들에 대한 계수들의 최소 개수와 연관되는 V-벡터의 엘리먼트들 및 추가적인 HOA 채널들을 포함한 V-벡터의 모든 엘리먼트들이 송신되지 않는다; 및 (모드 2), 주변 HOA 계수들에 대한 계수들의 최소 개수와 연관되는 V-벡터의 엘리먼트들이 송신되지 않는다. VVecData 의 신택스 테이블은 스위치 (switch) 및 케이스 (case) 스테이트먼트와 함께 모드들을 예시한다. 3개의 구성 모드들에 대해 설명되지만, 본 기법들은 3개의 구성 모드들에 한정되지 않아야 하며 단일 구성 모드 또는 복수의 모드들을 포함한, 임의 개수의 구성 모드들을 포함할 수도 있다. 공개번호 제 WO 2014/194099호는 4개의 모드들을 가진 상이한 예를 제공한다. 계수 감소 유닛 (46) 은 또한 부 채널 정보 (57) 에서의 다른 신택스 엘리먼트로서 플래그 (63) 를 규정할 수도 있다.

[0086] 양자화 유닛 (52) 은 감소된 포그라운드 V[k] 벡터들 (55) 을 압축하여 코딩된 포그라운드 V[k] 벡터들 (57) 을 발생시키기 위해 임의 유형의 양자화를 수행하여 코딩된 포그라운드 V[k] 벡터들 (57) 을 비트스트림 발생 유닛 (42) 으로 출력하도록 구성된 유닛을 나타낼 수도 있다. 동작 시, 양자화 유닛 (52) 은 음장의 공간 구성요소, 즉, 이 예에서는, 감소된 포그라운드 V[k] 벡터들 (55) 중 하나 이상을 압축하도록 구성된 유닛을 압축하도록 구성된 유닛을 나타낼 수도 있다. 예의 목적들을 위해, 감소된 포그라운드 V[k] 벡터들 (55) 은 계수 감소의 결과로서, (음장의 제 4 차수 HOA 표현을 암시하는) 25 개 미만인 엘리먼트들 각각을 가지는 2개의 로우 벡터들을 포함하는 것으로 가정된다. 2개의 로우 벡터들에 대해 설명되지만, 임의 개수의 벡터들이 $(n+1)^2$ 까지 그 감소된 포그라운드 V[k] 벡터들 (55) 에 포함될 수도 있으며, 여기서, n 은 음장의 HOA 표현의 차수를 나타낸다. 더욱이, 스칼라 및/또는 엔트로피 양자화를 수행하는 것으로 아래에서 설명되지만, 양자화 유닛 (52) 은 감소된 포그라운드 V[k] 벡터들 (55) 의 압축을 초래하는 임의 유형의 양자화를 수행할 수도 있다.

[0087] 양자화 유닛 (52) 은 감소된 포그라운드 V[k] 벡터들 (55) 을 수신하고 압축 방식을 수행하여 코딩된 포그라운드 V[k] 벡터들 (57) 을 발생시킬 수도 있다. 압축 방식은 벡터 또는 데이터의 엘리먼트들을 압축하는 임의의 상상가능한 압축 방식을 일반적으로 수반할 수도 있으며, 아래에서 좀더 자세하게 설명된 예에 한정되지 않

아야 한다. 양자화 유닛 (52) 은 일 예로서, 감소된 포그라운드 V[k] 벡터들 (55) 의 각각의 엘리먼트의 부동 소수점 표현들을 감소된 포그라운드 V[k] 벡터들 (55) 의 각각의 엘리먼트의 정수 표현들로 변환하는 것, 감소된 포그라운드 V[k] 벡터들 (55) 의 정수 표현들의 균일한 양자화, 및 나머지 포그라운드 V[k] 벡터들 (55) 의 양자화된 정수 표현들의 범주화 및 코딩 중 하나 이상을 포함하는 압축 방식을 수행할 수도 있다.

[0088] 일부 예들에서, 압축 방식의 하나 이상의 프로세스들 중 몇 개는, 일 예로서, 최종 비트스트림 (21) 에 대한 목표 비트레이트 (41) 를 달성하거나 또는 거의 달성하기 위해 파라미터들에 의해 동적으로 제어될 수도 있다. 감소된 포그라운드 V[k] 벡터들 (55) 의 각각이 서로에 대해 직교하다고 가정하면, 감소된 포그라운드 V[k] 벡터들 (55) 의 각각은 독립적으로 코딩될 수도 있다. 일부 예들에서, 아래에서 좀더 자세히 설명하는 바와 같이, 각각의 감소된 포그라운드 V[k] 벡터들 (55) 의 각각의 엘리먼트는 (여러 서브-모드들에 의해 정의된) 동일한 코딩 모드를 이용하여 코딩될 수도 있다.

[0089] 공개 번호 제 WO 2014/194099호에 설명된 바와 같이, 양자화 유닛 (52) 은 감소된 포그라운드 V[k] 벡터들 (55) 을 압축하기 위해 스칼라 양자화 및/또는 Huffman 인코딩을 수행하여, 부 채널 정보 (57) 로서 또한 지칭될 수도 있는 코딩된 포그라운드 V[k] 벡터들 (57) 을 출력할 수도 있다. 부 채널 정보 (57) 는 나머지 포그라운드 V[k] 벡터들 (55) 을 코딩하는데 사용되는 신택스 엘리먼트들을 포함할 수도 있다.

[0090] 더욱이, 스칼라 양자화의 유형에 대해서 설명되지만, 양자화 유닛 (52) 은 벡터 양자화 또는 임의의 다른 유형의 양자화를 수행할 수도 있다. 일부의 경우, 양자화 유닛 (52) 은 벡터 양자화와 스칼라 양자화 사이에 스위칭할 수도 있다. 상기 설명된 스칼라 양자화 동안, 양자화 유닛 (52) 은 (프레임-대-프레임에서와 같이 연속적인) 2개의 연속적인 V-벡터들 사이의 차이를 계산하고 그 차이 (또는, 즉, 잔차) 를 코딩할 수도 있다. 이 스칼라 양자화는 이전에 규정된 벡터 및 차이 신호에 기초하여 예측 코딩하는 유형을 나타낼 수도 있다. 벡터 양자화는 이러한 차이 코딩을 수반하지 않는다.

[0091] 다시 말해서, 양자화 유닛 (52) 은 입력 V-벡터 (예컨대, 감소된 포그라운드 V[k] 벡터들 (55) 중 하나) 를 수신하고 상이한 유형들의 양자화를 수행하여, 입력 V-벡터를 입력하는데 사용될 양자화의 유형들 중 하나를 선택할 수도 있다. 양자화 유닛 (52) 은 일 예로서, 벡터 양자화, Huffman 코딩에 의하지 않는 스칼라 양자화 및 Huffman 코딩에 의한 스칼라 양자화를 수행할 수도 있다.

[0092] 이 예에서, 양자화 유닛 (52) 은 벡터 양자화 모드에 따라서 입력 V-벡터를 벡터 양자화하여, 벡터-양자화된 V-벡터를 발생시킬 수도 있다. 벡터 양자화된 V-벡터는 입력 V-벡터를 표시하는 벡터-양자화된 가중 값들을 포함할 수도 있다. 벡터-양자화된 가중 값들은, 일부 예들에서, 양자화 코드워드들의 양자화 코드북에서 양자화 코드워드 (즉, 양자화 벡터) 를 가리키는 하나 이상의 양자화 인덱스들로서 표현될 수도 있다. 양자화 유닛 (52) 은, 벡터 양자화를 수행하도록 구성될 때, 코드 벡터들 (63) ("CV (63)") 에 기초하여 감소된 포그라운드 V[k] 벡터들 (55) 의 각각을 코드 벡터들의 가중 총합으로 분해할 수도 있다. 양자화 유닛 (52) 은 코드 벡터들 (63) 중 선택된 코드 벡터들의 각각에 대해 가중 값들을 발생시킬 수도 있다.

[0093] 양자화 유닛 (52) 은 다음으로, 가중 값들의 선택된 서브세트를 발생시키기 위해 가중 값들의 서브세트를 선택할 수도 있다. 예를 들어, 양자화 유닛 (52) 은 가중 값들의 세트로부터 Z 최대-크기 가중 값들을 선택하여, 가중 값들의 선택된 서브세트를 발생시킬 수도 있다. 일부 예들에서, 양자화 유닛 (52) 은 그 선택된 가중 값들을 재정리하여, 가중 값들의 선택된 서브세트를 발생시킬 수도 있다. 예를 들어, 양자화 유닛 (52) 은 선택된 가중 값들을 최고-크기 가중 값에서 시작하여 최저-크기 가중 값에서 끝나는 크기에 기초하여 재정리할 수도 있다.

[0094] 벡터 양자화를 수행할 때, 양자화 유닛 (52) 은 양자화 코드북 중에서 Z-구성요소 벡터를 선택하여 Z 가중 값들을 표시할 수도 있다. 다시 말해서, 양자화 유닛 (52) 은 Z 가중 값들을 벡터 양자화하여, Z 가중 값들을 표시하는 Z-구성요소 벡터를 발생시킬 수도 있다. 일부 예들에서, Z 는 단일 V-벡터를 표시하기 위해 양자화 유닛 (52) 에 의해 선택된 가중 값들의 개수에 대응할 수도 있다. 양자화 유닛 (52) 은 Z 가중 값들을 표시하기 위해 선택된 Z-구성요소 벡터를 표시하는 데이터를 발생시키고, 이 데이터를 비트스트림 발생 유닛 (42) 로, 코딩된 가중치들 (57) 로서 제공할 수도 있다. 일부 예들에서, 양자화 코드북은 인덱싱된 복수의 Z-구성요소 벡터들을 포함할 수도 있으며, Z-구성요소 벡터를 표시하는 데이터는 선택된 벡터를 가리키는 양자화 코드북으로의 인덱스 값일 수도 있다. 이러한 예들에서, 디코더는 인덱스 값을 디코딩하기 위해, 유사하게 인덱스된 양자화 코드북을 포함할 수도 있다.

[0095] 수학적으로, 감소된 포그라운드 V[k] 벡터들 (55) 의 각각은 다음 수식에 기초하여 표현될 수도 있다:

$$V \approx \sum_{j=1}^{J} \omega_j \Omega_j$$

[0096] (1)

[0097] 여기서, Ω_j 는 코드 벡터들 ($\{\Omega_j\}$) 의 세트에서의 j번째 코드 벡터를 표시하며, ω_j 는 가중치들 ($\{\omega_j\}$) 의 세트에서의 j번째 가중치를 나타내며, V 는 V-벡터 코딩 유닛 (52) 에 의해 표현되고, 분해되고, 및/또는 코딩되고 있는 V-벡터에 대응하며, J 는 V 를 표현하는데 사용되는 코드 벡터들의 개수 및 가중치들의 개수를 표시한다. 수식 (1) 의 우변은 가중치들 ($\{\omega_j\}$) 의 세트 및 코드 벡터들 ($\{\Omega_j\}$) 의 세트를 포함하는 코드 벡터들의 가중 총합을 표시할 수도 있다.

[0098] 일부 예들에서, 양자화 유닛 (52) 은 다음 방정식에 기초하여 가중 값들을 결정할 수도 있다:

$$\omega_k = V \Omega_k^T$$

[0099] (2)

[0100] 여기서, Ω_k^T 는 코드 벡터들 ($\{\Omega_k\}$) 의 세트에서의 k번째 코드 벡터의 전치를 나타내며, V 는 양자화 유닛 (52) 에 의해 표시되고, 분해되고, 및/또는 코딩되고 있는 V-벡터에 대응하며, ω_k 는 가중치들 ($\{\omega_k\}$) 의 세트에서의 k번째 가중치를 나타낸다.

[0101] 25 개의 가중치들 및 25 개의 코드 벡터들이 V-벡터, V_{FG} 를 표시하는데 사용되는 예를 고려한다. 이러한 V_{FG} 의 분해는 다음과 같이 쓸 수도 있다:

$$V_{FG} \approx \sum_{j=1}^{25} \omega_j \Omega_j$$

[0102] (3)

[0103] 여기서, Ω_j 는 코드 벡터들 ($\{\Omega_j\}$) 의 세트에서의 j번째 코드 벡터를 표시하며, ω_j 는 가중치들 ($\{\omega_j\}$) 의 세트에서의 j번째 가중치를 표시하며, V_{FG} 는 양자화 유닛 (52) 에 의해 표시되고, 분해되고, 및/또는 코딩되고 있는 V-벡터에 대응한다.

[0104] 코드 벡터들 ($\{\Omega_j\}$) 의 세트가 직교한 예들에서, 다음 수식이 적용될 수도 있다:

$$\Omega_j \Omega_k^T = \begin{cases} 1 \text{ for } j = k \\ 0 \text{ for } j \neq k \end{cases}$$

[0105] (4)

[0106] 이러한 예들에서, 방정식 (3) 의 우변은 다음과 같이 단순화될 수도 있다:

$$V_{FG} \Omega_k^T \approx \left(\sum_{j=1}^{25} \omega_j \Omega_j \right) \Omega_k^T = \omega_k$$

[0107] (5)

[0108] 여기서, ω_k 는 코드 벡터들의 가중 총합에서의 k번째 가중치에 대응한다.

[0109] 방정식 (3) 에서 사용되는 코드 벡터들의 예시적인 가중 총합에 대해, 양자화 유닛 (52) 은 (방정식 (2) 와 유사한) 방정식 (5) 을 이용하여 코드 벡터들의 가중 총합에서의 가중치들의 각각에 대해 가중 값들을 계산할 수도 있으며, 최종 가중치들은 다음과 같이 나타낼 수도 있다:

[0110] $\{\omega_k\}_{k=1,\cdots,25}$ (6)

[0111] 양자화 유닛 (52) 이 5개의 최대치들 가중 값들 (즉, 최대 값들 또는 절대값들을 가지는 가중치들) 을 선택하는 예를 고려한다. 양자화될 가중 값들의 서브세트는 다음과 같이 나타낼 수도 있다:

[0112] $\{\overline{\omega}_k\}_{k=1,\cdots,5}$ (7)

[0113] 가중 값들의 서브세트가 그들의 대응하는 코드 벡터들과 함께 다음 수식에 나타낸 바와 같이, V-벡터를 추정하는 코드 벡터들의 가중 총합을 형성하는데 사용될 수도 있다:

[0114] $$\overline{V}_{FG} \approx \sum_{j=1}^{5} \overline{\omega}_j \Omega_j$$ (8)

[0115] 여기서, Ω_j 는 코드 벡터들 ($\{\Omega_j\}$) 의 서브세트에서의 j번째 코드 벡터를 표시하며, $\overline{\omega}_j$ 는 가중치들 ($\{\overline{\omega}_j\}$) 의 서브세트에서의 j번째 가중치를 표시하며, $\overline{V}_{FG}$ 는 양자화 유닛 (52) 에 의해 분해되고 및/또는 코딩되고 있는 V-벡터에 대응하는 추정된 V-벡터에 대응한다. 수식 (1) 의 우변은 가중치들 ($\{\overline{\omega}_j\}$) 의 세트 및 코드 벡터들 ($\{\Omega_j\}$) 의 세트를 포함하는 코드 벡터들의 가중 총합을 표시할 수도 있다.

[0116] 양자화 유닛 (52) 은 가중 값들의 서브세트를 양자화하여, 다음과 같이 나타낼 수도 있는 양자화된 가중 값들을 발생시킬 수도 있다:

[0117] $\{\hat{\omega}_k\}_{k=1,\cdots,5}$ (9)

[0118] 양자화된 가중 값들이 그들의 대응하는 코드 벡터들과 함께 다음 수식에 나타낸 바와 같이, 추정된 V-벡터의 양자화된 버전을 표시하는 코드 벡터들의 가중 총합을 형성하기 위해 사용될 수도 있다:

[0119] $$\hat{V}_{FG} \approx \sum_{j=1}^{5} \hat{\omega}_j \Omega_j$$ (10)

[0120] 여기서, Ω_j 는 코드 벡터들 ($\{\Omega_j\}$) 의 서브세트에서의 j번째 코드 벡터를 표시하며, $\hat{\omega}_j$ 는 가중치들 ($\{\hat{\omega}_j\}$) 의 서브세트에서의 j번째 가중치를 표시하며, $\hat{V}_{FG}$ 는 양자화 유닛 (52) 에 의해 분해되고 및/또는 코딩되고 있는 V-벡터에 대응하는 추정된 V-벡터에 대응한다. 수식 (1) 의 우변은 가중치들 ($\{\hat{\omega}_j\}$) 의 세트 및 코드 벡터들 ($\{\Omega_j\}$) 의 세트를 포함하는 코드 벡터들의 서브세트의 가중 총합을 표시할 수도 있다.

[0121] (위에서 설명된 것과 대체로 동등한) 전술한 것의 대안적인 수정 (restatement) 은 다음과 같을 수도 있다. V-벡터들은 코드 벡터들의 미리 정의된 세트에 기초하여 코딩될 수도 있다. V-벡터들을 코딩하기 위해, 각각의 V-벡터는 코드 벡터들의 가중 총합으로 분해된다. 코드 벡터들의 가중 총합은 미리 정의된 코드 벡터들 및 연관된 가중치들의 k 개의 쌍들로 이루어진다:

[0122] $$V \approx \sum_{j=0}^{k} \omega_j \Omega_j$$ (11)

[0123] 여기서, Ω_j 는 미리 정의된 코드 벡터들 ($\{\Omega_j\}$) 의 세트에서의 j번째 코드 벡터를 표시하며, ω_j 는 미리

정의된 가중치들 ($\{\omega_j\}$) 의 세트에서의 j번째 실수 값의 가중치를 표시하며, k 는 최고 7 일 수 있는, 가수들의 인덱스에 대응하며, V 는 코딩될 V-벡터에 대응한다. k 의 선택은 인코더에 의존한다. 인코더가 2개 이상의 코드 벡터들의 가중 총합을 선택하면, 인코더가 선택할 수 있는 미리 정의된 코드 벡터들의 총 개수는 $(N+1)^2$ 이며, 여기서, 미리 정의된 코드 벡터들은 2014년 07월 25일로 기재된, 그리고 문서 번호 ISO/IEC DIS 23008-3 로 식별되는, ISO/IEC JTC 1/SC 29/WG 11 에 의한, "Information technology - High effeciency coding and media delivery in heterogeneous environments - Part 3: 3D audio"란 표제로 된, 3D 오디오 표준의 테이블들 F.3 내지 F.7 로부터 HOA 확장 계수들로서 유도된다. N 이 4 일 때, 32 개의 미리 정의된 방향들을 가지는 상기 언급된 3D 오디오 표준의 부속서 F.5 에서의 테이블이 사용된다. 모든 경우들에서, 가중치들 ω 의 절대값들은 상기 언급된 3D 오디오 표준의 테이블 F.12 에서의 테이블의 제 1 $k+1$ 칼럼들에서 발견되어 연관된 로우 개수 인덱스로 시그널링되는 미리 정의된 가중값들 $\hat{\omega}$ 에 대해 벡터-양자화된다.

[0124] 가중치들 ω 의 개수 부호들은 다음과 같이 별개로 코딩된다

[0125] $$s_j = \begin{cases} 1, & \omega_j \geq 0 \\ 0, & \omega_j < 0 \end{cases}. \quad (12)$$

[0126] 다시 말해서, 값 k 를 시그널링한 후, V-벡터는 $k+1$ 미리 정의된 코드 벡터들 $\{\Omega_j\}$ 를 가리키는 $k+1$ 인덱스들, 미리 정의된 가중 코드북에서 k 양자화된 가중치들 $\{\hat{\omega}_k\}$ 을 가리키는 하나의 인덱스, 및 $k+1$ 개수 부호 값들 s_j 로 인코딩된다:

[0127] $$\hat{V} = \sum_{j=0}^{k} (2s_j - 1)\hat{\omega}_j \Omega_j . \quad (13)$$

[0128] 인코더가 하나의 코드 벡터의 가중 총합을 선택하면, 상기 언급된 3D 오디오 표준의 테이블 F.8 로부터 유도된 코드북이 상기 언급된 3D 오디오 표준의 테이블 F.11 의 테이블에서 절대 가중값들 $\hat{\omega}$ 과 조합하여 사용되며, 여기서, 이들 테이블들의 양쪽이 아래에 나타낸다. 또한, 가중값 ω 의 개수 부호는 별개로 코딩될 수도 있다. 양자화 유닛 (52) 은 위에서 언급된 테이블들 F.3 내지 F.12 에 개시된 전술한 코드북들 중 어느 코브둑이 ("CodebkIdx" 로서 아래에 표시될 수도 있는) 코드북 인덱스 신택스 엘리먼트를 이용하여, 입력 V-벡터를 코딩하는데 사용되는지를 시그널링할 수도 있다. 양자화 유닛 (52) 은 또한 입력 V-벡터를 스칼라 양자화하여, 스칼라-양자화된 V-벡터를 Huffman 코딩 없이 출력 스칼라-양자화된 V-벡터를 발생시킬 수도 있다. 양자화 유닛 (52) 은 Huffman 코딩 스칼라 양자화 모드에 따라서 입력 V-벡터를 추가로 스칼라 양자화하여, Huffman-코딩된 스칼라-양자화된 V-벡터를 발생시킬 수도 있다. 예를 들어, 양자화 유닛 (52) 은 입력 V-벡터를 스칼라 양자화하여 스칼라-양자화된 V-벡터를 발생시키고, 스칼라-양자화된 V-벡터를 Huffman 코딩하여 출력 Huffman-코딩된 스칼라-양자화된 V-벡터를 발생시킬 수도 있다.

