



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2025년05월09일
 (11) 등록번호 10-2804193
 (24) 등록일자 2025년04월30일

(51) 국제특허분류(Int. Cl.)
 H04N 19/70 (2014.01) H04N 19/119 (2014.01)
 H04N 19/122 (2014.01) H04N 19/186 (2014.01)
 H04N 19/423 (2014.01) H04N 19/96 (2014.01)
 (52) CPC특허분류
 H04N 19/70 (2015.01)
 H04N 19/119 (2015.01)
 (21) 출원번호 10-2021-7035633
 (22) 출원일자(국제) 2021년02월22일
 심사청구일자 2021년11월01일
 (85) 번역문제출일자 2021년11월01일
 (65) 공개번호 10-2021-0141745
 (43) 공개일자 2021년11월23일
 (86) 국제출원번호 PCT/US2021/019020
 (87) 국제공개번호 WO 2021/202012
 국제공개일자 2021년10월07일
 (30) 우선권주장
 63/003,123 2020년03월31일 미국(US)
 17/081,392 2020년10월27일 미국(US)
 (56) 선행기술조사문헌
 Benjamin Bross, et al., Versatile Video
 Coding (Draft 8), JVET-Q2001-vE, 2020.03.12.*
 (뒷면에 계속)

(73) 특허권자
텐센트 아메리카 엘엘씨
 미국 94306 캘리포니아주 팔로 알토 파크 블러바
 드 2747
 (72) 발명자
최, 병두
 미국 94306 캘리포니아주 팔로 알토 파크 블러바
 드 2747 텐센트 아메리카 엘엘씨 내
류, 산
 미국 94306 캘리포니아주 팔로 알토 파크 블러바
 드 2747 텐센트 아메리카 엘엘씨 내
헝거, 스테판
 미국 94306 캘리포니아주 팔로 알토 파크 블러바
 드 2747 텐센트 아메리카 엘엘씨 내
 (74) 대리인
양영준, 김연송, 백만기

전체 청구항 수 : 총 10 항

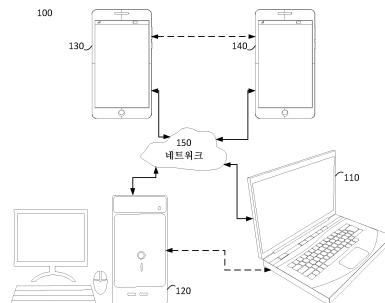
심사관 : 이남숙

(54) 발명의 명칭 코딩된 비디오 스트림에서 서브픽처 파티셔닝을 시그널링하기 위한 방법

(57) 요약

현재 픽처의 적어도 하나의 서브픽처의 수가 1보다 큰 것에 기초하여, 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는지를 표시하는 플래그를 획득하는 것; 플래그가 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는 것을 표시하는 것에 기초하여, 서브픽처 파티셔닝 정보를 획득하는 것; 및 서브픽처 파티셔닝 정보에 기초하여 현재 픽처를 디코딩하는 것을 포함하는, 적어도 하나의 프로세서를 사용하여 인코딩된 비디오 비트스트림을 디코딩하기 위한 방법, 디바이스, 및 비일시적 컴퓨터 판독가능 매체.

대표도



(52) CPC특허분류

HO4N 19/122 (2015.01)
HO4N 19/186 (2015.01)
HO4N 19/423 (2015.01)
HO4N 19/96 (2015.01)

(56) 선행기술조사문헌

iska M. Hamuksela, et al., AHG12: Sub-picture-based picture partitioning and decoding, JVET-N0046, 2019.03.13.
Byeongdoo Choi, et al., AHG12: On sub-picture partitioning, JVET-N0055-v2, 2019.03.20.
Byeongdoo Choi, et al., AHG8: On spatial scalability support with reference picture resampling, JVET-00333, 2019.06.26.
Byeongdoo Choi, et al., AHG8/AHG12: On sub-picture partitioning support with layers, JVET-00334, 2019.06.27.
Lulin Chen, et al., AHG17/AHG12: On associating slices with a subpicture, JVET-P0144-v1, 2019.09.24.
Nael Ouedraogo, et al., [AHG12] On Subpictures merging for VVC, JVET-P0351-v2, 2019.10.01.
Yao-Jen Chang, et al., AhG12: On the slice number signaling dependent on subpictures, JVET-Q0332r2, 2020.01.09.

*는 심사관에 의하여 인용된 문헌

명세서

청구범위

청구항 1

적어도 하나의 프로세서를 사용하여 인코딩된 비디오 비트스트림을 디코딩하는 방법으로서,

현재 픽처의 적어도 하나의 서브픽처의 수를 결정하는 단계 - 상기 결정하는 단계는,

상기 인코딩된 비트스트림으로부터, 현재 픽처의 픽처 폭(pic_width_max_in_luma_samples) 및 픽처 높이(pic_height_max_in_luma_samples)를 획득하는 단계; 및

상기 픽처 폭이 상기 현재 픽처의 루마 성분(CtbSizeY)의 코딩 트리 단위의 크기보다 크지 않고 상기 픽처 높이가 상기 루마 성분의 코딩 트리 단위의 크기보다 크지 않은 경우, 적어도 하나의 서브픽처의 수를 1과 동일하도록 결정하는 단계

를 포함함 -;