[0129] 일부 예들에서, 양자화 유닛 (52) 은 예측된 벡터 양자화의 유형을 수행할 수도 있다. 양자화 유닛 (52) 은 (양자화 모드를 표시하는, 하나 이상의 비트들, 예컨대, NbitsQ 신택스 엘리먼트에 의해 식별되는 것 처럼) 예측이 벡터 양자화를 위해 수행되는지 여부를 표시하는 하나 이상의 비트들 (예컨대, PFlag 신택스 엘리먼트) 을 비트스트림 (21) 에 규정함으로써, 벡터 양자화가 예측되는지 여부를 식별할 수도 있다.

[0130] 예측된 벡터 양자화를 예시하기 위해, 양자화 유닛 (42) 은 벡터 (예컨대, v-벡터) 의 코드 벡터-기반의 분해에 대응하는 가중 값들 (예컨대, 가중 값 크기들) 을 수신하고, 수신된 가중 값들에 기초하여 그리고 복원된 가중 값들 (예컨대, 하나 이상의 이전 또는 후속 오디오 프레임들로부터의 복원된 가중 값들) 에 기초하여 예측 가중 값들을 발생시키고, 그리고 예측 가중 값들의 세트들을 벡터-양자화하도록 구성될 수도 있다. 일부의 경우, 예측 가중 값들의 세트에서의 각각의 가중 값은 단일 벡터의 코드-벡터-기반의 분해에 포함된 가중 값에 대응할

수도 있다.

[0131] 양자화 유닛 (52) 은 벡터의 이전 또는 후속 코딩으로부터 가중 값 및 가중된 복원된 가중 값을 수신할 수도 있다. 양자화 유닛 (52) 은 가중 값 및 가중된 복원된 가중 값에 기초하여 예측 가중 값을 발생시킬 수도 있다. 양자화 유닛 (42) 은 가중 값에서 가중된 복원된 가중 값을 감산하여, 예측 가중 값을 발생시킬 수도 있다. 예측 가중 값은 대안적으로, 예를 들어, 잔차, 예측 잔차, 잔차 가중 값, 가중 값 차이, 에러, 또는 예측 에러로서 지칭될 수도 있다.

[0132] 가중 값은 대응하는 가중 값, $w_{i,j}$ 의 크기 (또는, 절대값) 인 $|w_{i,j}|$ 로서 표시될 수도 있다. 이와 같이, 가중 값은 대안적으로 가중 값 크기로서 또는 가중 값의 크기로서 지칭될 수도 있다. 가중 값, $w_{i,j}$ 는, i 번째 오디오 프레임에 대한 가중 값들의 순서정렬된 서브세트로부터 j번째 가중 값에 대응한다. 일부 예들에서, 가중 값들의 순서정렬된 서브세트는 가중 값들의 크기에 기초하여 순서화된 (예컨대, 최대 크기로부터 최소 크기까지 순서화된) 벡터 (예컨대, v-벡터) 의 코드 벡터-기반의 분해에서 가중 값들의 서브세트에 대응할 수도 있다.

[0133] 가중된 복원된 가중 값은 대응하는 복원된 가중 값, $\hat{w}_{i-1,j}$ 의 크기 (또는, 절대값) 에 대응하는 $|\hat{w}_{i-1,j}|$ 항을 포함할 수도 있다. 복원된 가중 값, $\hat{w}_{i-1,j}$ 은, (i-1)번째 오디오 프레임에 대한 복원된 가중 값들의 순서정렬된 서브세트에서 j번째 복원된 가중 값에 대응한다. 일부 예들에서, 복원된 가중 값들의 순서정렬된 서브세트 (또는, 세트) 는 복원된 가중 값들에 대응하는 양자화된 예측 가중 값들에 기초하여 발생될 수도 있다.

[0134] 양자화 유닛 (42) 은 또한 가중 인자, α_j 를 포함한다. 일부 예들에서, $\alpha_j = 1$ 이며, 이 경우 가중된 복원된 가중 값이 $|\hat{w}_{i-1,j}|$ 로 감소될 수도 있다. 다른 예들에서, $\alpha_j \neq 1$ 이다. 예를 들어, α_j 는 다음 방정식에 기초하여 결정될 수도 있다:

$$\alpha_j = \frac{\sum_{i=1}^{I} w_{i,j} w_{i-1,j}}{\sum_{i=1}^{I} w_{i-1,j}^{\ 2}}$$

[0135]

[0136] 여기서, I 는 α_j 를 결정하는데 사용되는 오디오 프레임들의 개수에 대응한다. 이전 방정식에서 나타낸 바와 같이, 가중 인자는, 일부 예들에서, 복수의 상이한 오디오 프레임들로부터의 복수의 상이한 가중 값들에 기초하여 결정될 수도 있다.

[0137] 또한, 예측된 벡터 양자화를 수행하도록 구성될 때, 양자화 유닛 (52) 은 다음 방정식에 기초하여 예측 가중 값을 발생시킬 수도 있다:

$$e_{i,j} = |w_{i,j}| - \alpha_j |\hat{w}_{i-1,j}|$$

[0138]

[0139] 여기서, $e_{i,j}$ 는 i번째 오디오 프레임에 대한 가중 값들의 순서정렬된 서브세트로부터 j번째 가중 값에 대한 예측 가중 값에 대응한다.

[0140] 양자화 유닛 (52) 은 예측 가중 값 및 예측된 벡터 양자화 (PVQ) 코드북에 기초하여, 양자화된 예측 가중 값을 발생시킨다. 예를 들어, 양자화 유닛 (52) 은 양자화된 예측 가중 값을 발생시키기 위해, 코딩될 벡터에 대해 또는 코딩될 프레임에 대해 발생된 다른 예측 가중 값들과 조합하여 예측 가중 값을 벡터 양자화할 수도 있다.

[0141] 양자화 유닛 (52) 은 PVQ 코드북에 기초하여 예측 가중 값 (620) 을 벡터 양자화할 수도 있다. PVQ 코드북은 복수의 M-구성요소 후보 양자화 벡터들을 포함할 수도 있으며, 양자화 유닛 (52) 은 Z 예측 가중 값들을 표

시하기 위해 후보 양자화 벡터들 중 하나를 선택할 수도 있다. 일부 예들에서, 양자화 유닛 (52) 은 양자화 에러를 최소화하는 (예컨대, 최소 자승 에러를 최소화하는) 후보 양자화 벡터를 PVQ 코드북으로부터 선택할 수도 있다.

[0142] 일부 예들에서, PVQ 코드북은 엔트리들의 각각이 양자화 코드북 인덱스 및 대응하는 M-구성요소 후보 양자화 벡터를 포함하는 복수의 엔트리들을 포함할 수도 있다. 양자화 코드북에서의 인덱스들의 각각은 복수의 M-구성요소 후보 양자화 벡터들의 개개의 하나에 대응할 수도 있다.

[0143] 양자화 벡터들의 각각에서의 구성요소들의 개수는 단일 v-벡터를 표시하도록 선택된 가중치들의 개수 (즉, Z) 에 의존할 수도 있다. 일반적으로, Z-구성요소 후보 양자화 벡터들을 가지는 코드북에 대해, 양자화 유닛 (52) 은 한번에 Z 예측 가중 값들을 벡터 양자화하여 단일 양자화된 벡터를 발생시킬 수도 있다. 양자화 코드북에서의 엔트리들의 개수는 가중 값들을 벡터 양자화하는데 사용되는 비트-레이트에 의존할 수도 있다.

[0144] 양자화 유닛 (52) 이 예측 가중 값을 벡터 양자화할 때, 양자화 유닛 (52) 은 PVQ 코드북으로부터 Z-구성요소 벡터를 Z 예측 가중 값들을 표시하는 양자화 벡터로 선택할 수도 있다. 양자화된 예측 가중 값은 i번째 오디오 프레임에 대한 j번째 예측 가중 값의 벡터-양자화된 버전에 추가로 대응할 수도 있는, i번째 오디오 프레임에 대한 Z-구성요소 양자화 벡터의 j번째 구성요소에 대응할 수도 있는, $\hat{e}_{i,j}$ 로서 표시될 수도 있다.

[0145] 예측된 벡터 양자화를 수행하도록 구성될 때, 양자화 유닛 (52) 은 또한 양자화된 예측 가중 값 및 가중된 복원된 가중 값에 기초하여, 복원된 가중 값을 발생시킬 수도 있다. 예를 들어, 양자화 유닛 (52) 은 가중된 복원된 가중 값을 양자화된 예측 가중 값에 가산하여, 복원된 가중 값을 발생시킬 수도 있다. 가중된 복원된 가중 값은 위에서 설명된 가중된 복원된 가중 값과 동일할 수도 있다. 일부 예들에서, 가중된 복원된 가중 값은 복원된 가중 값의 가중된 및 지연된 버전일 수도 있다.

[0146] 복원된 가중 값은 대응하는 복원된 가중 값, $\hat{w}_{i-1,j}$ 의 크기 (또는, 절대값) 에 대응하는 $\left|\hat{w}_{i-1,j}\right|$ 로서 표현될 수도 있다. 복원된 가중 값, $\hat{w}_{i-1,j}$ 은, (i-1)번째 오디오 프레임에 대한 복원된 가중 값들의 순서정렬된 서브세트로부터 j번째 복원된 가중 값에 대응한다. 일부 예들에서, 양자화 유닛 (52) 은 예측 코딩되는 가중 값의 부호를 표시하는 데이터를 별개로 코딩할 수도 있으며, 디코더는 이 정보를 이용하여, 복원된 가중 값의 부호를 결정할 수도 있다.

[0147] 양자화 유닛 (52) 은 다음 방정식에 기초하여, 복원된 가중 값을 발생시킬 수도 있다:

[0148] $$\left|\hat{w}_{i,j}\right| = \hat{e}_{i,j} + \alpha_j \left|\hat{w}_{i-1,j}\right|$$

[0149] 여기서, $\hat{e}_{i,j}$ 는 i번째 오디오 프레임에 대한 가중 값들의 순서정렬된 서브세트 (예컨대, M-구성요소 양자화 벡터의 j번째 구성요소) 로부터 j번째 가중 값에 대한 양자화된 예측 가중 값에 대응하며, $\left|\hat{w}_{i-1,j}\right|$ 는 (i-1)번째 오디오 프레임에 대한 가중 값들의 순서정렬된 서브세트로부터 j번째 가중 값에 대한 복원된 가중 값의 크기에 대응하며, α_j 는 가중 값들의 순서정렬된 서브세트로부터 j번째 가중 값에 대한 가중 인자에 대응한다.

[0150] 양자화 유닛 (52) 은 복원된 가중 값에 기초하여, 지연된 복원된 가중 값을 발생시킬 수도 있다. 예를 들어, 양자화 유닛 (52) 은 복원된 가중 값을 하나의 오디오 프레임 만큼 지연시켜, 지연된 복원된 가중 값을 발생시킬 수도 있다.

[0151] 양자화 유닛 (52) 은 또한 지연된 복원된 가중 값 및 가중 인자에 기초하여, 가중된 복원된 가중 값을 발생시킬 수도 있다. 예를 들어, 양자화 유닛 (52) 은 지연된 복원된 가중 값을 가중 인자으로 곱하여, 가중된 복원된 가중 값을 발생시킬 수도 있다.

[0152] 이와 유사하게, 양자화 유닛 (52) 은 지연된 복원된 가중 값 및 가중 인자에 기초하여, 가중된 복원된 가중 값을 발생시킨다. 예를 들어, 양자화 유닛 (52) 은 지연된 복원된 가중 값을 가중 인자로 곱하여, 가중된 복원된 가중 값을 발생시킬 수도 있다.

[0153] PVQ 코드북로부터 Z-구성요소 벡터를 Z 예측 가중 값들에 대한 양자화 벡터로 선택하는 것에 응답하여, 양자화 유닛 (52) 은, 일부 예들에서, 선택된 Z-구성요소 벡터 자체를 코딩하는 대신, 그 선택된 Z-구성요소 벡터에 대응하는 (PVQ 코드북으로부터의) 인덱스를 코딩할 수도 있다. 인덱스는 양자화된 예측 가중 값들의 세트를 표시할 수도 있다. 이러한 예들에서, 디코더 (24) 는 PVQ 코드북과 유사한 코드북을 포함할 수도 있으며, 디코더 코드북에서의 대응하는 Z-구성요소 벡터로의 인덱스를 맵핑함으로써, 양자화된 예측 가중 값들을 표시하는 인덱스를 디코딩할 수도 있다. Z-구성요소 벡터에서의 구성요소들의 각각은 양자화된 예측 가중 값에 대응할 수도 있다.

[0154] 벡터 (예컨대, V-벡터) 를 스칼라 양자화하는 것은 벡터의 구성요소들의 각각을 다른 구성요소들과는 개별적으로 및/또는 독립적으로 양자화하는 것을 포함할 수도 있다. 예를 들어, 다음 예시적인 V-벡터를 고려한다:

[0155] $$V = [0.23 \quad 0.31 \quad -0.47 \quad \cdots \quad 0.85]$$

[0156] 이 예시적인 V-벡터를 스칼라 양자화하기 위해, 구성요소들의 각각은 개별적으로 양자화될 (즉, 스칼라-양자화될) 수도 있다. 예를 들어, 양자화 단계가 0.1 이면, 0.23 구성요소는 0.2 로 양자화될 수도 있으며, 0.31 구성요소는 0.3 으로 양자화될 수도 있으며, 기타등등으로 양자화될 수도 있다. 스칼라-양자화된 구성요소들은 일괄하여 스칼라-양자화된 V-벡터를 형성할 수도 있다.

[0157] 다시 말해서, 양자화 유닛 (52) 은 감소된 포그라운드 V[k] 벡터들 (55) 의 주어진 하나의 엘리먼트들의 모두에 대해 균일한 스칼라 양자화를 수행할 수도 있다. 양자화 유닛 (52) 은 NbitsQ 신택스 엘리먼트로서 표시될 수도 있는 값에 기초하여, 양자화 단계 사이즈를 식별할 수도 있다. 양자화 유닛 (52) 은 목표 비트레이트 (41) 에 기초하여, 이 NbitsQ 신택스 엘리먼트를 동적으로 결정할 수도 있다. NbitsQ 신택스 엘리먼트는 또한 아래에 재현된 ChannelSideInfoData 신택스 테이블에서 언급된 바와 같이 양자화 모드를 식별할 수도 있지만, 또한 단계 사이즈를 스칼라 양자화하기 위한 목적들을 위해 식별할 수도 있다. 즉, 양자화 유닛 (52) 은 양자화 단계 사이즈를 이 NbitsQ 신택스 엘리먼트의 함수로서 결정할 수도 있다. 일 예로서, 양자화 유닛 (52) 은 (본 개시물에서 "delta" 또는 "Δ" 로서 표시된) 양자화 단계 사이즈를 $2^{16-NbitsQ}$ 와 동일하게 결정할 수도 있다. 이 예에서, NbitsQ 신택스 엘리먼트의 값이 6 과 동일할 때, delta 는 2^{10} 와 동일하며, 2^6 개의 양자화 레벨들이 존재한다. 이 점에서, 벡터 엘리먼트 v 에 대해, 양자화된 벡터 엘리먼트 v_q 는 [v/Δ] 및 $-2^{NbitsQ-1} < v_q < 2^{NbitsQ-1}$ 와 동일하다.

[0158] 양자화 유닛 (52) 은 그후 양자화된 벡터 엘리먼트들의 범주화 및 잔차 코딩을 수행할 수도 있다. 일 예로서, 양자화 유닛 (52) 은, 주어진 양자화된 벡터 엘리먼트 v_q 에 대해, 이 엘리먼트가 대응하는 카테고리를 (카테고리 식별자 cid 를 결정함으로써) 다음 방정식을 이용하여 식별할 수도 있다:

[0159] $$cid = \begin{cases} 0, & v_q = 0 \text{ 이면} \\ \lfloor log_2|v_q| \rfloor + 1, & v_q \neq 0 \text{ 이면} \end{cases}$$

[0160] 양자화 유닛 (52) 은 그후 이 카테고리 인덱스 cid 를 Huffman 코딩할 수도 있으며, 한편, v_q 가 양의 값 또는 음의 값인지 여부를 표시하는 부호 비트를 또한 식별할 수도 있다. 양자화 유닛 (52) 은 다음으로, 이 카테고리에서 잔차를 식별할 수도 있다. 일 예로서, 양자화 유닛 (52) 은 다음 방정식에 따라서 이 잔차를 결정할 수도 있다:

[0161] $$\text{잔차} = |v_q| - 2^{cid-1}$$

[0162] 양자화 유닛 (52) 은 그후 cid-1 비트들로 이 잔차를 블록 코딩할 수도 있다.

[0163] 양자화 유닛 (52) 은, 일부 예들에서, cid 를 코딩할 때 NbitsQ 신택스 엘리먼트의 상이한 값들에 대해 상이한 Huffman 코드 북들을 선택할 수도 있다. 일부 예들에서, 양자화 유닛 (52) 은 NbitsQ 신택스 엘리먼트 값들

6, …, 15 에 대해 상이한 Huffman 코딩 테이블을 제공할 수도 있다. 더욱이, 양자화 유닛 (52) 은 총 50 개의 Huffman 코드 북들에 대해 6, …, 15 의 범위인 상이한 NbitsQ 신택스 엘리먼트 값들의 각각에 대해, 5개의 상이한 Huffman 코드 북들을 포함할 수도 있다. 이 점에서, 양자화 유닛 (52) 은 다수의 상이한 통계적 상황들에서 cid 의 코딩을 수용하기 위해 복수의 상이한 Huffman 코드 북들을 포함할 수도 있다.

[0164] 예시하기 위하여, 양자화 유닛 (52) 은, NbitsQ 신택스 엘리먼트 값들의 각각에 대해, 벡터 엘리먼트들 1 내지 4 를 코딩하기 위한 제 1 Huffman 코드 북, 벡터 엘리먼트들 5 내지 9 를 코딩하기 위한 제 2 Huffman 코드 북, 벡터 엘리먼트들 9 및 이상을 코딩하기 위한 제 3 Huffman 코드 북을 포함할 수도 있다. 이들 처음 3개의 Huffman 코드 북들은 압축될 감소된 포그라운드 V[k] 벡터들 (55) 중 하나가 감소된 포그라운드 V[k] 벡터들 (55) 중 시간적으로 후속 대응하는 하나로부터 예측되지 않고 합성 오디오 오브젝트의 공간 정보 (예를 들어, 펄스부호 변조된 (PCM) 오디오 오브젝트에 의해 원래 정의된 것) 를 표시하지 않을 때 사용될 수도 있다. 양자화 유닛 (52) 은 추가적으로, NbitsQ 신택스 엘리먼트 값들의 각각에 대해, 이 감소된 포그라운드 V[k] 벡터들 (55) 중 하나가 감소된 포그라운드 V[k] 벡터들 (55) 중 시간적으로 후속 대응하는 하나로부터 예측될 때, 감소된 포그라운드 V[k] 벡터들 (55) 중 하나를 코딩하기 위한 제 4 Huffman 코드 북을 포함할 수도 있다. 양자화 유닛 (52) 은 또한, NbitsQ 신택스 엘리먼트 값들의 각각에 대해, 이 감소된 포그라운드 V[k] 벡터들 (55) 중 하나가 합성 오디오 오브젝트를 표시할 때, 감소된 포그라운드 V[k] 벡터들 (55) 중 하나를 코딩하기 위한 제 5 Huffman 코드 북을 포함할 수도 있다. 여러 Huffman 코드 북들이 이 예에서, 이들 상이한 통계적 상황들, 즉, 비-예측 및 비-합성 상황, 예측된 상황 및 합성 상황의 각각에 대해 전개될 수도 있다.

[0165] 다음 테이블은 압축해제 유닛이 적합한 Huffman 테이블을 선택하도록 하기 위해서 비트스트림에 규정될 Huffman 테이블 선택 및 비트들을 예시한다:

예측 (Pred) 모드	HT 정보	HT 테이블
0	0	HT5
0	1	HT{1,2,3}
1	0	HT4
1	1	HT5

[0166]

[0167] 상기 테이블에서, 예측 모드 ("Pred 모드") 는 현재의 벡터에 대해 예측이 수행되었는지 여부를 표시하며, 한편 Huffman 테이블 ("HT 정보") 은 Huffman 테이블들 1 내지 5 중 하나를 선택하는 사용되는 추가적인 Huffman 코드 북 (또는, 테이블) 정보를 나타낸다. 예측 모드는 또한 아래에서 설명되는 PFlag 신택스 엘리먼트로서 표현될 수도 있으며, 한편 HT 정보는 아래에서 설명되는 CbFlag 신택스 엘리먼트에 의해 표현될 수도 있다.

[0168] 다음 테이블은 이 Huffman 테이블 선택 프로세스 주어진 여러 통계적 컨텍스트들 또는 시나리오들을 추가로 예시한다.

	리코딩	합성
W/O 예측 (Pred)	HT{1,2,3}	HT5
예측에 의해 (With Pred)	HT4	HT5

[0169]

[0170] 상기 테이블에서, "리코딩" 칼럼은 벡터가 리코딩된 오디오 오브젝트를 표시할 때의 코딩 상황을 표시하며, 한편, "합성 (Synthetic)" 칼럼은 벡터가 합성 오디오 오브젝트를 표시할 때에 대한 코딩 상황을 표시한다. "W/O 예측 (Pred)" 로우는 벡터 엘리먼트들에 대해 예측이 수행되지 않을 때의 코딩 상황을 표시하며, 한편, "예측에 의해 (With Pred)" 로우는 벡터 엘리먼트들에 대해 예측이 수행될 때의 코딩 상황을 표시한다. 이 테이블에 나타낸 바와 같이, 양자화 유닛 (52) 은 벡터가 리코딩된 오디오 오브젝트를 표시하고 벡터 엘리먼트들에 대해 예측이 수행되지 않을 때 HT{1, 2, 3} 를 선택한다. 양자화 유닛 (52) 은 오디오 오브젝트가 합성 오디오 오브젝트를 표시하며 벡터 엘리먼트들에 대해 예측이 수행되지 않을 때 HT5 를 선택한다. 양자화 유닛 (52) 은 벡터가 리코딩된 오디오 오브젝트를 표시하며 예측이 벡터 엘리먼트들에 대해 예측이 수행될 때 HT4 를 선택한다. 양자화 유닛 (52) 은 오디오 오브젝트가 합성 오디오 오브젝트를 표시하며 벡터 엘리먼트들에 대해 예측이 수행될 때 HT5 를 선택한다.

[0171] 양자화 유닛 (52) 은 출력 스위칭된-양자화된 V-벡터로서 사용할, 비-예측된 벡터-양자화된 V-벡터, 예측된 벡터-양자화된 V-벡터, 비-Huffman-코딩된 스칼라-양자화된 V-벡터, 및 Huffman-코딩된 스칼라-양자화된 V-벡터 중 하나를, 본 개시물에서 설명되는 기준들의 임의의 조합에 기초하여 선택할 수도 있다. 일부 예들에서, 양자화 유닛 (52) 은 벡터 양자화 모드 및 하나 이상의 스칼라 양자화 모드들을 포함하는 양자화 모드들의 세트로부터 양자화 모드를 선택하고, 그 선택된 모드에 기초하여 (또는, 그에 따라서) 입력 V-벡터를 양자화할 수도 있다. 양자화 유닛 (52) 은 그후 (예컨대, 가중 값들 또는 그의 표시하는 비트들의 관점에서) 비-예측된 벡터-양자화된 V-벡터, (예컨대, 에러 값들 또는 그의 표시하는 비트들의 관점에서) 예측된 벡터-양자화된 V-벡터, 비-Huffman-코딩된 스칼라-양자화된 V-벡터 및 Huffman-코딩된 스칼라-양자화된 V-벡터 중 선택된 하나를, 비트스트림 발생 유닛 (52) 에, 코딩된 포그라운드 V[k] 벡터들 (57) 로서 제공할 수도 있다. 양자화 유닛 (52) 은 또한 양자화 모드를 표시하는 신택스 엘리먼트들 (예컨대, NbitsQ 신택스 엘리먼트) 및 도 4 및 도 7 의 예에 대해 아래에서 좀더 자세히 설명되는 바와 같이 V-벡터를 역양자화하거나 또는 아니면 복원하는데 사용되는 임의의 다른 신택스 엘리먼트들을 제공할 수도 있다.

[0172] 오디오 인코딩 디바이스 (20) 내에 포함되는 음향심리 오디오 코더 유닛 (40) 은 음향심리 오디오 코더의 다수의 인스턴스들을 나타낼 수도 있으며, 이의 각각은 에너지 보상된 주변 HOA 계수들 (47') 및 내삽된 nFG 신호들 (49') 의 각각의 상이한 오디오 오브젝트 또는 HOA 채널을 인코딩하여 인코딩된 주변 HOA 계수들 (59) 및 인코딩된 nFG 신호들 (61) 을 발생시키는데 사용된다. 음향심리 오디오 코더 유닛 (40) 은 인코딩된 주변 HOA 계수들 (59) 및 인코딩된 nFG 신호들 (61) 을 비트스트림 발생 유닛 (42) 으로 출력할 수도 있다.

[0173] 오디오 인코딩 디바이스 (20) 내에 포함된 비트스트림 발생 유닛 (42) 은 (디코딩 디바이스에 의해 알려진 포맷을 지칭할 수도 있는) 기지의 포맷을 따르도록 데이터를 포맷하여, 벡터-기반 비트스트림 (21) 을 발생시키는 유닛을 나타낸다. 즉, 비트스트림 (21) 은 위에서 설명된 방법으로 인코딩되어 있는 인코딩된 오디오 데이터를 나타낼 수도 있다. 비트스트림 발생 유닛 (42) 은 일부 예들에서, 코딩된 포그라운드 V[k] 벡터들 (57), 인코딩된 주변 HOA 계수들 (59), 인코딩된 nFG 신호들 (61) 및 백그라운드 채널 정보 (43) 를 수신할 수도 있는 멀티플렉서를 나타낼 수도 있다. 비트스트림 발생 유닛 (42) 은 그후 코딩된 포그라운드 V[k] 벡터들 (57), 인코딩된 주변 HOA 계수들 (59), 인코딩된 nFG 신호들 (61) 및 백그라운드 채널 정보 (43) 에 기초하여, 비트스트림 (21) 을 발생시킬 수도 있다. 이러한 방법으로, 비트스트림 발생 유닛 (42) 은 이에 의해 도 7 의 예에 대해 아래에서 좀더 자세히 설명되는 바와 같이 비트스트림 (21) 을 획득하기 위해 벡터들 (57) 을 비트스트림 (21) 에 규정할 수도 있다. 비트스트림 (21) 은 1차 또는 메인 비트스트림 및 하나 이상의 부 채널 비트스트림들을 포함할 수도 있다.

[0174] 도 3 의 예에서는 나타내지 않았지만, 오디오 인코딩 디바이스 (20) 는 또한 현재의 프레임이 방향-기반 합성 또는 벡터-기반 합성을 이용하여 인코딩되는지 여부에 기초하여 오디오 인코딩 디바이스 (20) 로부터 출력된 비트스트림 출력을 (예컨대, 방향-기반 비트스트림 (21) 과 벡터-기반 비트스트림 (21) 사이에) 스위칭하는 비트스트림 출력 유닛을 포함할 수도 있다. 비트스트림 출력 유닛은 방향-기반 합성이 (HOA 계수들 (11) 이 합성 오디오 오브젝트로부터 발생되었다고 검출한 결과로서) 수행되었는지 여부 또는 벡터-기반 합성이 (HOA 계수들이 기록되었다고 검출한 결과로서) 수행되었는지 여부를 나타내는 콘텐츠 분석 유닛 (26) 에 의해 출력된 신택스 엘리먼트에 기초하여 스위칭을 수행할 수도 있다. 비트스트림 출력 유닛은 비트스트림들 (21) 의 개개의 하나와 함께 현재의 프레임에 대해 수행되는 스위치 또는 현재의 인코딩을 나타내는 올바른 헤더 신택스를 규정할 수도 있다.

[0175] 더욱이, 위에서 언급한 바와 같이, 음장 분석 유닛 (44) 은 (때로는 BG_{TOT} 가 2개 이상의 (시간에서) 인접한 프레임들에 걸쳐서 일정하거나 또는 동일하게 유지할 수도 있지만) 프레임 단위로 변할 수도 있는 BG_{TOT} 주변 HOA 계수들 (47) 을 식별할 수도 있다. BG_{TOT} 에서의 변화는 감소된 포그라운드 V[k] 벡터들 (55) 로 표현된 계수들에 대해 변화들을 초래할 수도 있다. BG_{TOT} 에서의 변화는 (또한, 때로는 BG_{TOT} 가 2개 이상의 (시간에서) 인접한 프레임들에 걸쳐서 일정하거나 또는 동일하게 유지할 수도 있지만) 프레임 단위로 변하는 ("주변 HOA 계수들" 로서 또한 지칭될 수도 있는) 백그라운드 HOA 계수들을 초래할 수도 있다. 이 변화들은 종종 추가적인 주변 HOA 계수들의 추가 또는 제거, 및 감소된 포그라운드 V[k] 벡터들 (55) 로부터의 계수들의 대응하는 제거 또는 감소된 포그라운드 V[k] 벡터들 (55) 에의 계수들의 추가로 표현되는 음장의 양태들에 대해 에너지의 변화를 초래한다.

[0176] 그 결과, 음장 분석 유닛 (44) 은 주변 HOA 계수들이 프레임들 간에 변하는 시점을 추가로 결정하고, (변화가 주변 HOA 계수의 "전이" 로서 또는 주변 HOA 계수의 "전이" 로서 또한 지칭될 수도 있는) 음장의 주변 구성요소들을 나타내는데 사용되는 관점에서 주변 HOA 계수에 대한 변화를 나타내는 플래그 또는 다른 신택스 엘리먼트를 발생시킬 수도 있다. 특히, 계수 감소 유닛 (46) 은 (AmbCoeffTransition 플래그 또는 AmbCoeffIdxTransition 플래그로서 표시될 수도 있는) 플래그를 발생시켜, 그 플래그가 (가능한 한 부 채널 정보의 일부로서) 비트스트림 (21) 에 포함될 수 있도록 그 플래그를 비트스트림 발생 유닛 (42) 에 제공할 수도 있다.

[0177] 계수 감소 유닛 (46) 은 주변 계수 전이 플래그를 규정하는 것에 더하여, 또한 감소된 포그라운드 V[k] 벡터들 (55) 이 발생되는 방법을 수정할 수도 있다. 일 예에서, 주변 HOA 주변 계수들 중 하나가 현재의 프레임 동안 전이 중이라고 결정하자 마자, 계수 감소 유닛 (46) 은 전이 중인 주변 HOA 계수에 대응하는 감소된 포그라운드 V[k] 벡터들 (55) 의 V-벡터들의 각각에 대해 ("벡터 엘리먼트" 또는 "엘리먼트" 로서 또한 지칭될 수도 있는) 벡터 계수를 규정할 수도 있다. 또, 전이 중인 주변 HOA 계수는 백그라운드 계수들의 총 개수 BG_{TOT} 에 추가하거나 또는 그로부터 제거될 수도 있다. 따라서, 백그라운드 계수들의 총 개수에서의 최종 변화는 주변 HOA 계수가 비트스트림에 포함되는지 여부, 및 V-벡터들의 대응하는 엘리먼트가 위에서 설명된 제 2 및 제 3 구성 모드들에서 비트스트림에 규정된 V-벡터들을 위해 포함되는지 여부에 영향을 미친다. 계수 감소 유닛 (46) 이 에너지에서의 변화들을 극복하기 위해 감소된 포그라운드 V[k] 벡터들 (55) 을 규정할 수 있는 방법에 관한 더 많은 정보는 "TRANSITIONING OF AMBIENT HIGHER_ORDER AMBISONIC COEFFICIENTS" 란 발명의 명칭으로, 2015년 1월 12일에 출원된, 미국 출원 번호 제 14/594,533호에서 제공된다.

[0178] 도 4 는 도 2 의 오디오 디코딩 디바이스 (24) 를 좀더 자세하게 예시하는 블록도이다. 도 4 의 예에 나타낸 바와 같이, 오디오 디코딩 디바이스 (24) 는 추출 유닛 (72), 방향성-기반 복원 유닛 (90) 및 벡터-기반 복원 유닛 (92) 을 포함할 수도 있다. 아래에서 설명되지만, 오디오 디코딩 디바이스 (24) 및 HOA 계수들을 분해하거나 또는 아니면 디코딩하는 여러 양태들에 관한 더 많은 정보는 "INTERPOLATION FOR DECOMPOSED REPRESENTATIONS OF A SOUND FIELD"란 발명의 명칭으로, 2014년 5월 29일에 출원된 국제 특허 출원 공개 번호 제 WO 2014/194099호에서 입수가능하다.

[0179] 추출 유닛 (72) 은 비트스트림 (21) 을 수신하여 HOA 계수들 (11) 의 여러 인코딩된 버전들 (예컨대, 방향-기반 인코딩된 버전 또는 벡터-기반의 인코딩된 버전) 을 추출하도록 구성된 유닛을 나타낼 수도 있다. 추출 유닛 (72) 은 HOA 계수들 (11) 이 여러 방향-기반의 또는 벡터-기반의 버전들을 통해서 인코딩되었는지 여부를 표시하는 위에서 언급된 신택스 엘리먼트로부터 결정할 수도 있다. 방향-기반 인코딩이 수행되었을 때, 추출 유닛 (72) 은 HOA 계수들 (11) 의 방향-기반 버전 및 (도 4 의 예에서 방향-기반 정보 (91) 로서 표시된) 인코

딩된 버전과 연관된 신택스 엘리먼트들을 추출하여, 방향 기반의 정보 (91) 를 방향-기반 복원 유닛 (90) 으로 전달할 수도 있다. 방향-기반 복원 유닛 (90) 은 방향-기반 정보 (91) 에 기초하여 HOA 계수들을 HOA 계수들 (11') 의 유형으로 복원하도록 구성된 유닛을 나타낼 수도 있다. 비트스트림 및 비트스트림 내 신택스 엘리먼트들의 배열이 도 7a 내지 도 7j 의 예에 대해 아래에서 좀더 자세히 설명된다.

[0180] HOA 계수들 (11) 이 벡터-기반 합성을 이용하여 인코딩되었다고 신택스 엘리먼트가 표시할 때, 추출 유닛 (72) 은 (코딩된 가중치들 (57) 및/또는 인덱스들 (63) 또는 스칼라 양자화된 V-벡터들을 포함할 수도 있는) 코딩된 포그라운드 V[k] 벡터들 (57), 인코딩된 주변 HOA 계수들 (59) 및 (인코딩된 nFG 신호들 (61) 로서 또한 지칭될 수도 있는) 대응하는 오디오 오브젝트들 (61) 을 추출할 수도 있다. 오디오 오브젝트들 (61) 은 각각 벡터들 (57) 중 하나에 대응한다. 추출 유닛 (72) 은 코딩된 포그라운드 V[k] 벡터들 (57) 을 V-벡터 복원 유닛 (74) 으로, 그리고 인코딩된 주변 HOA 계수들 (59) 을 인코딩된 nFG 신호들 (61) 과 함께 음향심리 디코딩 유닛 (80) 으로 전달할 수도 있다.

[0181] 코딩된 포그라운드 V[k] 벡터들 (57) 을 추출하기 위해, 추출 유닛 (72) 은 다음 ChannelSideInfoData (CSID) 신택스 테이블에 따라서 신택스 엘리먼트들을 추출할 수도 있다.

[0182] 테이블 - ChannelSideInfoData(i) 의 신택스

신택스	비트들의 수	Mnemonic
ChannelSideInfoData(i)		
{		
ChannelType[i]	**2**	**uimsbf**
switch ChannelType[i]		
{		
case 0:		
ActiveDirsIds[i];	NumOfBits PerDirIdx	**uimsbf**
break;		
case 1:		
bA;	**1**	**bslbf**
bB;	**1**	**bslbf**
if ((bA + bB) == 0) {		
NbitsQ(k)[i] = NbitsQ(k-1)[i];		
PFlag(k)[i] = PFlag(k-1)[i];		
CbFlag(k)[i] = CbFlag(k-1)[i];		
CodebkIdx(k)[i] = CodebkIdx(k-1)[i];		
NumVecIndices(k)[i] = NumVecIndices[k-1][i];		
}		
else{		
NbitsQ(k)[i] = (8*bA)+(4*bB)+**uintC**;	**2**	**uimsbf**
if (NbitsQ(k)[i] == 4) {		
PFlag(k)[i];	**1**	**bslbf**
CodebkIdx(k)[i];	**3**	**uimsbf**
NumVecIndices(k)[i]++;	**NumVVec VqElemen tsBits**	**uimsbf**
}		
elseif (NbitsQ(k)[i] >= 6) {		
PFlag(k)[i];	**1**	**bslbf**
CbFlag(k)[i];	**1**	**bslbf**
}		
}		
}		
break;		
case 2:		
AddAmbHoaInfoChannel(i);		
break;		
default:		
}		
}		

[0183]

[0184] 상기 테이블에 대한 의미들은 다음과 같다.

[0185] 이 페이로드는 i-번째 채널에 대한 부수 정보를 유지한다. 페이로드의 사이즈 및 데이터는 채널의 유형에 의존한다.

[0186] ChannelType[i] 이 엘리먼트는 테이블 (95) 에 정의되는 i-번째 채널의 유형을 저장한다.

[0187] ActiveDirsIds[i] 이 엘리먼트는 부속서 F.7 로부터의 900 개의 미리 정의된, 균일하게 분포된 지점들의 인덱스를 이용하여 활성 방향 신호의 방향을 나타낸다. 코드 워드 0 은 방향 신호의 끝을 시그널링하는데 사용된다.

[0188] PFlag[i] i-번째 채널의 벡터-기반의 신호와 연관된 [[스칼라-양자화된 V-벡터의 Huffman 디코딩에 사용되는]] (이중 브라켓 [[ ]] 은 그 안의 내용이 삭제된 것임을 나타낸다) 예측 플래그.

[0189] CbFlag[i] i-번째 채널의 벡터-기반의 신호와 연관된 스칼라-양자화된 V-벡터의 Huffman 디코딩

에 사용되는 코드북 플래그.

[0190] CodebkIdx[i] i-번째 채널의 벡터-기반의 신호와 연관된 벡터-양자화된 V-벡터를 역양자화하는데 사용되는 특정의 코드북을 시그널링한다.

[0191] NbitsQ[i] 이 인덱스는 i-번째 채널의 벡터-기반의 신호와 연관된 데이터의 Huffman 디코딩에 사용되는 Huffman 테이블을 결정한다. 코드 워드 5 는 균일한 8비트 역양자화기의 사용을 결정한다. 2개의 MSB들 00 은 이전 프레임 (k-1) 의 NbitsQ[i], PFlag[i] 및 CbFlag[i] 데이터를 재사용하는 것을 결정한다.

[0192] bA, bB NbitsQ[i] 필드의 msb (bA) 및 제 2 msb (bB).

[0193] uintC NbitsQ[i] 필드의 나머지 2 비트들의 코드 워드.

[0194] NumVecIndices 벡터-양자화된 V 벡터를 역양자화하는데 사용되는 벡터들의 개수.

[0195] AddAmbHoaInfoChannel(i) 이 페이로드는 추가적인 주변 HOA 계수들에 대한 정보를 유지한다.

[0196] CSID 신택스 테이블에 따라서, 추출 유닛 (72) 은 채널의 유형을 표시하는 ChannelType 신택스 엘리먼트를 먼저 획득할 수도 있다 (예컨대, 여기서, 제로의 값은 방향-기반의 신호를 시그널링하고, 1 의 값은 벡터-기반의 신호를 시그널링하고, 2 의 값은 추가적인 주변 HOA 신호를 시그널링한다). ChannelType 신택스 엘리먼트에 기초하여, 추출 유닛 (72) 은 3개의 경우들 사이에 스위칭할 수도 있다.

[0197] 본 개시물에서 설명하는 기법들의 일 예를 예시하기 위해 케이스 1 에 집중하면, 추출 유닛 (72) 은 NbitsQ 신택스 엘리먼트 (즉, 상기 예시적인 CSID 신택스 테이블에서의 bA 신택스 엘리먼트) 의 최상위 비트 및 NbitsQ 신택스 엘리먼트 (즉, 상기 예시적인 CSID 신택스 테이블에서의 bB 신택스 엘리먼트) 의 제 2 최상위 비트를 획득할 수도 있다. NbitsQ(k)[i] 의 (k)[i] 는 NbitsQ 신택스 엘리먼트가 i 번째 전송 채널의 k 번째 프레임에 대해 획득된다는 것을 표시할 수도 있다. NbitsQ 신택스 엘리먼트는 HOA 계수들 (11) 에 의해 표시되는 음장의 공간 구성요소를 양자화하는데 사용되는 양자화 모드를 표시하는 하나 이상의 비트들을 표시할 수도 있다. 공간 구성요소는 또한 본 개시물에서 V-벡터로서 또는 코딩된 포그라운드 V[k] 벡터들 (57) 로서 지칭될 수도 있다.

[0198] 상기 예시적인 CSID 신택스 테이블에서, NbitsQ 신택스 엘리먼트는 (NbitsQ 신택스 엘리먼트에 대해 제로 내지 3 의 값이 예약되거나 또는 미사용됨에 따라), 대응하는 VVecData 필드에 규정된 벡터를 압축하는데 사용되는 12 개의 양자화 모드들 중 하나를 표시하기 위해 4 비트를 포함할 수도 있다. 12 개의 양자화 모드들은 아래에 나타낸 다음을 포함한다:

[0199] 0-3: 예약됨

[0200] 4: 벡터 양자화

[0201] 5: Huffman 코딩에 의하지 않은 스칼라 양자화

[0202] 6: Huffman 코딩에 의한 6-비트 스칼라 양자화

[0203] 7: Huffman 코딩에 의한 7-비트 스칼라 양자화

[0204] 8: Huffman 코딩에 의한 8-비트 스칼라 양자화

[0205] … …

[0206] 16: Huffman 코딩에 의한 16-비트 스칼라 양자화

[0207] 위에서, 6-16 의 NbitsQ 신택스 엘리먼트의 값은, 스칼라 양자화뿐만 아니라, 스칼라 양자화의 양자화 스텝 사이즈가 Huffman 코딩으로 수행된다는 것을 표시한다. 이 점에서, 양자화 모드는 벡터 양자화 모드, Huffman 코딩에 의하지 않은 스칼라 양자화 모드 및 Huffman 코딩에 의한 스칼라 양자화 모드를 포함할 수도 있다.

[0208] 상기 예시적인 CSID 신택스 테이블을 다시 참조하면, 추출 유닛 (72) 은 bA 신택스 엘리먼트를 bB 신택스 엘리먼트와 결합할 수도 있으며, 여기서, 이 조합은 상기 예시적인 CSID 신택스 테이블에 나타낸 바와 같이 가산 (추가) 일 수도 있다. 결합된 bA/bB 신택스 엘리먼트는 벡터를 압축하는데 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자를 나타낼 수도 있다. 추출 유닛 (72) 은 다음으로 그 결합된 bA/bB 신택스 엘리먼트를 제로의 값과 비교한다. 결합된 bA/bB 신택스 엘리먼트가 제로의 값을 가질 때, 추출 유닛 (72) 은 i 번째 전송 채널의 현재의 k 번째 프레임에 대한 양자화

모드 정보 (즉, 상기 예시적인 CSID 신택스 테이블에서의 양자화 모드를 표시하는 NbitsQ 신택스 엘리먼트) 가 i 번째 전송 채널의 k-1 번째 프레임의 양자화 모드 정보와 동일하다고 결정할 수도 있다. 다시 말해서, 표시자는, 제로 값으로 설정될 때, 이전 프레임으로부터의 적어도 하나의 신택스 엘리먼트를 재사용하는 것을 표시한다.

[0209] 추출 유닛 (72) 은 유사하게, i 번째 전송 채널의 현재의 k 번째 프레임에 대한 예측 정보 (즉, 이 예에서 예측이 벡터 양자화 또는 스칼라 양자화 동안 수행되는지 여부를 표시하는 PFlag 신택스 엘리먼트) 가 i 번째 전송 채널의 k-1 번째 프레임의 예측 정보와 동일하다고 결정한다. 추출 유닛 (72) 은 또한 i 번째 전송 채널의 현재의 k 번째 프레임에 대한 Huffman 코드북 정보 (즉, V-벡터를 복원하는데 사용되는 Huffman 코드북을 표시하는 CbFlag 신택스 엘리먼트) 가 i 번째 전송 채널의 k-1 번째 프레임의 Huffman 코드북 정보와 동일하다고 결정할 수도 있다. 추출 유닛 (72) 은 또한 i 번째 전송 채널의 현재의 k 번째 프레임에 대한 벡터 양자화 정보 (즉, V-벡터를 복원하는데 사용되는 벡터 양자화 코드북을 표시하는 CodebkIdx 신택스 엘리먼트 및 V-벡터를 복원하는데 사용되는 코드 벡터들의 개수를 표시하는 NumVecIndices 신택스 엘리먼트) 가 i 번째 전송 채널의 k-1 번째 프레임의 벡터 양자화 정보와 동일하다고 결정할 수도 있다.

[0210] 결합된 bA/bB 신택스 엘리먼트가 제로의 값을 가지지 않을 때, 추출 유닛 (72) 은 i 번째 전송 채널의 k 번째 프레임에 대한 양자화 모드 정보, 예측 정보, Huffman 코드북 정보 및 벡터 양자화 정보가 i 번째 전송 채널의 k-1 번째 프레임의 것들과 동일하지 않다고 결정할 수도 있다. 그 결과, 추출 유닛 (72) 은 NbitsQ 신택스 엘리먼트 (즉, 상기 예시적인 CSID 신택스 테이블에서의 uintC 신택스 엘리먼트) 의 최하위 비트들을 획득하고, bA, bB 및 uintC 신택스 엘리먼트를 결합하여 NbitsQ 신택스 엘리먼트를 획득할 수도 있다. 이 NbitsQ 신택스 엘리먼트에 기초하여, 추출 유닛 (72) 은, NbitsQ 신택스 엘리먼트가 벡터 양자화, PFlag, CodebkIdx, 및 NumVecIndices 신택스 엘리먼트들을 시그널링하는 시점, 또는, NbitsQ 신택스 엘리먼트가 Huffman 코딩에 의한 스칼라 양자화, PFlag 및 CbFlag 신택스 엘리먼트들을 시그널링하는 시점을 획득할 수도 있다. 이러한 방법으로, 추출 유닛 (72) 은 V-벡터를 복원하는데 사용되는 전술한 신택스 엘리먼트들을 추출하여, 이들 신택스 엘리먼트들을 벡터-기반 복원 유닛 (72) 으로 파싱할 수도 있다.

[0211] 추출 유닛 (72) 은 다음으로, i 번째 전송 채널의 k 번째 프레임으로부터 V-벡터를 추출할 수도 있다. 추출 유닛 (72) 은 CodedVVecLength 로 표시되는 신택스 엘리먼트를 포함하는 HOADecoderConfig 컨테이너를 획득할 수도 있다. 추출 유닛 (72) 은 HOADecoderConfig 컨테이너로부터 CodedVVecLength 를 파싱할 수도 있다. 추출 유닛 (72) 은 다음 VVecData 신택스 테이블에 따라서 V-벡터를 획득할 수도 있다.

[0212]

신택스	비트들의 수	Mnemonic
VVectorData(i)		
{		
if (NbitsQ(k)[i] == 4){		
if (NumVecIndices(k)[i] == 1) {		
VecIdx[0] = **VecIdx** + 1;	**10**	**uimsbf**
WeightVal[0] = ((**SgnVal***2)-1);	**1**	**uimsbf**
} else {		
WeightIdx;	**nbitsW**	**uimsbf**
nbitsIdx = ceil(log2(NumOfHoaCoeffs));		
for (j=0; j< NumVecIndices(k)[i]; ++j) {		
VecIdx[j] = **VecIdx** + 1;	**nbitsIdx**	**uimsbf**
if (PFlag[i] == 0) {		
tmpWeightVal(k) [j] = WeightValCdbk[CodebkIdx(k)[i]][WeightIdx][j];		
else {		

tmpWeightVal(k) [j] = WeightValPredCdbk[CodebkIdx(k)[i]][WeightIdx][j] + WeightValAlpha[j] * tmpWeightVal(k-1) [j];		
}		
WeightVal[j] = ((**SgnVal***2)-1) * tmpWeightVal(k) [j];	**1**	**uimsbf**
}		
}		
}		
else if (NbitsQ(k)[i] == 5) {		
for (m=0; m< VVecLength; ++m)		
aVal[i][m] = (**VecVal** / 128.0) – 1.0;	**8**	**uimsbf**
}		
else if(NbitsQ(k)[i] >= 6) {		
for (m=0; m< VVecLength; ++m){		
huffIdx = *huffSelect*(VVecCoeffId[m], PFlag[i], CbFlag[i]);		
cid = *huffDecode*(NbitsQ[i], huffIdx, **huffVal**);	**dynamic**	**huffDecode**
aVal[i][m] = 0.0;		
if (cid > 0) {		
aVal[i][m] = sgn = (**SgnVal** * 2) - 1;	**1**	**bslbf**
if (cid > 1) {		
aVal[i][m] = sgn * (2.0^(cid -1) + **intAddVal**);	**cid-1**	**uimsbf**
}		
}		
}		
}		
}		
주: VVecLength의 계산에 대해서는 **Error! Reference source not found.** 을 참조한다		

[0213]

[0214] VVec(k)[i] 이것은 i-번째 채널에 대해 k-번째 HOAframe() 에 대한 V 벡터이다.

[0215] VVecLength 이 변수는 읽어낼 벡터 엘리먼트들의 개수를 표시한다.

[0216] VVecCoeffId 이 벡터는 송신된 V 벡터 계수들의 인덱스들을 포함한다.

[0217] **VecVal** 0 과 255 사이의 정수 값.

[0218] **aVal** VVectorData 의 디코딩 동안 사용되는 임시의 변수.

[0219] **huffVal** Huffman-디코딩될, Huffman 코드 워드.

[0220] **SgnVal** 이것은 디코딩 동안 사용되는 코딩된 부호 값이다.

[0221] **intAddVal** 이것은 디코딩 동안 사용되는 추가적인 정수 값이다.

[0222] NumVecIndices 벡터-양자화된 V 벡터를 역양자화하는데 사용되는 벡터들의 개수.

[0223] **WeightIdx** 벡터-양자화된 V 벡터를 역양자화하는데 사용되는 WeightValCdbk 에서의 인덱스.

[0224] nBitsW 벡터-양자화된 V 벡터를 디코딩하기 위해 **WeightIdx** 를 판독하기 위한 필드 사이즈.

[0225] WeightValCbk 양의 실수 값의 가중 계수들의 벡터를 포함하는 Huffman 코드 워드. 단지 NumVecIndices 가 > 1 이면 필요하다. 256 개의 엔트리들을 가지는 WeightValCdbk 가 제공된다.

[0226] WeightValPredCdbk 예측 가중 계수들의 벡터를 포함하는 코드북. 단지 NumVecIndices 가 > 1 이면 필요하다. 256 개의 엔트리들을 가지는 WeightValPredCdbk 가 제공된다.

[0227] WeightValAlpha V-벡터 양자화의 예측 코딩 모드에 대해 사용되는 예측 코딩 계수들.

[0228] **VvecIdx** 벡터-양자화된 V 벡터를 역양자화하는데 사용되는, VecDict 에 대한 인덱스.

[0229] nbitsIdx 벡터-양자화된 V 벡터를 디코딩하기 위해 **VvecIdx** 를 판독하기 위한 필드 사이즈.

[0230] WeightVal 벡터-양자화된 V 벡터를 디코딩하기 위한 실수 값의 가중 계수.

[0231] 전술한 신택스 테이블에서, 추출 유닛 (72) 은 NbitsQ 신택스 엘리먼트의 값이 4 와 동일한 (또는, 즉, 벡터 역양자화가 V-벡터를 복원하는데 사용된다는 것을 시그널링하는) 지 여부를 결정할 수도 있다. NbitsQ 신택스 엘리먼트의 값이 4 와 동일할 때, 추출 유닛 (72) 은 NumVecIndices 신택스 엘리먼트의 값을 1 의 값과 비교할 수도 있다. NumVecIndices 의 값이 1 과 동일할 때, 추출 유닛 (72) 은 VecIdx 신택스 엘리먼트를 획득할 수도 있다. VecIdx 신택스 엘리먼트는 벡터 양자화된 V-벡터를 역양자화하는데 사용되는 VecDict 에 대한 인덱스를 표시하는 하나 이상의 비트들을 나타낼 수도 있다. 추출 유닛 (72) 은 제로-번째 엘리먼트가 VecIdx 신택스 엘리먼트의 값 플러스 1 로 설정된 상태에서, VecIdx 어레이를 인스턴스화할 수도 있다. 추출 유닛 (72) 은 또한 SgnVal 신택스 엘리먼트를 획득할 수도 있다. SgnVal 신택스 엘리먼트는 V-벡터의 디코딩 동안 사용되는 코딩된 부호 값을 표시하는 하나 이상의 비트들을 나타낼 수도 있다. 추출 유닛 (72) 은 WeightVal 어레이를 인스턴스화하고, 제로-번째 엘리먼트를 SgnVal 신택스 엘리먼트의 값의 함수로서 설정할 수도 있다.

[0232] NumVecIndices 신택스 엘리먼트의 값이 1 의 값과 동일하지 않을 때, 추출 유닛 (72) 은 WeightIdx 신택스 엘리먼트를 획득할 수도 있다. WeightIdx 신택스 엘리먼트는 벡터 양자화된 V-벡터를 역양자화하는데 사용되는 WeightValCdbk 어레이에서의 인덱스를 표시하는 하나 이상의 비트들을 나타낼 수도 있다. WeightValCdbk 어레이는 양의 실수 값의 가중 계수들의 벡터를 포함하는 코드북을 나타낼 수도 있다. 추출 유닛 (72) 은 다음으로, nbitsIdx 를 HOAConfig 컨테이너에 규정된 (일 예로서 비트스트림 (21) 의 시작부분에 규정된) NumOfHoaCoeffs 신택스 엘리먼트의 함수로서 결정할 수도 있다. 추출 유닛 (72) 은 그후 NumVecIndices 를 통해서 반복하여 비트스트림 (21) 으로부터 VecIdx 신택스 엘리먼트를 획득하고, 각각의 획득된 VecIdx 신택스 엘리먼트로 VecIdx 어레이 엘리먼트들을 설정할 수도 있다.

[0233] 추출 유닛 (72) 은 비트스트림 (21) 으로부터의 신택스 엘리먼트들의 추출과 비관련되는 tmpWeightVal 변수 값들을 결정하는 것을 포함하는, 다음 PFlag 신택스 비교를 수행하지 않는다. 이와 같이, 추출 유닛 (72) 은 다음으로, WeightVal 신택스 엘리먼트를 결정하는데 사용하기 위한 SgnVal 신택스 엘리먼트를 획득할 수도 있다.

[0234] NbitsQ 신택스 엘리먼트의 값이 (Huffman 디코딩에 의하지 않는 스칼라 역양자화가 V-벡터를 복원하는데 사용된다는 것을 시그널링하는) 5 와 동일할 때, 추출 유닛 (72) 은 0 으로부터 VVecLength 까지 반복하여, aVal 변수를 비트스트림 (21) 으로부터 획득된 VecVal 신택스 엘리먼트로 설정한다. VecVal 신택스 엘리먼트는 0 과 255 사이의 정수를 표시하는 하나 이상의 비트들을 나타낼 수도 있다.

[0235] NbitsQ 신택스 엘리먼트의 값이 (Huffman 디코딩에 의한 NbitsQ-비트 스칼라 역양자화가 V-벡터를 복원하는데 사용된다는 것을 시그널링하는) 6 이상일 때, 추출 유닛 (72) 은 0 으로부터 VVecLength 까지 반복하여, huffVal, SgnVal, 및 intAddVal 신택스 엘리먼트들 중 하나 이상을 획득한다. huffVal 신택스 엘리먼트는 Huffman 코드 워드를 표시하는 하나 이상의 비트들을 나타낼 수도 있다. intAddVal 신택스 엘리먼트는 디코딩 동안 사용되는 추가적인 정수 값들을 표시하는 하나 이상의 비트들을 나타낼 수도 있다. 추출 유닛 (72) 은 이들 신택스 엘리먼트들을 벡터-기반 복원 유닛 (92) 에 제공할 수도 있다.

[0236] 벡터-기반 복원 유닛 (92) 은 HOA 계수들 (11') 을 복원하기 위해, 벡터-기반의 합성 유닛 (27) 에 대해 위에서 설명된 동작들과는 반대인 동작을 수행하도록 구성된 유닛을 나타낼 수도 있다. 벡터 기반의 복원 유닛

(92) 은 V-벡터 복원 유닛 (74), 시공간적 내삽 유닛 (76), 포그라운드 포뮬레이션 유닛 (78), 음향심리 디코딩 유닛 (80), HOA 계수 포뮬레이션 유닛 (82), 페이드 유닛 (770), 및 재정리 유닛 (84) 을 포함할 수도 있다. 페이드 유닛 (770) 의 파선들은 페이드 유닛 (770) 이 벡터-기반 복원 유닛 (92) 에 포함되는 관점에서, 옵션적인 유닛일 수도 있다는 것을 표시한다.

[0237] V-벡터 복원 유닛 (74) 은 인코딩된 포그라운드 V[k] 벡터들 (57) 로부터 V-벡터들을 복원하도록 구성된 유닛을 나타낼 수도 있다. V-벡터 복원 유닛 (74) 은 양자화 유닛 (52) 의 방법과는 반대인 방법으로 동작할 수도 있다.

[0238] V-벡터 복원 유닛 (74) 은, 즉, V-벡터들을 복원하기 위해 다음 의사 코드에 따라서 동작할 수도 있다:

```
if (NbitsQ(k)[i] == 4){
    if (NumVvecIndicies == 1){
        for (m=0; m< VVecLength; ++m){
            idx = VVecCoeffID[m];
            v^(i)_VVecCoeffId[m](k) = WeightVal[0] * VecDict[900].[VecIdx[0]][idx];
        }
    } else {
        cdbLen = O;
        if (N==4)
            cdbLen = 32;
        if
        for (m=0; m< O; ++m){
            TmpVVec[m] = 0;
            for (j=0; j< NumVecIndecies; ++j){
                TmpVVec[m] += WeightVal[j] * VecDict[cdbLen].[VecIdx[j]][m];
            }
        }
        FNorm = 0.0;
        for (m=0; m < O; ++ m) {
            FNorm += TmpVVec[m] * TmpVVec[m];
        }
        FNorm = (N+1)/sqrt(FNorm);
        for (m=0; m< VVecLength; ++m){
```

[0239] idx = VVecCoeffID[m];

```
                $v^{(i)}_{\text{VVecCoeffId[m]}}(k)$= TmpVVec[idx] * FNorm;
        }
    }
}
elseif (NbitsQ(k)[i] == 5){
    for (m=0; m< VVecLength; ++m){
        $v^{(i)}_{\text{VVecCoeffId[m]}}(k)$ =    (N+1)*aVal[i][m];
    }
}
elseif (NbitsQ(k)[i] >= 6){
    for (m=0; m< VVecLength; ++m){
        $v^{(i)}_{\text{VVecCoeffId[m]}}(k)$ = (N+1)*(2^(16 - NbitsQ(k)[i])*aVal[i][m])/2^15;
        if (PFlag(k)[i] == 1) {
            $v^{(i)}_{\text{VVecCoeffId[m]}}(k)$ += $v^{(i)}_{\text{VVecCoeffId[m]}}(k-1)$;
        }
    }
}
```

[0240]

[0241] 상기 의사 코드에 따라서, V-벡터 복원 유닛 (74) 은 i 번째 전송 채널의 k 번째 프레임에 대한 NbitsQ 신택스 엘리먼트를 획득할 수도 있다. NbitsQ 신택스 엘리먼트가 (또한, 벡터 양자화가 수행되었다는 것을 시그널링하는) 4 와 동일할 때, V-벡터 복원 유닛 (74) 은 NumVecIndicies 신택스 엘리먼트를 1 과 비교할 수도 있다. NumVecIndicies 신택스 엘리먼트는, 위에서 설명한 바와 같이, 벡터-양자화된 V-벡터를 역양자화하는데 사용되는 벡터들의 개수를 표시하는 하나 이상의 비트들을 나타낼 수도 있다. NumVecIndicies 신택스 엘리먼트의 값이 1 과 동일할 때, V-벡터 복원 유닛 (74) 은 그후 제로로부터 VVecLength 신택스 엘리먼트의 값까지 반복하여, idx 변수를 VVecCoeffId 로, 그리고 VVecCoeffId 번째 V-벡터 엘리먼트 $(v^{(i)}_{\text{VVecCoeffId[m]}}(k))$ 를 [900] [VecIdx[0]][idx] 로 식별되는 VecDict 엔트리로 곱한 WeightVal 로 설정할 수도 있다. 다시 말해서, NumVvecIndicies 의 값이 1 과 동일할 때, 벡터 코드북 HOA 확장 계수들은 테이블 F.11 에 나타낸 8x1 가중값들의 코드북과 연계하여, 테이블 F.8 로부터 유도된다.

[0242] NumVecIndicies 신택스 엘리먼트의 값이 1 과 동일하지 않을 때, V-벡터 복원 유닛 (74) 은 cdbLen 변수를, 벡터들의 개수를 표시하는 변수인 0 로 설정할 수도 있다. cdbLen 신택스 엘리먼트는 코드 벡터들의 사전 또는 코드북에서 엔트리들의 개수를 표시한다 (여기서, 이 사전은 전술한 의사 코드에서 "VecDict" 로서 표시되며, 벡터 양자화된 V-벡터를 디코딩하는데 사용되는, HOA 확장 계수들의 벡터들을 포함하는 cdbLen 코드북 엔트리들을 가지는 코드북을 나타낸다). HOA 계수들 (11) 의 ("N" 으로 표시된) 차수가 4 와 동일할 때, V-벡터 복원 유닛 (74) 은 cdbLen 변수를 32 로 설정할 수도 있다. V-벡터 복원 유닛 (74) 은 다음으로, 제로로부터 0 까지 반복하여, TmpVVec 어레이를 제로로 설정할 수도 있다. 이 반복들 동안, v-벡터 복원 유닛 (74) 은 또한, 제로로부터 NumVecIndecies 신택스 엘리먼트의 값까지 반복하여, TempVVec 어레이의 m 번째 엔트리를 VecDict 의 [cdbLen][VecIdx[j]][m] 엔트리로 곱한 j 번째 WeightVal 과 동일하게 설정할 수도 있다.

[0243] V-벡터 복원 유닛 (74) 은 다음 의사 코드에 따라서 WeightVal 을 유도할 수도 있다:

```
for (j=0; j< NumVecIndices(k)[i]; ++j) {
    if (PFlag[i] == 0) {
        tmpWeightVal(k) [j] =
            WeightValCdbk[CodebkIdx(k)[i]][WeightIdx][j];
    else {
        tmpWeightVal(k) [j] =
            WeightValPredCdbk[CodebkIdx(k)[i]][WeightIdx][j]
            + WeightValAlpha[j] * tmpWeightVal(k-1) [j];
    }
    WeightVal[j] = ((SgnVal*2)-1) * tmpWeightVal(k) [j];
```

[0244]

[0245] 전술한 의사 코드에서, V-벡터 복원 유닛 (74) 은 제로로부터 NumVecIndices 신택스 엘리먼트의 값까지 반복하여, PFlag 신택스 엘리먼트의 값이 제로와 동일한지 여부를 먼저 결정할 수도 있다. PFlag 신택스 엘리먼트가 제로와 동일할 때, V-벡터 복원 유닛 (74) 은 tmpWeightVal 변수를 결정하여, tmpWeightVal 변수를 WeightValCdbk 코드북의 [CodebkIdx][WeightIdx] 엔트리와 동일하게 설정할 수도 있다. PFlag 신택스 엘리먼트의 값이 제로와 동일하지 않을 때, V-벡터 복원 유닛 (74) 은 tmpWeightVal 변수를, WeightValPredCdbk 코드북의 [CodebkIdx][WeightIdx] 엔트리 플러스 i 번째 전송 채널의 k-1 번째 프레임의 tempWeightVal 로 곱한 WeightValAlpha 변수와 동일하게 설정할 수도 있다. WeightValAlpha 변수는 오디오 인코딩 및 디코딩 디바이스들 (20 및 24) 에서 정적으로 정의될 수도 있는, 위에서 언급된 알파 값을 지칭할 수도 있다. V-벡터 복원 유닛 (74) 은 그후 WeightVal 를 추출 유닛 (72) 에 의해 획득된 SgnVal 신택스 엘리먼트 및 tmpWeightVal 변수의 함수로서 획득할 수도 있다.

[0246] V-벡터 복원 유닛 (74) 은 즉, 가중 값 코드북 (비-예측된 벡터 양자화에 대해 "WeightValCdbk" 로서 그리고 예측된 벡터 양자화에 대해 "WeightValPredCdbk" 로서 표시되며, 이 양쪽은 (전술한 VVectorData(i) 신택스 테이블에서 "CodebkIdx" 신택스 엘리먼트로 표시된) 코드북 인덱스 및 (전술한 VVectorData(i) 신택스 테이블에서 "WeightIdx" 신택스 엘리먼트로 표시된) 가중 인덱스 중 하나 이상에 기초하여 인덱싱된 다차원의 테이블을 나타낼 수도 있다) 에 기초하여, V-벡터를 복원하는데 사용되는 각각의 대응하는 코드 벡터에 대한 가중 값을 유도할 수도 있다. 이 CodebkIdx 신택스 엘리먼트는 아래 ChannelSideInfoData(i) 신택스 테이블에서 나타낸 바와 같이, 부 채널 정보의 부분에 정의될 수도 있다.

[0247] 상기 의사 코드의 나머지 벡터 양자화 부분은, V-벡터의 엘리먼트들을 정규화하기 위한 FNorm 의 계산, 및 뒤이어서, FNorm 로 곱한 TmpVVec[idx] 와 동일한 V-벡터 엘리먼트 $\left(v^{(i)}_{\mathrm{VVecCoeffId[m]}}(k)\right)$ 의 계산에 관한 것이다. V-벡터 복원 유닛 (74) 은 idx 변수를 VVecCoeffID 의 함수로서 획득할 수도 있다.

[0248] NbitsQ 가 5 와 동일할 때, 균일한 8 비트 스칼라 역양자화가 수행된다. 이에 반해, 6 이상의 NbitsQ 값은 Huffman 디코딩의 적용을 초래할 수도 있다. 위에서 언급된 cid 값은 NbitsQ 값의 2개의 최하위 비트들과 동일할 수도 있다. 예측 모드는 상기 신택스 테이블에 PFlag 로서 표시되며, 한편 Huffman 테이블 정보 비트는 상기 신택스 테이블에 CbFlag 로 표시된다. 나머지 신택스는 어떻게 디코딩이 위에서 설명된 방법과 실질적으로 유사한 방법으로 발생하는지를 규정한다.

[0249] 음향심리 디코딩 유닛 (80) 은 인코딩된 주변 HOA 계수들 (59) 및 인코딩된 nFG 신호들 (61) 을 디코딩하여 에너지 보상된 주변 HOA 계수들 (47') 및 (내삽된 nFG 오디오 오브젝트들 (49') 로서 또한 지칭될 수도 있는) 내삽된 nFG 신호들 (49') 을 발생시키기 위해 도 3 의 예에 나타낸 음향심리 오디오 코더 유닛 (40) 과 반대인 방법으로 동작할 수도 있다. 음향심리 디코딩 유닛 (80) 은 에너지 보상된 주변 HOA 계수들 (47') 을 페이드 유닛 (770) 으로, 그리고 nFG 신호들 (49') 을 포그라운드 포뮬레이션 유닛 (78) 으로 전달할 수도 있다.

[0250] 시공간적 내삽 유닛 (76) 은 시공간적 내삽 유닛 (50) 에 대해 위에서 설명한 방법과 유사한 방법으로 동작할 수도 있다. 시공간적 내삽 유닛 (76) 은 감소된 포그라운드 V[k] 벡터들 (55_k) 을 수신하고 포그라운드

V[k] 벡터들 (55_k) 및 감소된 포그라운드 V[k-1] 벡터들 (55_{k-1}) 에 대해 시공간적 내삽을 수행하여 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 을 발생시킬 수도 있다. 시공간적 내삽 유닛 (76) 은 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 을 페이드 유닛 (770) 으로 포워딩할 수도 있다.

[0251] 추출 유닛 (72) 은 또한 주변 HOA 계수들 중 하나가 전이 중인 시점을 나타내는 신호 (757) 를 페이드 유닛 (770) 으로 출력할 수도 있으며, 그 페이드 유닛은 그후 SHC_{BG} (47') (여기서, SHC_{BG} (47') 는 또한 "주변 HOA 채널들 (47')" 또는 "주변 HOA 계수들 (47')" 로서 표시될 수도 있다) 및 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 의 엘리먼트들 중 어느 것이 페이드-인되거나 또는 페이드-아웃되는지를 결정할 수도 있다. 일부 예들에서, 페이드 유닛 (770) 은 주변 HOA 계수들 (47') 및 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 의 엘리먼트들의 각각에 대해 반대로 동작할 수도 있다. 즉, 페이드 유닛 (770) 은 주변 HOA 계수들 (47') 의 대응하는 하나에 대해 페이드-인 또는 페이드-아웃, 또는 페이드-인 또는 페이드-아웃 양쪽을 수행할 수도 있지만, 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 의 엘리먼트들의 대응하는 하나에 대해 페이드-인 또는 페이드-아웃 또는 페이드-인 및 페이드-아웃 양쪽을 수행할 수도 있다. 페이드 유닛 (770) 은 조정된 주변 HOA 계수들 (47'') 을 HOA 계수 포뮬레이션 유닛 (82) 으로, 그리고, 조정된 포그라운드 V[k] 벡터들 ($55_k'''$) 을 포그라운드 포뮬레이션 유닛 (78) 으로 출력할 수도 있다. 이 점에서, 페이드 유닛 (770) 은 예컨대, 주변 HOA 계수들 (47') 및 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 의 엘리먼트들의 유형인, HOA 계수들 또는 그의 유도체들의 여러 양태들에 대해 페이드 동작을 수행하도록 구성된 유닛을 나타낸다.

[0252] 포그라운드 포뮬레이션 유닛 (78) 은 포그라운드 HOA 계수들 (65) 을 발생시키기 위해 조정된 포그라운드 V[k] 벡터들 ($55_k'''$) 및 내삽된 nFG 신호들 (49') 에 대해 매트릭스 곱셈을 수행하도록 구성된 유닛을 나타낼 수도 있다. 이 점에서, 포그라운드 포뮬레이션 유닛 (78) 은 (내삽된 nFG 신호들 (49') 을 표시하는 다른 방식인) 오디오 오브젝트들 (49') 을 벡터들 ($55_k'''$) 과 결합하여, HOA 계수들 (11') 의 포그라운드 또는, 즉, 지배적인 양태들을 복원할 수도 있다. 포그라운드 포뮬레이션 유닛 (78) 은 조정된 포그라운드 V[k] 벡터들 ($55_k'''$) 에 의한 내삽된 nFG 신호들 (49') 의 매트릭스 곱셈을 수행할 수도 있다.

[0253] HOA 계수 포뮬레이션 유닛 (82) 은 HOA 계수들 (11') 을 획득하기 위해 포그라운드 HOA 계수들 (65) 을 조정된 주변 HOA 계수들 (47'') 에 결합하도록 구성된 유닛을 나타낼 수도 있다. 프라임 표기는 HOA 계수들 (11') 이 HOA 계수들 (11) 과 유사하지만 동일하지 않을 수도 있다는 것을 반영한다. HOA 계수들 (11 및 11') 사이의 차이들은 손실되는 전송 매체, 양자화 또는 다른 손실되는 동작들을 통한 송신으로 인해 손실을 초래할 수도 있다.

[0254] 도 5a 는 본 개시물에서 설명되는 벡터-기반 합성 기법들의 여러 양태들을 수행할 때에, 도 3 의 예에 나타낸 오디오 인코딩 디바이스 (20) 와 같은, 오디오 인코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다. 먼저, 오디오 인코딩 디바이스 (20) 는 HOA 계수들 (11) 을 수신한다 (106). 오디오 인코딩 디바이스 (20) 는 LIT 유닛 (30) 을 호출할 수도 있으며, 그 LIT 유닛은 HOA 계수들에 대해 LIT 를 적용하여 변환된 HOA 계수들을 출력할 수도 있다 (예컨대, SVD 의 경우, 변환된 HOA 계수들은 US[k] 벡터들 (33) 및 V[k] 벡터들 (35) 을 포함할 수도 있다) (107).

[0255] 오디오 인코딩 디바이스 (20) 는 다음으로, US[k] 벡터들 (33), US[k-1] 벡터들 (33), V[k] 및/또는 V[k-1] 벡터들 (35) 의 임의의 조합에 대해 상기 설명된 분석을 수행하여 여러 파라미터들을 위에서 설명된 방법으로 식별하기 위해 파라미터 계산 유닛 (32) 을 호출할 수도 있다. 즉, 파라미터 계산 유닛 (32) 은 변환된 HOA 계수들 (33/35) 의 분석에 기초하여 적어도 하나의 파라미터를 결정할 수도 있다 (108).

[0256] 오디오 인코딩 디바이스 (20) 는 그후 재정리 유닛 (34) 을 호출할 수도 있으며, 그 재정리 유닛은 위에서 설명한 바와 같이, 파라미터에 기초하여 (또한, SVD 의 상황에서, US[k] 벡터들 (33) 및 V[k] 벡터들 (35) 을 지칭할 수도 있는) 변환된 HOA 계수들을 재정리하여, 재정리된 변환된 HOA 계수들 (33'/35') (또는, 즉, US[k] 벡터들 (33') 및 V[k] 벡터들 (35')) 을 발생시킬 수도 있다 (109). 오디오 인코딩 디바이스 (20) 는 전술한 동작들 또는 후속 동작들 중 임의의 동작 동안, 음장 분석 유닛 (44) 을 또한 호출할 수도 있다. 음장 분석 유닛 (44) 은 위에서 설명한 바와 같이, HOA 계수들 (11) 및/또는 변환된 HOA 계수들 (33/35) 에 대해서 음장 분석을 수행하여, (도 3 의 예에서 백그라운드 채널 정보 (43) 로서 일괄하여 표시될 수도 있는) 전송할 포그라운드 채널들의 총 개수 (nFG) (45), 백그라운드 음장의 차수 (N_{BG}) 및 추가적인 BG HOA 채널들의 개수 (nBGa)

및 인덱스들 (i) 를 결정할 수도 있다 (109).

[0257] 오디오 인코딩 디바이스 (20) 는 또한 백그라운드 선택 유닛 (48) 을 호출할 수도 있다. 백그라운드 선택 유닛 (48) 은 백그라운드 채널 정보 (43) 에 기초하여 백그라운드 또는 주변 HOA 계수들 (47) 을 결정할 수도 있다 (110). 오디오 인코딩 디바이스 (20) 는 포그라운드 선택 유닛 (36) 을 추가로 호출할 수도 있으며, 이 포그라운드 선택 유닛은 음장의 포그라운드 또는 특유한 구성요소들을 나타내는 재정리된 US[k] 벡터들 (33') 및 재정리된 V[k] 벡터들 (35') 을 (포그라운드 벡터들을 식별하는 하나 이상의 인덱스들을 나타낼 수도 있는) nFG (45) 에 기초하여 선택할 수도 있다 (112).

[0258] 오디오 인코딩 디바이스 (20) 는 에너지 보상 유닛 (38) 을 호출할 수도 있다. 에너지 보상 유닛 (38) 은 주변 HOA 계수들 (47) 에 대해 에너지 보상을 수행하여, 백그라운드 선택 유닛 (48) 에 의한 HOA 계수들의 여러 HOA 계수들의 제거로 인한 에너지 손실을 보상하고 (114), 이에 따라서 에너지 보상된 주변 HOA 계수들 (47') 을 발생시킬 수도 있다.

[0259] 오디오 인코딩 디바이스 (20) 는 또한 시공간적 내삽 유닛 (50) 을 호출할 수도 있다. 시공간적 내삽 유닛 (50) 은 재정리된 변환된 HOA 계수들 (33'/35') 에 대해 시공간적 내삽을 수행하여 ("내삽된 nFG 신호들 (49')" 로서 또한 지칭될 수도 있는) 내삽된 포그라운드 신호들 (49') 및 ("V[k] 벡터들 (53)" 로서 또한 지칭될 수도 있는) 나머지 포그라운드 방향 정보 (53) 를 획득할 수도 있다 (116). 오디오 인코딩 디바이스 (20) 는 그 후 계수 감소 유닛 (46) 을 호출할 수도 있다. 계수 감소 유닛 (46) 은 백그라운드 채널 정보 (43) 에 기초하여 나머지 포그라운드 V[k] 벡터들 (53) 에 대해 계수 감소를 수행하여, (감소된 포그라운드 V[k] 벡터들 (55) 로서 또한 지칭될 수도 있는) 감소된 포그라운드 방향 정보 (55) 를 획득할 수도 있다 (118).

[0260] 오디오 인코딩 디바이스 (20) 는 그후 양자화 유닛 (52) 을 호출하여, 위에서 설명된 방법으로, 감소된 포그라운드 V[k] 벡터들 (55) 을 압축하여, 코딩된 포그라운드 V[k] 벡터들 (57) 을 발생시킬 수도 있다 (120).

[0261] 오디오 인코딩 디바이스 (20) 는 또한 음향심리 오디오 코더 유닛 (40) 을 호출할 수도 있다. 음향심리 오디오 코더 유닛 (40) 은 내삽된 nFG 신호들 (49') 및 에너지 보상된 주변 HOA 계수들 (47') 의 각각의 벡터를 음향심리 코딩하여, 인코딩된 주변 HOA 계수들 (59) 및 인코딩된 nFG 신호들 (61) 을 발생시킬 수도 있다. 오디오 인코딩 디바이스는 그후 비트스트림 발생 유닛 (42) 을 호출할 수도 있다. 비트스트림 발생 유닛 (42) 은 코딩된 포그라운드 방향 정보 (57), 코딩된 주변 HOA 계수들 (59), 코딩된 nFG 신호들 (61) 및 백그라운드 채널 정보 (43) 에 기초하여, 비트스트림 (21) 을 발생시킬 수도 있다.

[0262] 도 5b 는 본 개시물에서 설명되는 코딩 기법들을 수행할 때에 오디오 인코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다. 도 3 의 예에 나타낸 오디오 인코딩 디바이스 (20) 의 비트스트림 발생 유닛 (42) 은 본 개시물에서 설명하는 기법들을 수행하도록 구성된 일 예시적인 유닛을 나타낼 수도 있다. 비트스트림 발생 유닛 (42) 은 프레임의 양자화 모드가 ("제 2 프레임" 으로서 표시될 수도 있는) 시간적으로 이전 프레임의 양자화 모드와 동일한지 여부를 결정할 수도 있다 (314). 이전 프레임에 대해 설명되었지만, 본 기법들은 시간적으로 후속 프레임들에 대해 수행될 수도 있다. 프레임은 하나 이상의 전송 채널들의 부분을 포함할 수도 있다. 전송 채널의 부분은 일부 페이로드 (예컨대, 도 7 의 예에서의 VVectorData 필드들 (156)) 와 함께, (ChannelSideInfoData 신택스 테이블에 따라서 형성된) ChannelSideInfoData 를 포함할 수도 있다. 페이로드들의 다른 예들은 AddAmbientHOACoeffs 필드들을 포함할 수도 있다.

[0263] 양자화 모드들이 동일할 때 ("예" 316), 비트스트림 발생 유닛 (42) 은 비트스트림 (21) 에 양자화 모드의 부분을 규정할 수도 있다 (318). 양자화 모드의 부분은 bA 신택스 엘리먼트 및 bB 신택스 엘리먼트를 포함하지만 uintC 신택스 엘리먼트를 포함하지 않을 수도 있다. bA 신택스 엘리먼트는 NbitsQ 신택스 엘리먼트의 최상위 비트를 표시하는 비트를 나타낼 수도 있다. bB 신택스 엘리먼트는 NbitsQ 신택스 엘리먼트의 제 2 최상위 비트를 표시하는 비트를 나타낼 수도 있다. 비트스트림 발생 유닛 (42) 은 bA 신택스 엘리먼트 및 bB 신택스 엘리먼트의 각각의 값을 제로로 설정함으로써, 비트스트림 (21) 에서의 양자화 모드 필드 (즉, 일 예로서 NbitsQ 필드) 가 uintC 신택스 엘리먼트를 포함하지 않는다는 것을 시그널링할 수도 있다. 이 제로 값 bA 신택스 엘리먼트 및 bB 신택스 엘리먼트의 시그널링은 또한 이전 프레임으로부터의 NbitsQ 값, PFlag 값, CbFlag 값, 및 CodebkIdx 값이 현재의 프레임의 동일한 신택스 엘리먼트들에 대한 대응하는 값들로서 사용된다는 것을 표시한다.

[0264] 양자화 모드들이 동일하지 않을 때 ("아니오" 316), 비트스트림 발생 유닛 (42) 은 비트스트림 (21) 에 전체 양자화 모드를 표시하는 하나 이상의 비트들을 규정할 수도 있다 (320). 즉, 비트스트림 발생 유닛 (42) 은

비트스트림 (21) 에 bA, bB 및 uintC 신택스 엘리먼트들을 규정한다. 비트스트림 발생 유닛 (42) 은 또한 양자화 모드에 기초하여 양자화 정보를 규정할 수도 있다 (322). 이 양자화 정보는 벡터 양자화 정보, 예측 정보, 및 Huffman 코드북 정보와 같은, 양자화에 관련된 임의의 정보를 포함할 수도 있다. 벡터 양자화 정보는 일 예로서, CodebkIdx 신택스 엘리먼트 및 NumVecIndices 신택스 엘리먼트 중 하나 (또는, 양쪽) 를 포함할 수도 있다. 예측 정보는 일 예로서, PFlag 신택스 엘리먼트를 포함할 수도 있다. Huffman 코드북 정보는 일 예로서, CbFlag 신택스 엘리먼트를 포함할 수도 있다.

[0265] 이 점에서, 본 기법들은 오디오 인코딩 디바이스 (20) 가 음장의 공간 구성요소의 압축된 버전을 포함하는 비트스트림 (21) 을 획득하도록 구성될 수 있게 할 수도 있다. 공간 구성요소는 복수의 구면 고조파 계수들에 대해 벡터 기반의 합성을 수행함으로써 발생될 수도 있다. 비트스트림은 헤더 필드의 하나 이상의 비트들을 재사용할지 여부에 대한, 이전 프레임으로부터의, 표시자를 더 포함할 수도 있다.

[0266] 다시 말해서, 본 기법들은 오디오 인코딩 디바이스 (20) 가, 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터 (57) 를 포함하는 비트스트림 (21) 을 획득하도록 구성될 수 있게 할 수도 있다. 비트스트림 (21) 은 벡터를 압축하는데 (예컨대, 양자화하는데) 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자 (예컨대, NbitsQ 신택스 엘리먼트의 bA/bB 신택스 엘리먼트들) 를 더 포함할 수도 있다.

[0267] 도 6a 는 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 때에, 도 4 에 나타낸 오디오 디코딩 디바이스 (24) 와 같은, 오디오 디코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다. 먼저, 오디오 디코딩 디바이스 (24) 는 비트스트림 (21) 을 수신할 수도 있다 (130). 비트스트림을 수신하자 마자, 오디오 디코딩 디바이스 (24) 는 추출 유닛 (72) 을 호출할 수도 있다. 논의의 목적들을 위해, 벡터-기반의 복원이 수행된다는 것을 비트스트림 (21) 이 표시한다고 가정하면, 추출 유닛 (72) 은 비트스트림을 파싱하여 상기 언급된 정보를 취출하고, 그 정보를 벡터-기반 복원 유닛 (92) 으로 전달할 수도 있다.

[0268] 다시 말해서, 추출 유닛 (72) 은 비트스트림 (21) 으로부터 위에서 설명된 방법으로 (또한, 코딩된 포그라운드 V[k] 벡터들 (57) 로서 또한 지칭될 수도 있는) 코딩된 포그라운드 방향 정보 (57), 코딩된 주변 HOA 계수들 (59) 및 (코딩된 포그라운드 nFG 신호들 (59) 또는 코딩된 포그라운드 오디오 오브젝트들 (59) 로서 또한 지칭될 수도 있는) 코딩된 포그라운드 신호들을 취출할 수도 있다 (132).

[0269] 오디오 디코딩 디바이스 (24) 는 역양자화 유닛 (74) 을 추가로 호출할 수도 있다. 역양자화 유닛 (74) 은 코딩된 포그라운드 방향 정보 (57) 를 엔트로피 디코딩하여 역양자화하여 감소된 포그라운드 방향 정보 (55_k) 를 획득할 수도 있다 (136). 오디오 디코딩 디바이스 (24) 는 또한 음향심리 디코딩 유닛 (80) 을 호출할 수도 있다. 음향심리 오디오 디코딩 유닛 (80) 은 인코딩된 주변 HOA 계수들 (59) 및 인코딩된 포그라운드 신호들 (61) 을 디코딩하여, 에너지 보상된 주변 HOA 계수들 (47') 및 내삽된 포그라운드 신호들 (49') 을 획득할 수도 있다 (138). 음향심리 디코딩 유닛 (80) 은 에너지 보상된 주변 HOA 계수들 (47') 을 페이드 유닛 (770) 으로, 그리고 nFG 신호들 (49') 을 포그라운드 포뮬레이션 유닛 (78) 으로 전달할 수도 있다.

[0270] 오디오 디코딩 디바이스 (24) 는 다음으로 시공간적 내삽 유닛 (76) 을 호출할 수도 있다. 시공간적 내삽 유닛 (76) 은 재정리된 포그라운드 방향 정보 ($55_k'$) 를 수신하고 감소된 포그라운드 방향 정보 ($55_k/55_{k-1}$) 에 대해 시공간적 내삽을 수행하여, 내삽된 포그라운드 방향 정보 ($55_k''$) 을 발생시킬 수도 있다 (140). 시공간적 내삽 유닛 (76) 은 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 을 페이드 유닛 (770) 으로 포워딩할 수도 있다.

[0271] 오디오 디코딩 디바이스 (24) 는 페이드 유닛 (770) 을 호출할 수도 있다. 페이드 유닛 (770) 은 에너지 보상된 주변 HOA 계수들 (47') 이 전이 중인 시점을 나타내는 (예컨대, 추출 유닛 (72) 으로부터의) 신택스 엘리먼트들 (예컨대, AmbCoeffTransition 신택스 엘리먼트) 을 수신하거나 또는 아니면 획득할 수도 있다. 페이드 유닛 (770) 은, 전이 신택스 엘리먼트들 및 유지된 전이 상태 정보에 기초하여, 에너지 보상된 주변 HOA 계수들 (47') 을 페이드-인 또는 페이드-아웃하여, 조정된 주변 HOA 계수들 (47'') 을 HOA 계수 포뮬레이션 유닛 (82) 으로 출력할 수도 있다. 페이드 유닛 (770) 은 또한, 신택스 엘리먼트들 및 유지된 전이 상태 정보에 기초하여, 내삽된 포그라운드 V[k] 벡터들 ($55_k''$) 의 대응하는 하나 이상의 엘리먼트들을 페이드-아웃 또는 페이드-인하여, 조정된 포그라운드 V[k] 벡터들 ($55_k'''$) 을 포그라운드 포뮬레이션 유닛 (78) 으로 출력할 수도 있다 (142).

[0272] 오디오 디코딩 디바이스 (24) 는 포그라운드 포뮬레이션 유닛 (78) 을 호출할 수도 있다. 포그라운드 포뮬레이션 유닛 (78) 은 조정된 포그라운드 방향 정보 ($55_k'''$) 에 의한 nFG 신호들 (49') 의 매트릭스 곱셈을 수행하여, 포그라운드 HOA 계수들 (65) 을 획득할 수도 있다 (144). 오디오 디코딩 디바이스 (24) 는 또한 HOA 계수 포뮬레이션 유닛 (82) 을 호출할 수도 있다. HOA 계수 포뮬레이션 유닛 (82) 은 HOA 계수들 (11') 을 획득하기 위해 포그라운드 HOA 계수들 (65) 을 조정된 주변 HOA 계수들 (47'') 에 가산할 수도 있다 (146).

[0273] 도 6b 는 본 개시물에서 설명되는 코딩 기법들을 수행할 때에 오디오 디코딩 디바이스의 예시적인 동작을 예시하는 플로우차트이다. 도 4 의 예에 나타낸 오디오 인코딩 디바이스 (24) 의 추출 유닛 (72) 은 본 개시물에서 설명되는 기법들을 수행하도록 구성된 일 예시적인 유닛을 나타낼 수도 있다. 비트스트림 추출 유닛 (72) 은 프레임의 양자화 모드가 ("제 2 프레임" 으로서 표시될 수도 있는) 시간적으로 이전 프레임의 양자화 모드와 동일한지 여부를 표시하는 비트들을 획득할 수도 있다 (362). 또, 이전 프레임에 대해 설명되었지만, 본 기법들은 시간적으로 후속 프레임들에 대해 수행될 수도 있다.

[0274] 양자화 모드들이 동일할 때 ("예" 364), 추출 유닛 (72) 은 비트스트림 (21) 으로부터 양자화 모드의 부분을 획득할 수도 있다 (366). 양자화 모드의 부분은 bA 신택스 엘리먼트 및 bB 신택스 엘리먼트를 포함하지만 uintC 신택스 엘리먼트를 포함하지 않을 수도 있다. 추출 유닛 (42) 은 또한 현재의 프레임에 대한 NbitsQ 값, PFlag 값, CbFlag 값, CodebkIdx 값, 및 NumVertIndices 값의 값들을 이전 프레임에 대해 설정된 NbitsQ 값, PFlag 값, CbFlag 값, CodebkIdx 값, 및 NumVertIndices 값의 값들과 동일하게 설정할 수도 있다 (368).

[0275] 양자화 모드들이 동일하지 않을 때 ("아니오" 364), 추출 유닛 (72) 은 비트스트림 (21) 으로부터 전체 양자화 모드를 표시하는 하나 이상의 비트들을 획득할 수도 있다. 즉, 추출 유닛 (72) 은 비트스트림 (21) 으로부터 bA, bB 및 uintC 신택스 엘리먼트들을 획득한다 (370). 추출 유닛 (72) 은 또한 양자화 모드에 기초하여 양자화 정보를 표시하는 하나 이상의 비트들을 획득할 수도 있다 (372). 도 5b 와 관련하여 위에서 언급한 바와 같이, 양자화 정보는 벡터 양자화 정보, 예측 정보, 및 Huffman 코드북 정보와 같은, 양자화에 관련된 임의의 정보를 포함할 수도 있다. 벡터 양자화 정보는 일 예로서, CodebkIdx 신택스 엘리먼트 및 NumVecIndices 신택스 엘리먼트 중 하나 (또는, 양쪽) 를 포함할 수도 있다. 예측 정보는 일 예로서, PFlag 신택스 엘리먼트를 포함할 수도 있다. Huffman 코드북 정보는 일 예로서, CbFlag 신택스 엘리먼트를 포함할 수도 있다.

[0276] 이 점에서, 본 기법들은 오디오 디코딩 디바이스 (24) 가 음장의 공간 구성요소의 압축된 버전을 포함하는 비트스트림 (21) 을 획득하도록 구성될 수 있게 할 수도 있다. 공간 구성요소는 복수의 구면 고조파 계수들에 대해 벡터 기반의 합성을 수행함으로써 발생될 수도 있다. 비트스트림은 헤더 필드의 하나 이상의 비트들을 재사용할지 여부에 대한, 이전 프레임으로부터의, 표시자를 더 포함할 수도 있다.

[0277] 다시 말해서, 본 기법들은 오디오 디코딩 디바이스 (24) 가 구면 고조파들 도메인에서의 직교 공간 축을 나타내는 벡터 (57) 를 포함하는 비트스트림 (21) 을 획득하도록 구성될 수 있게 할 수도 있다. 비트스트림 (21) 은 벡터를 압축하는데 (예컨대, 양자화하는데) 사용되는 정보를 표시하는 적어도 하나의 신택스 엘리먼트를 이전 프레임으로부터 재사용할지 여부에 대한 표시자 (예컨대, NbitsQ 신택스 엘리먼트의 bA/bB 신택스 엘리먼트들) 를 더 포함할 수도 있다.

[0278] 도 7 은 본 개시물에서 설명하는 기법들의 여러 양태들에 따라서 규정된 예시적인 프레임들 (249S 및 249T) 을 예시하는 다이어그램이다. 도 7 의 예에 나타낸 바와 같이, 프레임 (249S) 은 ChannelSideInfoData (CSID) 필드들 (154A-154D), HOAGainCorrectionData (HOAGCD) 필드들, VVectorData 필드들 (156A 및 156B) 및 HOAPredictionInfo 필드들을 포함한다. CSID 필드 (154A) 는 01 의 값으로 설정된 ChannelType 신택스 엘리먼트 ("ChannelType") (269) 와 함께, 10 의 값으로 설정된 uintC 신택스 엘리먼트 ("uintC") (267), 1 의 값으로 설정된 bb 신택스 엘리먼트 ("bB") (266) 및 0 의 값으로 설정된 bA 신택스 엘리먼트 ("bA") (265) 를 포함한다.

[0279] uintC 신택스 엘리먼트 (267), bB 신택스 엘리먼트 (266) 및 bA 신택스 엘리먼트 (265) 는 함께 NbitsQ 신택스 엘리먼트 (261) 를 형성하며, 여기서, bA 신택스 엘리먼트 (265) 는 최상위 비트를 형성하며, bB 신택스 엘리먼트 (266) 는 제 2 최상위 비트를 형성하며, 그리고 uintC 신택스 엘리먼트 (267) 는 NbitsQ 신택스 엘리먼트 (261) 의 최하위 비트들을 형성한다. NbitsQ 신택스 엘리먼트 (261) 는 위에서 언급한 바와 같이, 고-차수 앰비소닉 오디오 데이터를 인코딩하는데 사용되는 양자화 모드 (예컨대, 벡터 양자화 모드, Huffman 코딩 모드에 의하지 않은 스칼라 양자화, 및 Huffman 코딩 모드에 의한 스칼라 양자화 중 하나) 를 표시하는 하나 이상의

비트들을 나타낼 수도 있다.

[0280] CSID 신택스 엘리먼트 (154A) 는 또한 여러 신택스 테이블들에서 위에서 언급된, PFlag 신택스 엘리먼트 (300) 및 CbFlag 신택스 엘리먼트 (302) 를 포함한다. PFlag 신택스 엘리먼트 (300) 는 제 1 프레임 (249S) 의 HOA 계수들 (11) 에 의해 표현되는 음장의 공간 구성요소의 코딩된 엘리먼트 (여기서, 또한 공간 구성요소는 V-벡터를 지칭할 수도 있다) 가 제 2 프레임 (예컨대, 이 예에서 이전 프레임) 으로부터 예측되는지 여부를 표시하는 하나 이상의 비트들을 나타낼 수도 있다. CbFlag 신택스 엘리먼트 (302) 는 Huffman 코드북들 (또는, 즉, 테이블들) 중 어느 코드북이 공간 구성요소의 엘리먼트들 (또는, 즉, V-벡터 엘리먼트들) 을 인코딩하는데 사용되는지를 식별할 수도 있는 Huffman 코드북 정보를 표시하는 하나 이상의 비트들을 나타낼 수도 있다.

[0281] CSID 필드 (154B) 는 bB 신택스 엘리먼트 (266) 및 bB 신택스 엘리먼트 (265) 를 ChannelType 신택스 엘리먼트 (269) 와 함께 포함하며, 이들 각각은 도 7 의 예에서 대응하는 값들 0 및 0 및 01 로 설정된다. CSID 필드들 (154C 및 154D) 의 각각은 3 의 값 (11_2) 을 가지는 ChannelType 필드 (269) 를 포함한다. CSID 필드들 (154A-154D) 의 각각은 전송 채널들 1, 2, 3 및 4 의 개개의 하나에 대응한다. 실제로, 각각의 CSID 필드 (154A-154D) 는 대응하는 페이로드가 (대응하는 ChannelType 이 제로와 동일할 때) 방향-기반 신호들인지, (대응하는 ChannelType 이 1 과 동일할 때) 벡터-기반의 신호들인지, (대응하는 ChannelType 이 2 와 동일할 때) 추가적인 주변 HOA 계수인지, 또는 (ChannelType 이 3 과 동일할 때) 공백 (empty) 인지 여부를 표시한다.

[0282] 도 7 의 예에서, 프레임 (249S) 은 (ChannelType 신택스 엘리먼트들 (269) 이 CSID 필드들 (154A 및 154B) 에서 1 과 동일하다고 가정하면) 2개의 벡터-기반의 신호들 및 (ChannelType (269) 이 CSID 필드들 (154C 및 154D) 에서 3 과 동일하다고 가정하면) 2개의 공백 (empty) 을 포함한다. 더욱이, 오디오 인코딩 디바이스 (20) 는 1 로 설정되는 PFlag 신택스 엘리먼트 (300) 에 의해 표시된 바와 같은 예측을 채용하였다. 또, PFlag 신택스 엘리먼트 (300) 에 의해 표시된 바와 같은 예측은 예측이 압축된 공간 구성요소들 v1-vn 의 대응하는 필드에 대해 수행되었는지 여부를 표시하는 예측 모드 표시를 지칭한다. PFlag 신택스 엘리먼트 (300) 가 1 로 설정될 때, 오디오 인코딩 디바이스 (20) 는 스칼라 양자화에 대해, 이전 프레임으로부터의 벡터 엘리먼트와 현재의 프레임의 대응하는 벡터 엘리먼트 사이의 차이, 또는, 벡터 양자화에 대해, 이전 프레임으로부터의 가중치와 현재의 프레임의 대응하는 가중치 사이의 차이를 취함으로써 예측을 채용할 수도 있다.

[0283] 오디오 인코딩 디바이스 (20) 는 또한 프레임 (249S) 에서의 제 2 전송 채널의 CSID 필드 (154B) 에 대한 NbitsQ 신택스 엘리먼트 (261) 에 대한 값이 이전 프레임, 예를 들어, 도 7 의 예에서 프레임 (249T) 의 제 2 전송 채널의 CSID 필드 (154B) 에 대한 NbitsQ 신택스 엘리먼트 (261) 의 값과 동일한다고 결정하였다. 그 결과, 오디오 인코딩 디바이스 (20) 는 이전 프레임 (249T) 에서 제 2 전송 채널의 NbitsQ 신택스 엘리먼트 (261) 의 값이 프레임 (249S) 에서 제 2 전송 채널의 NbitsQ 신택스 엘리먼트 (261) 에 대해 재사용된다는 것을 시그널링하기 위해 bA 신택스 엘리먼트 (265) 및 bB 신택스 엘리먼트 (266) 의 각각에 대해 제로의 값을 규정하였다. 그 결과, 오디오 인코딩 디바이스 (20) 는 위에서 식별된 다른 신택스 엘리먼트와 함께, 프레임 (249S) 에서 제 2 전송 채널에 대해 uintC 신택스 엘리먼트 (267) 를 규정하는 것을 회피할 수도 있다.

[0284] 도 8 은 본원에서 설명되는 기법들에 따른, 적어도 하나의 비트스트림의 하나 이상의 채널들에 대한 예시적인 프레임들을 예시하는 다이어그램이다. 비트스트림 (450) 은 하나 이상의 채널들을 각각 포함할 수도 있는 프레임들 (810A-810H) 을 포함한다. 비트스트림 (450) 은 도 7 의 예에 나타낸 비트스트림 (21) 의 일 예일 수도 있다. 도 8 의 예에서, 오디오 디코딩 디바이스 (24) 는 상태 정보를 유지하고, 현재의 프레임 k 를 디코딩하는 방법을 결정하기 위해 상태 정보를 업데이트한다. 오디오 디코딩 디바이스 (24) 는 config (814), 및 프레임들 (810B-810D) 로부터의 상태 정보를 이용할 수도 있다.

[0285] 다시 말해서, 오디오 인코딩 디바이스 (20) 는 비트스트림 발생 유닛 (42) 내에, 예를 들어, 비트스트림 발생 유닛 (42) 이 상태 머신 (402) 에 기초하여 프레임들 (810A-810E) 의 각각에 대한 신택스 엘리먼트들을 규정할 수도 있다는 점에서, 프레임들 (810A-810E) 의 각각을 인코딩하기 위한 상태 정보를 유지하는 상태 머신 (402) 을 포함할 수도 있다.

[0286] 오디오 디코딩 디바이스 (24) 는 유사하게, 비트스트림 추출 유닛 (72) 내에, 예를 들어, 상태 머신 (402) 에 기초하여 신택스 엘리먼트들 (이의 일부는 비트스트림 (21) 에 명시적으로 규정되지 않는다) 을 출력하는 유사한 상태 머신 (402) 을 포함할 수도 있다. 오디오 디코딩 디바이스 (24) 의 상태 머신 (402) 은 오디오 인코딩 디바이스 (20) 의 상태 머신 (402) 의 방법과 유사한 방법으로 동작할 수도 있다. 이와 같이, 오디오 디코딩 디바이스 (24) 의 상태 머신 (402) 은 상태 정보를 유지하고, config (814) 그리고, 도 8 의 예에서는 프레임들 (810B-810D) 의 디코딩에 기초하여 상태 정보를 업데이트할 수도 있다. 상태 정보에 기초하여, 비

트스트림 추출 유닛 (72) 은 상태 머신 (402) 에 의해 유지된 상태 정보에 기초하여 프레임 (810E) 을 추출할 수도 있다. 상태 정보는 오디오 인코딩 디바이스 (20) 가 프레임 (810E) 의 여러 전송 채널들을 디코딩할 때 이용할 수도 있는 다수의 암시적인 신택스 엘리먼트들을 제공할 수도 있다.

[0287] 전술한 기법들은 임의 개수의 상이한 상황들 및 오디오 생태계들에 대해 수행될 수도 있다. 다수의 예시적인 상황들이 아래에 설명되지만, 본 기법들은 예시적인 상황들에 한정되지 않아야 한다. 일 예시적인 오디오 생태계는 오디오 콘텐츠, 영화 스튜디오들, 음악 스튜디오들, 게이밍 오디오 스튜디오들, 채널 기반 오디오 콘텐츠, 코딩 엔진들, 게임 오디오 시스템들, 게임 오디오 코딩 / 렌더링 엔진들, 및 전달 시스템들을 포함할 수도 있다.

[0288] 영화 스튜디오들, 음악 스튜디오들, 및 게이밍 오디오 스튜디오들은 오디오 콘텐츠를 수신할 수도 있다. 일부 예들에서, 오디오 콘텐츠는 획득의 출력을 나타낼 수도 있다. 영화 스튜디오들은 채널 기반 오디오 콘텐츠를 (예컨대, 2.0, 5.1, 및 7.1 에서) 예컨대, 디지털 오디오 워크스테이션 (DAW) 을 이용함으로써 출력할 수도 있다. 음악 스튜디오들은 채널 기반 오디오 콘텐츠를 (예컨대, 2.0, 및 5.1 에서) 예컨대, DAW 를 이용함으로써 출력할 수도 있다. 어느 경우에나, 코딩 엔진들은 전달 시스템들에 의한 출력을 위해 채널 기반 오디오 콘텐츠 기반의 하나 이상의 코덱들 (예컨대, AAC, AC3, Dolby True HD, Dolby 디지털 플러스, 및 DTS 마스터 오디오) 을 수신하여 인코딩할 수도 있다. 게이밍 오디오 스튜디오들은 하나 이상의 게임 오디오 시스템들을, 예컨대, DAW 를 이용함으로써 출력할 수도 있다. 게임 오디오 코딩 / 렌더링 엔진들은 전달 시스템들에 의한 출력을 위해 오디오 시스템들을 채널 기반 오디오 콘텐츠로 코딩하고 및/또는 렌더링할 수도 있다.

본 기법들이 수행될 수도 있는 다른 예시적인 상황은 브로드캐스트 리코딩 오디오 오브젝트들, 전문 오디오 시스템들, 소비자 온-디바이스 캡쳐, HOA 오디오 포맷, 온-디바이스 렌더링, 소비자 오디오, TV 및 부속물들, 및 카 오디오 시스템들을 포함할 수도 있는 오디오 생태계를 포함한다.

[0289] 브로드캐스트 리코딩 오디오 오브젝트들, 전문 오디오 시스템들, 및 소비자 온-디바이스 캡쳐는 그들의 출력을 HOA 오디오 포맷을 이용하여 모두 코딩할 수도 있다. 이러한 방법으로, 오디오 콘텐츠는 HOA 오디오 포맷을 이용하여, 온-디바이스 렌더링, 소비자 오디오, TV, 및 부속물들, 및 카 오디오 시스템들을 이용하여 플레이백될 수도 있는 단일 표현으로 코딩될 수도 있다. 다시 말해서, 오디오 콘텐츠의 단일 표현은 오디오 플레이백 시스템 (16) 과 같은, (즉, 5.1, 7.1, 등과 같은 특정의 구성을 필요로 하는 것과는 반대로) 일반적인 오디오 플레이백 시스템에서 플레이백될 수도 있다.

[0290] 본 기법들이 수행될 수도 있는 상황의 다른 예들은 획득 엘리먼트들, 및 플레이백 엘리먼트들을 포함할 수도 있는 오디오 생태계를 포함한다. 획득 엘리먼트들은 유선 및/또는 무선 획득 디바이스들 (acquisition devices) (예컨대, 아이겐 (Eigen) 마이크로폰들), 온-디바이스 서라운드 사운드 캡쳐, 및 모바일 디바이스들 (예컨대, 스마트폰들 및 태블릿들) 을 포함할 수도 있다. 일부 예들에서, 유선 및/또는 무선 획득 디바이스들은 유선 및/또는 무선 통신 채널(들)을 통해서 모바일 디바이스에 커플링될 수도 있다.

[0291] 본 개시물의 하나 이상의 기법들에 따르면, 모바일 디바이스가 음장을 획득하는데 사용될 수도 있다. 예를 들어, 모바일 디바이스는 유선 및/또는 무선 획득 디바이스들 및/또는 온-디바이스 서라운드 사운드 캡쳐 (예컨대, 모바일 디바이스에 통합된 복수의 마이크로폰들) 를 통해서 음장을 획득할 수도 있다. 모바일 디바이스는 그후 플레이백 엘리먼트들 중 하나 이상에 의한 플레이백을 위해 그 획득된 음장을 HOA 계수들로 코딩할 수도 있다. 예를 들어, 모바일 디바이스의 사용자는 라이브 이벤트 (예컨대, 미팅, 회의, 연극, 콘서트, 등) 을 리코딩하여 (그의 음장을 획득하여), 그 리코딩을 HOA 계수들로 코딩할 수도 있다.

[0292] 모바일 디바이스는 또한 플레이백 엘리먼트들 중 하나 이상을 이용하여, HOA 코딩된 음장을 플레이백할 수도 있다. 예를 들어, 모바일 디바이스는 HOA 코딩된 음장을 디코딩하고, 플레이백 엘리먼트들 중 하나 이상이 음장을 재생하도록 하는 신호를 플레이백 엘리먼트들 중 하나 이상으로 출력할 수도 있다. 일 예로서, 모바일 디바이스는 무선 및/또는 무선 통신 채널들을 이용하여, 하나 이상의 스피커들 (예컨대, 스피커 어레이들, 사운드 바들, 등) 로 그 신호를 출력할 수도 있다. 다른 예로서, 모바일 디바이스는 도킹 솔루션들을 이용하여, 그 신호를 하나 이상의 도킹 스테이션들 및/또는 하나 이상의 도킹된 스피커들 (예컨대, 사운드 시스템들 in 스마트 차들 및/또는 홈들) 로 출력할 수도 있다. 다른 예로서, 모바일 디바이스는 헤드폰 렌더링을 이용하여, 예컨대, 실제적인 바이노럴 사운드를 생성하기 위해 그 신호를 헤드폰들의 세트로 출력할 수도 있다.

[0293] 일부 예들에서, 특정의 모바일 디바이스가 3D 음장을 획득할 뿐만 아니라 그 동일한 3D 음장을 추후에 플레이백할 수도 있다. 일부 예들에서, 모바일 디바이스는 플레이백을 위해, 3D 음장을 획득하고, 3D 음장을 HOA 로 인코딩하고, 그리고 인코딩된 3D 음장을 하나 이상의 다른 디바이스들 (예컨대, 다른 모바일 디바이스들 및/또

는 다른 비-모바일 디바이스들) 로 송신할 수도 있다.

[0294] 본 기법들이 수행될 수도 있는 또 다른 상황은 오디오 콘텐츠, 게임 스튜디오들, 코딩된 오디오 콘텐츠, 렌더링 엔진들, 및 전달 시스템들을 포함할 수도 있는 오디오 생태계를 포함한다. 일부 예들에서, 게임 스튜디오들은 HOA 신호들의 편집을 지원할 수도 있는 하나 이상의 DAW들을 포함할 수도 있다. 예를 들어, 하나 이상의 DAW들은 하나 이상의 게임 오디오 시스템들과 동작하도록 (예컨대, 그들과 작업하도록) 구성될 수도 있는 HOA 플러그인들 및/또는 툴들을 포함할 수도 있다. 일부 예들에서, 게임 스튜디오들은 HOA 를 지원하는 새로운 시스템 포맷들을 출력할 수도 있다. 어쨌든, 게임 스튜디오들은 전달 시스템들에 의한 플레이백을 위해, 코딩된 오디오 콘텐츠를 음장을 렌더링할 수도 있는 렌더링 엔진들로 출력할 수도 있다.

[0295] 이 기법들은 또한 예시적인 오디오 획득 디바이스들에 대해 수행될 수도 있다. 예를 들어, 이 기법들은 3D 음장을 리코딩하도록 집합하여 구성되는 복수의 마이크로폰들을 포함할 수도 있는 아이겐 (Eigen) 마이크로폰에 대해 수행될 수도 있다. 일부 예들에서, 아이겐 마이크로폰의 복수의 마이크로폰들은 대략 4cm 의 반경을 가지는 실질적으로 구형인 볼의 표면 상에 로케이트될 수도 있다. 일부 예들에서, 오디오 인코딩 디바이스 (20) 는 마이크로폰으로부터 직접 비트스트림 (21) 을 출력하기 위해 아이겐 마이크로폰에 통합될 수도 있다.

[0296] 다른 예시적인 오디오 획득 상황은 하나 이상의 아이겐 마이크로폰들과 같은 하나 이상의 마이크로폰들로부터 신호를 수신하도록 구성될 수도 있는 프로덕션 트럭을 포함할 수도 있다. 프로덕션 트럭은 또한 도 3 의 오디오 인코더 (20) 와 같은 오디오 인코더를 포함할 수도 있다.

[0297] 모바일 디바이스는 또한, 일부 경우, 3D 음장을 리코딩하도록 종합하여 구성된 복수의 마이크로폰들을 포함할 수도 있다. 다시 말해서, 복수의 마이크로폰은 X, Y, Z 다이버시티를 가질 수도 있다. 일부 예들에서, 모바일 디바이스는 모바일 디바이스의 하나 이상의 다른 마이크로폰들에 대해 X, Y, Z 다이버시티를 제공하도록 회전될 수도 있는 마이크로폰을 포함할 수도 있다. 모바일 디바이스는 또한 도 3 의 오디오 인코더 (20) 와 같은 오디오 인코더를 포함할 수도 있다.

[0298] 러기다이즈드 (ruggedized) 비디오 캡쳐 디바이스는 3D 음장을 리코딩하도록 더 구성될 수도 있다. 일부 예들에서, 러기다이즈드 비디오 캡쳐 디바이스는 활동에 참가하는 사용자의 헬멧에 부착될 수도 있다. 예를 들어, 러기다이즈드 비디오 캡쳐 디바이스는 사용자 급류 래프팅의 헬멧에 부착될 수도 있다. 이러한 방법으로, 러기다이즈드 비디오 캡쳐 디바이스는 사용자 주변의 모든 액션 (예컨대, 사용자 뒤에서 부서지는 물, 사용자의 전면에서 말하고 있는 다른 래프터, 등) 을 나타내는 3D 음장을 캡쳐할 수도 있다.

[0299] 이 기법들은 또한 3D 음장을 리코딩하도록 구성될 수도 있는 부속물 향상된 (accessory enhanced) 모바일 디바이스에 대해 수행될 수도 있다. 일부 예들에서, 모바일 디바이스는 하나 이상의 부속물들의 추가에 따라, 위에서 설명된 모바일 디바이스들과 유사할 수도 있다. 예를 들어, 아이겐 마이크로폰은 부속물 향상된 모바일 디바이스를 형성하기 위해 위에서 언급된 모바일 디바이스에 부착될 수도 있다. 이러한 방법으로, 부속물 향상된 모바일 디바이스는 단지 부속물 향상된 모바일 디바이스에 통합된 사운드 캡쳐 구성요소들을 이용하는 것보다 더 높은 품질 버전의 3D 음장을 캡쳐할 수도 있다.

[0300] 본 개시물에서 설명하는 기법들의 여러 양태들을 수행할 수도 있는 예시적인 오디오 플레이백 디바이스들이 아래에서 추가로 설명된다. 본 개시물의 하나 이상의 기법들에 따르면, 스피커들 및/또는 사운드 바들은 임의의 임의의 구성으로 배열될 수도 있지만 여전히 3D 음장을 플레이백할 수도 있다. 더욱이, 일부 예들에서, 헤드폰 플레이백 디바이스들은 유선 또는 무선 접속을 통해서 디코더 (24) 에 커플링될 수도 있다. 본 개시물의 하나 이상의 기법들에 따르면, 음장의 단일 포괄 표현 (generic representation) 이 스피커들, 사운드 바들, 및 헤드폰 플레이백 디바이스들의 임의의 조합 상에서 음장을 렌더링하기 위해 이용될 수도 있다.

[0301] 다수의 상이한 예시적인 오디오 플레이백 환경들이 또한 본 개시물에서 설명하는 기법들의 여러 양태들을 수행하는데 적합할 수도 있다. 예를 들어, 5.1 스피커 플레이백 환경, 2.0 (예컨대, 스테레오) 스피커 플레이백 환경, 풀 높이 전면 라우드스피커들을 가지는 9.1 스피커 플레이백 환경, 22.2 스피커 플레이백 환경, 16.0 스피커 플레이백 환경, 자동차 스피커 플레이백 환경, 및 이어 버드 플레이백 환경을 가지는 모바일 디바이스가 본 개시물에서 설명하는 기법들의 여러 양태들을 수행하는데 적합한 환경들일 수도 있다.

[0302] 본 개시물의 하나 이상의 기법들에 따르면, 음장의 단일 포괄 표현이 전술한 플레이백 환경들 중 임의의 환경 상에서 음장을 렌더링하기 위해 이용될 수도 있다. 게다가, 본 개시물의 기법들은 위에서 설명된 것과는 다른 플레이백 환경들 상에서의 플레이백을 위해 렌더러가 포괄 표현으로부터 음장을 렌더링가능하게 한다. 예를 들어, 설계 고려사항들이 7.1 스피커 플레이백 환경에 따른 스피커들의 적합한 배치를 방해하면 (예컨대,

우측 서라운드 스피커를 배치하는 것이 가능하지 않으면), 본 개시물의 기법들은 플레이백이 6.1 스피커 플레이백 환경 상에서 달성될 수 있도록 렌더가 다른 6 개의 스피커들을 보상가능하게 한다.

[0303] 더욱이, 사용자는 헤드폰들을 착용한 상태에서 스포츠 게임을 볼 수도 있다. 본 개시물의 하나 이상의 기법들에 따르면, 스포츠 게임의 3D 음장이 획득될 수 있으며 (예컨대, 하나 이상의 아이겐 마이크로폰들이 야구 경기장 내 및/또는 둘레에 배치될 수도 있으며), 3D 음장에 대응하는 HOA 계수들이 획득되어 디코더로 송신될 수도 있으며, 디코더가 HOA 계수들에 기초하여 3D 음장을 복원하여 복원된 3D 음장을 렌더러로 출력할 수도 있으며, 렌더러가 플레이백 환경의 유형 (예컨대, 헤드폰들) 에 관한 표시를 획득하여 복원된 3D 음장을 헤드폰들이 스포츠 게임의 3D 음장의 표현을 출력시키는 신호들로 렌더링할 수도 있다.

[0304] 위에서 설명된 여러 경우들의 각각에서, 오디오 인코딩 디바이스 (20) 가 방법을 수행하거나 또는 아니면 오디오 인코딩 디바이스 (20) 가 수행되도록 구성되는 방법의 각각의 단계를 수행하는 수단을 포함할 수도 있는 것으로 이해되어야 한다. 일부의 경우, 수단은 하나 이상의 프로세서들을 포함할 수도 있다. 일부의 경우, 하나 이상의 프로세서들은 비일시성 컴퓨터-판독가능 저장 매체에 저장된 명령들에 의해 구성되는 특수 목적 프로세서를 나타낼 수도 있다. 다시 말해서, 인코딩 예들의 세트들 각각에서 본 기법들의 여러 양태들은, 실행될 때, 하나 이상의 프로세서들로 하여금, 오디오 인코딩 디바이스 (20) 가 수행하도록 구성되어 있는 방법을 수행하도록 하는 명령들을 저장하고 있는 비일시성 컴퓨터-판독가능 저장 매체를 제공할 수도 있다.

[0305] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 그 기능들은 하나 이상의 명령들 또는 코드로서, 컴퓨터-판독가능 매체 상에 저장되거나 또는 컴퓨터-판독가능 매체를 통해서 송신될 수도 있으며, 하드웨어-기반의 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터-판독가능 매체는 데이터 저장 매체들과 같은, 유형의 매체에 대응하는 컴퓨터-판독가능 저장 매체들을 포함할 수도 있다. 데이터 저장 매체는 본 개시물에서 설명하는 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 취출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터-판독가능 매체를 포함할 수도 있다.

[0306] 이와 유사하게, 위에서 설명된 여러 경우들 각각에서, 오디오 디코딩 디바이스 (24) 가 방법을 수행하거나 또는 아니면 오디오 디코딩 디바이스 (24) 가 수행하도록 구성되는 방법의 각각의 단계를 수행하는 수단을 포함할 수도 있는 것으로 이해되어야 한다. 일부의 경우, 수단은 하나 이상의 프로세서들을 포함할 수도 있다. 일부의 경우, 하나 이상의 프로세서들은 비일시성 컴퓨터-판독가능 저장 매체에 저장된 명령들에 의해 구성되는 특수 목적 프로세서를 나타낼 수도 있다. 다시 말해서, 인코딩 예들의 세트들의 각각에서 본 기법들의 여러 양태들은 실행될 때, 하나 이상의 프로세서들로 하여금, 오디오 디코딩 디바이스 (24) 가 수행하도록 구성되어 있는 방법을 수행가능하게 하는 명령들을 저장하고 있는 비일시성 컴퓨터-판독가능 저장 매체를 제공할 수도 있다.

[0307] 일 예로서, 이에 한정하지 않고, 이런 컴퓨터-판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광디스크 스토리지, 자기디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 그러나, 컴퓨터-판독가능 저장 매체 및 데이터 저장 매체는 접속부들, 캐리어 파들, 신호들, 또는 다른 일시성 매체를 포함하지 않고, 그 대신, 비-일시성 유형의 저장 매체로 송신되는 것으로 해석되어야 한다. 디스크 (disk) 및 디스크 (disc) 는, 본원에서 사용할 때, 컴팩트 디스크 (CD), 레이저 디스크, 광 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 Blu-ray 디스크를 포함하며, 디스크들 (disks) 은 데이터를 자기적으로 보통 재생하지만, 디스크들 (discs) 은 레이저로 데이터를 광학적으로 재생한다. 앞에서 언급한 것들의 결합들이 또한 컴퓨터-판독가능 매체들의 범위 내에 포함되어야 한다.

[0308] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 등가의 집적 또는 이산 로직 회로와 같은, 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 용어 "프로세서" 는, 본원에서 사용될 때 전술한 구조 중 임의의 구조 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능은 인코딩 및 디코딩을 위해 구성되는 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공되거나, 또는 결합된 코덱에 포함될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들로 전적으로 구현될 수 있다.

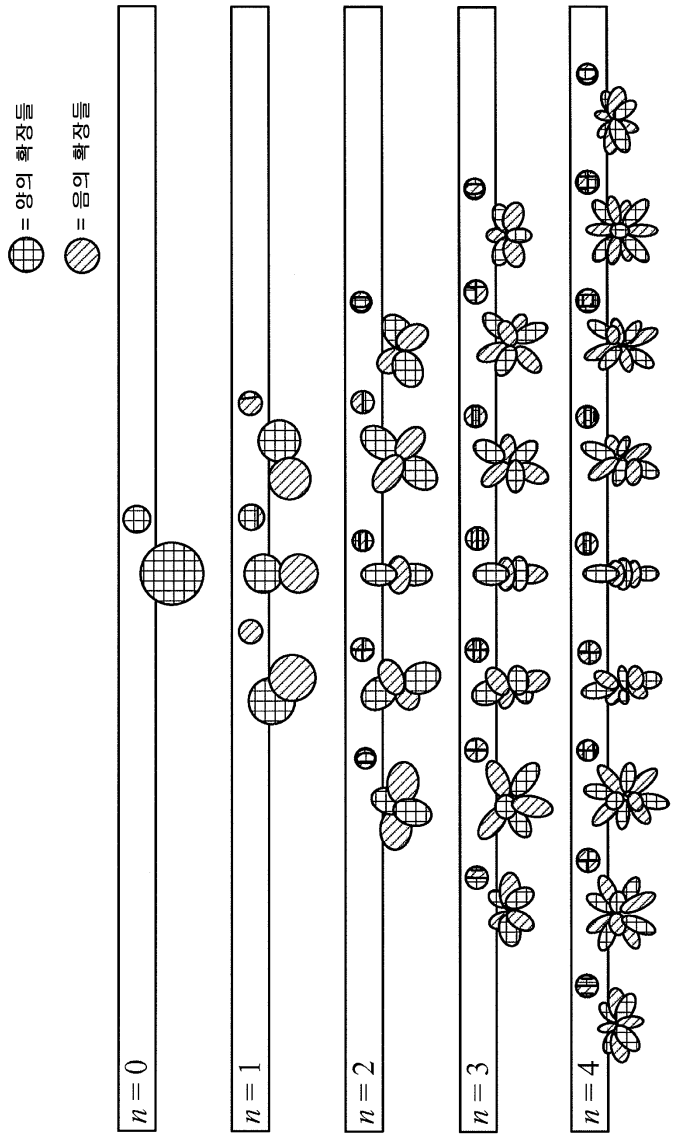
[0309] 본 개시물의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예컨대, 칩 세트) 를 포함한, 매우 다양

한 디바이스들 또는 장치들로 구현될 수도 있다. 개시한 기법들을 수행하도록 구성되는 디바이스들의 기능적 양태들을 강조하기 위해서 여러 구성요소들, 모듈들, 또는 유닛들이 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다. 대신, 위에서 설명한 바와 같이, 여러 유닛들이 코덱 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명한 바와 같은 하나 이상의 프로세서들을 포함한, 상호작용하는 하드웨어 유닛들의 컬렉션으로 제공될 수도 있다.

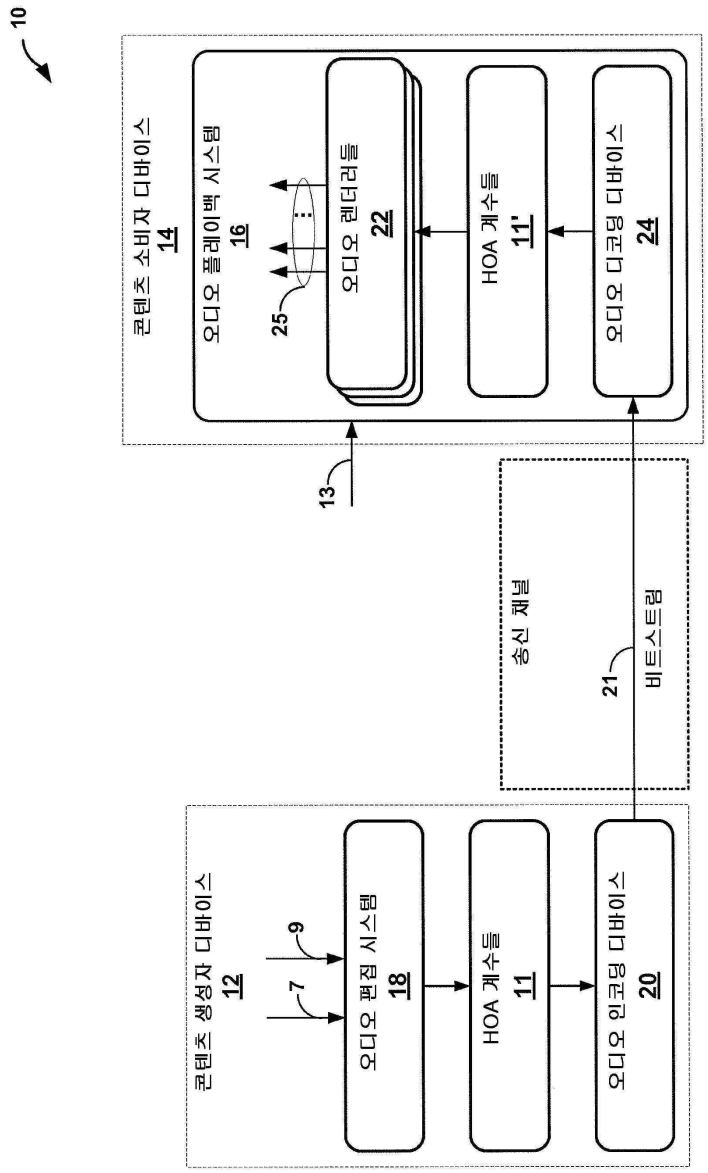
[0310] 본 기법들의 여러 양태들이 설명되었다. 본 기법들의 이들 및 다른 실시형태들은 다음 청구항들의 범위 이내이다.

도면

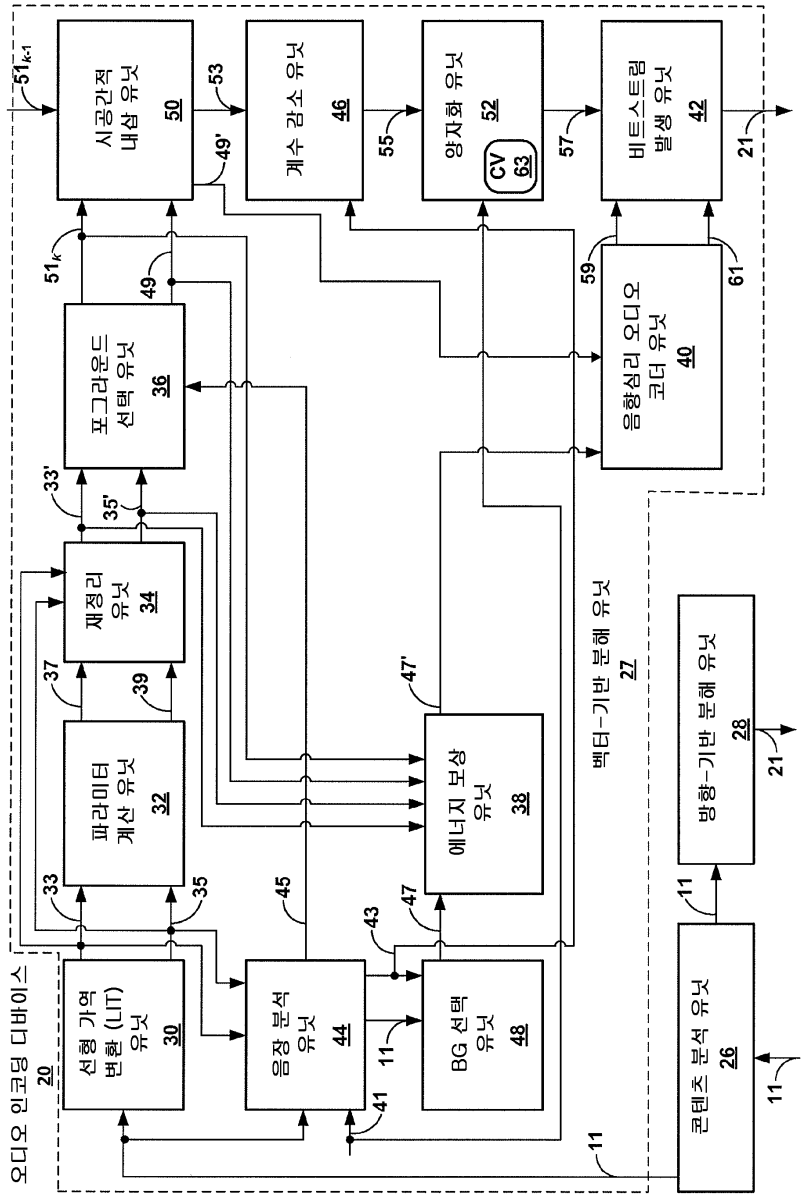
도면1



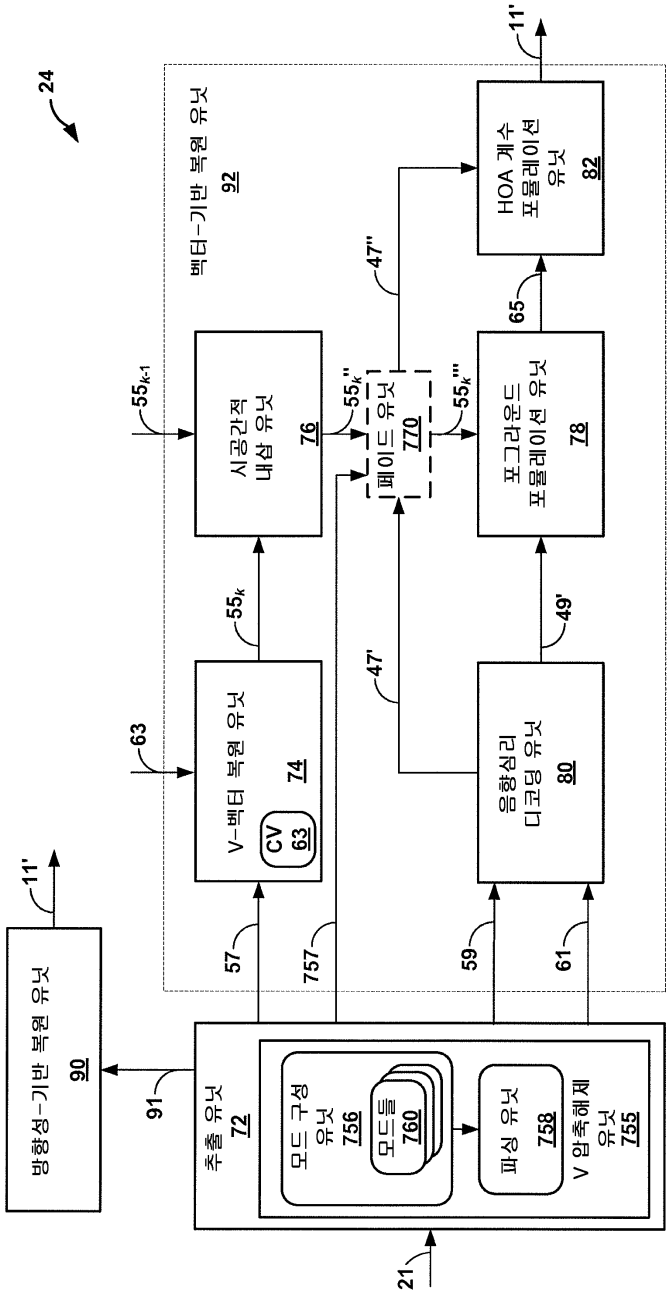
도면2


10
콘텐츠 소비자 디바이스
14
오디오 플레이백 시스템
16
25
오디오 렌더러들
22
HOA 계수들
11'
오디오 디코딩 디바이스
24
13
송신 채널
21
비트스트림
콘텐츠 생성자 디바이스
12
9
7
오디오 편집 시스템
18
HOA 계수들
11
오디오 인코딩 디바이스
20

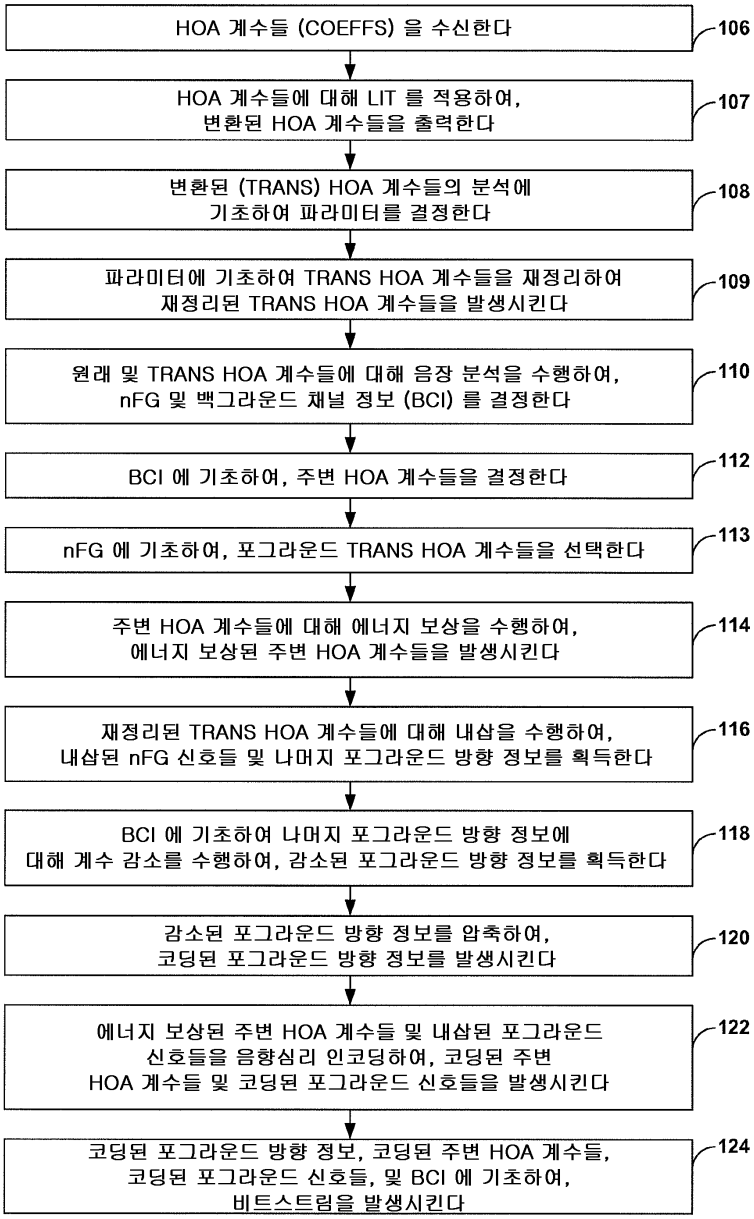
도면3


오디오 인코딩 디바이스
20
선형 가역 변환 (LIT) 유닛
30
파라미터 계산 유닛
32
재정리 유닛
34
포그라운드 선택 유닛
36
시공간적 내삽 유닛
50
음장 분석 유닛
44
BG 선택 유닛
48
에너지 보상 유닛
38
계수 감소 유닛
46
양자화 유닛
52
CV
63
비트스트림 발생 유닛
42
음향심리 오디오 코더 유닛
40
벡터-기반 분해 유닛
27
콘텐츠 분석 유닛
26
방향-기반 분해 유닛
28

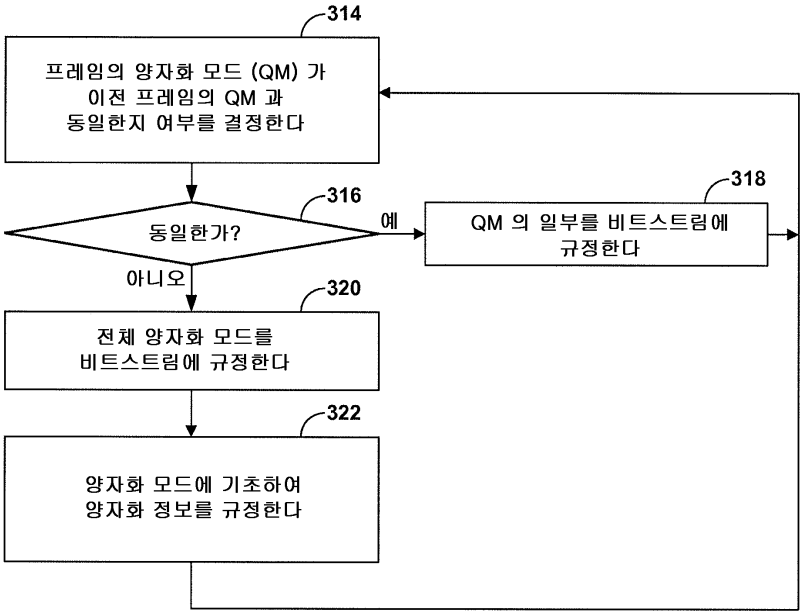
도면4


24
11'
방향성-기반 복원 유닛
90
91
추출 유닛
72
모드 구성 유닛
756
모드들
760
파싱 유닛
758
V 압축해제 유닛
755
21
57
757
59
61
63
V-벡터 복원 유닛
74
CV
63
55k
55k-1
시공간적 내삽 유닛
76
55k"
47'
음향심리 디코딩 유닛
80
49'
47"
페이드 유닛
770
55k'"
포그라운드 포뮬레이션 유닛
78
65
HOA 계수 포뮬레이션 유닛
82
11'
벡터-기반 복원 유닛
92

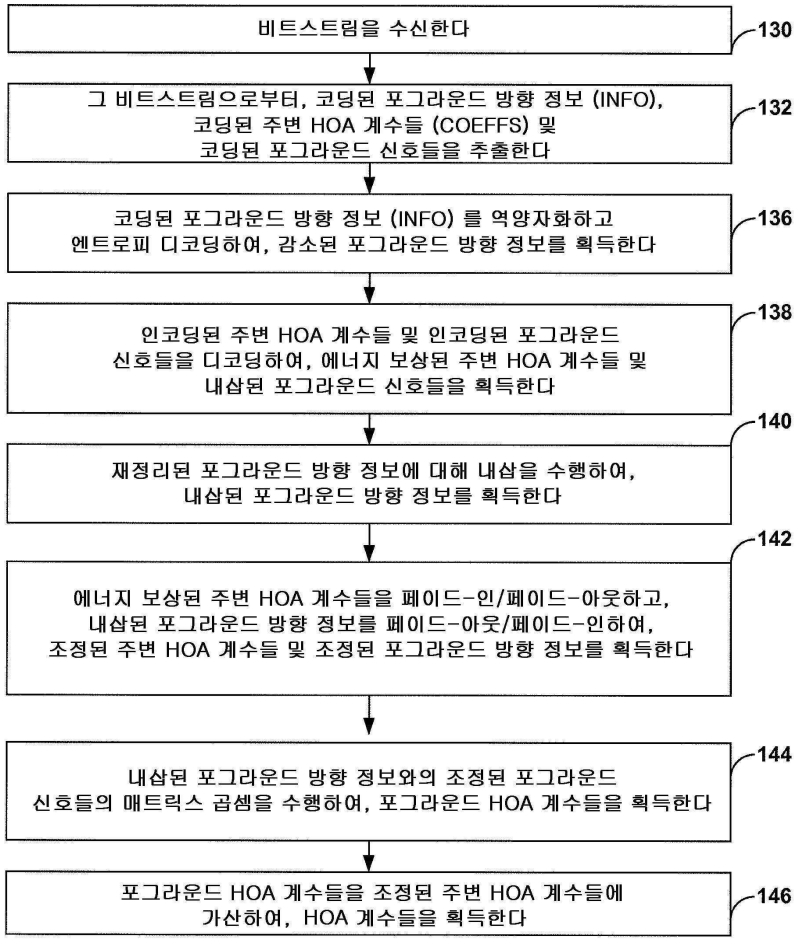
도면5a


HOA 계수들 (COEFFS) 을 수신한다
106
HOA 계수들에 대해 LIT 를 적용하여,
변환된 HOA 계수들을 출력한다
107
변환된 (TRANS) HOA 계수들의 분석에
기초하여 파라미터를 결정한다
108
파라미터에 기초하여 TRANS HOA 계수들을 재정리하여
재정리된 TRANS HOA 계수들을 발생시킨다
109
원래 및 TRANS HOA 계수들에 대해 음장 분석을 수행하여,
nFG 및 백그라운드 채널 정보 (BCI) 를 결정한다
110
BCI 에 기초하여, 주변 HOA 계수들을 결정한다
112
nFG 에 기초하여, 포그라운드 TRANS HOA 계수들을 선택한다
113
주변 HOA 계수들에 대해 에너지 보상을 수행하여,
에너지 보상된 주변 HOA 계수들을 발생시킨다
114
재정리된 TRANS HOA 계수들에 대해 내삽을 수행하여,
내삽된 nFG 신호들 및 나머지 포그라운드 방향 정보를 획득한다
116
BCI 에 기초하여 나머지 포그라운드 방향 정보에
대해 계수 감소를 수행하여, 감소된 포그라운드 방향 정보를 획득한다
118
감소된 포그라운드 방향 정보를 압축하여,
코딩된 포그라운드 방향 정보를 발생시킨다
120
에너지 보상된 주변 HOA 계수들 및 내삽된 포그라운드
신호들을 음향심리 인코딩하여, 코딩된 주변
HOA 계수들 및 코딩된 포그라운드 신호들을 발생시킨다
122
코딩된 포그라운드 방향 정보, 코딩된 주변 HOA 계수들,
코딩된 포그라운드 신호들, 및 BCI 에 기초하여,
비트스트림을 발생시킨다
124

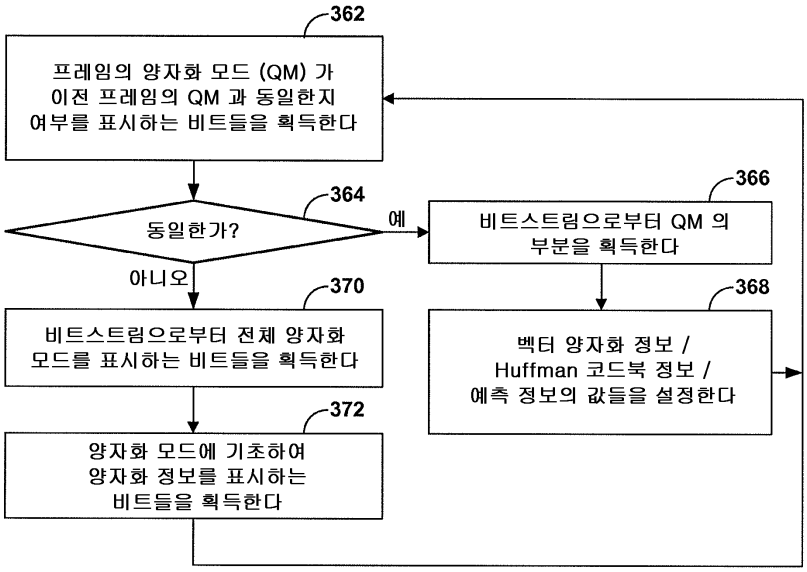
도면5b


314
프레임의 양자화 모드 (QM) 가
이전 프레임의 QM 과
동일한지 여부를 결정한다
316
동일한가?
예
318
QM 의 일부를 비트스트림에
규정한다
아니오
320
전체 양자화 모드를
비트스트림에 규정한다
322
양자화 모드에 기초하여
양자화 정보를 규정한다

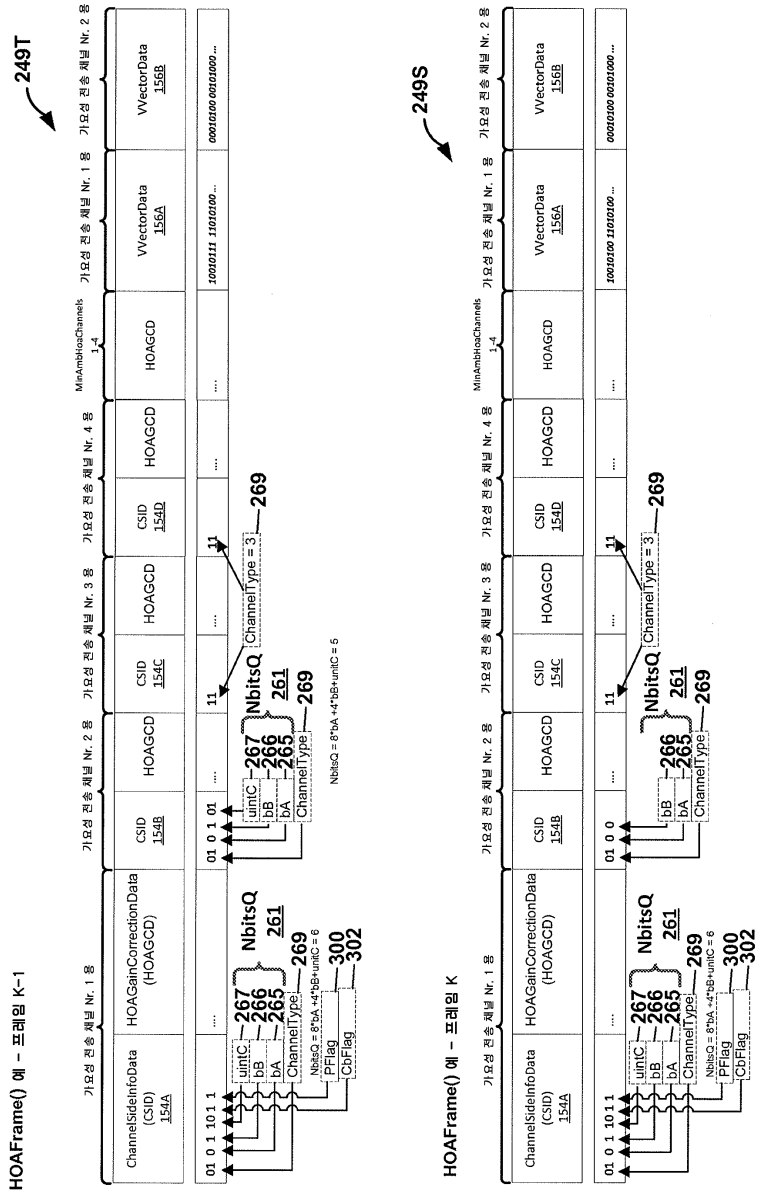
도면6a


비트스트림을 수신한다
130
그 비트스트림으로부터, 코딩된 포그라운드 방향 정보 (INFO),
코딩된 주변 HOA 계수들 (COEFFS) 및
코딩된 포그라운드 신호들을 추출한다
132
코딩된 포그라운드 방향 정보 (INFO) 를 역양자화하고
엔트로피 디코딩하여, 감소된 포그라운드 방향 정보를 획득한다
136
인코딩된 주변 HOA 계수들 및 인코딩된 포그라운드
신호들을 디코딩하여, 에너지 보상된 주변 HOA 계수들 및
내삽된 포그라운드 신호들을 획득한다
138
재정리된 포그라운드 방향 정보에 대해 내삽을 수행하여,
내삽된 포그라운드 방향 정보를 획득한다
140
에너지 보상된 주변 HOA 계수들을 페이드-인/페이드-아웃하고,
내삽된 포그라운드 방향 정보를 페이드-아웃/페이드-인하여,
조정된 주변 HOA 계수들 및 조정된 포그라운드 방향 정보를 획득한다
142
내삽된 포그라운드 방향 정보와의 조정된 포그라운드
신호들의 매트릭스 곱셈을 수행하여, 포그라운드 HOA 계수들을 획득한다
144
포그라운드 HOA 계수들을 조정된 주변 HOA 계수들에
가산하여, HOA 계수들을 획득한다
146

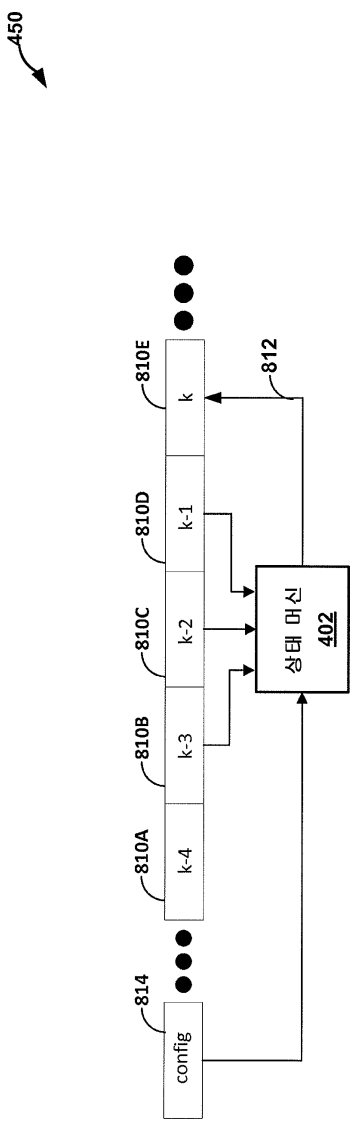
도면6b


362
프레임의 양자화 모드 (QM) 가
이전 프레임의 QM 과 동일한지
여부를 표시하는 비트들을 획득한다
364
동일한가?
예
366
비트스트림으로부터 QM 의
부분을 획득한다
368
벡터 양자화 정보 /
Huffman 코드북 정보 /
예측 정보의 값들을 설정한다
아니오
370
비트스트림으로부터 전체 양자화
모드를 표시하는 비트들을 획득한다
372
양자화 모드에 기초하여
양자화 정보를 표시하는
비트들을 획득한다

도면7


249T
HOAFrame() 예 – 프레임 K-1
가요성 전송 채널 Nr. 1 용
가요성 전송 채널 Nr. 2 용
가요성 전송 채널 Nr. 3 용
가요성 전송 채널 Nr. 4 용
MinAmbHoaChannels 1-4
가요성 전송 채널 Nr. 1 용
가요성 전송 채널 Nr. 2 용
ChannelSideInfoData (CSID) 154A
HOAGainCorrectionData (HOAGCD)
CSID 154B
HOAGCD
CSID 154C
HOAGCD
CSID 154D
HOAGCD
HOAGCD
VVectorData 156A
VVectorData 156B
01 0 1 10 1 1
....
01 0 1 01
....
11
....
11
....
....
10010111 11010100...
00010100 00101000...
uintC
bB
bA
ChannelType
NbitsQ = 8*bA +4*bB+uintC + 6
PFlag
CbFlag
267
266
265
269
261
300
302
NbitsQ
ChannelType = 3
NbitsQ = 8*bA +4*bB+uintC + 5
249S
HOAFrame() 예 – 프레임 K
01 0 1 10 1 1
....
01 0 0
....
11
....
11
....
....
10010100 11010100...
00010100 00101000...

도면8


450
810E
k
812
810D
k-1
상태 머신
402
810C
k-2
810B
k-3
810A
k-4
814
config