상기 현재 픽처의 상기 적어도 하나의 서브픽처의 수가 1보다 큰 것에 기초하여, 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는지를 표시하는 플래그(subpic_info_present_flag)를 결정하는 단계; 및

상기 플래그가 상기 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는 것을 표시하는 것에 기초하여,

상기 서브픽처 파티셔닝 정보를 획득하는 단계; 및

상기 서브픽처 파티셔닝 정보에 기초하여 상기 현재 픽처를 디코딩하는 단계

를 포함하고,

상기 방법은,

상기 현재 픽처가 코딩된 계층 비디오 시퀀스(Coded Layer Video Sequence, CLVS)의 제1 픽처가 아닐 때, 상기 현재 픽처의 각각의 서브픽처에 대해, 상기 현재 픽처의 SubpicIdVal[i]의 값이 동일한 계층에서의 디코딩 순서로 이전 픽처의 SubpicIdVal[i]의 값과 동일하지 않으면, 상기 현재 픽처의 상기 서브픽처의 sps_independent_subpics_flag가 1과 동일하거나 또는 상기 현재 픽처의 상기 서브픽처의 subpic_treated_as_pic_flag[i] 및 loop_filter_across_subpic_enabled_flag[i]가 1과 동일한 것으로 결정하는 단계

를 더 포함하는, 방법.

청구항 2

제1항에 있어서, 상기 플래그 및 상기 서브픽처 파티셔닝 정보는 시퀀스 파라미터 세트(SPS)에서 시그널링되는, 방법.

청구항 3

제2항에 있어서, 상기 적어도 하나의 서브픽처의 수는 상기 SPS에서 시그널링되는, 방법.

청구항 4

삭제

청구항 5

제1항에 있어서, 상기 적어도 하나의 서브픽처의 수가 1인 것에 기초하여, 상기 플래그는 상기 서브픽처 파티셔닝 정보가 명시적으로 시그널링되지 않는 것을 표시하는, 방법.

청구항 6

제1항에 있어서, 상기 서브픽처 파티셔닝 정보는 상기 적어도 하나의 서브픽처 중에서의 서브픽처의 크기를 표

시하는, 방법.

청구항 7

제6항에 있어서, 상기 서브픽처의 크기는 상기 서브픽처의 높이 및 상기 서브픽처의 폭을 포함하는, 방법.

청구항 8

제6항에 있어서, 상기 서브픽처의 크기는 상기 현재 픽처의 루마 성분의 코딩 트리 유닛의 크기의 단위로 상기 서브픽처 파티셔닝 정보에 표시되는, 방법.

청구항 9

제6항에 있어서, 다른 서브픽처의 크기는 상기 서브픽처 파티셔닝 정보에 의해 표시되는 상기 서브픽처의 크기에 기초하여 결정되는, 방법.

청구항 10

인코딩된 비디오 비트스트림을 디코딩하기 위한 디바이스로서,

프로그램 코드를 저장하도록 구성되는 적어도 하나의 메모리; 및

제1항 내지 제3항 및 제5항 내지 제9항 중 어느 한 항의 방법을 수행하기 위해 상기 프로그램 코드를 판독하고 상기 프로그램 코드에 의해 명령된 바와 같이 동작하도록 구성되는 적어도 하나의 프로세서

를 포함하는, 디바이스.

청구항 11

명령어들이 저장된 비일시적 컴퓨터 판독가능 매체로서,

상기 명령어들은: 인코딩된 비디오 비트스트림을 디코딩하기 위한 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 상기 하나 이상의 프로세서로 하여금 제1항 내지 제3항 및 제5항 내지 제9항 중 어느 한 항의 방법을 수행하게 하는 하나 이상의 명령어를 포함하는, 비일시적 컴퓨터 판독가능 매체.

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

발명의 설명

기술 분야

- [0001] 관련 출원에 대한 교차 참조
- [0002] 본 출원은 2020년 3월 31일자로 출원된 미국 가특허 출원 제63/003,123호 및 2020년 10월 27일자로 출원된 미국 특허 출원 제17/081,392호로부터 우선권을 주장하며, 그 전문이 본 명세서에 포함된다.
- [0003] 분야
- [0004] 개시된 주제는 비디오 코딩 및 디코딩에 관한 것으로, 더 구체적으로는, 코딩된 비디오 스트림에서 서브픽처 파티셔닝(subpicture partitioning)을 시그널링하는 것에 관한 것이다.

배경 기술

- [0005] 모션 보상(motion compensation)을 동반한 인터-픽처 예측(inter-picture prediction)을 사용하는 비디오 코딩 및 디코딩이 알려져 왔다. 압축되지 않은 디지털 비디오는 일련의 픽처들로 구성될 수 있고, 각각의 픽처는, 예를 들어, 1920x1080 루미넌스 샘플들 및 연관된 크로미넌스 샘플들의 공간적 차원(spatial dimension)을 갖는다. 이 일련의 픽처들은, 예를 들어, 초당 60 픽처 또는 60Hz의, 고정 또는 가변 픽처 레이트(비공식적으로 프레임 레이트로도 알려져 있음)를 가질 수 있다. 압축되지 않은 비디오는 상당한 비트레이트 요건들을 갖는다. 예를 들어, 샘플당 8 비트에서의 1080p60 4:2:0 비디오(60Hz 프레임 레이트에서의 1920x1080 루미넌스 샘플 해상도)는 1.5 gigabits/s에 근접한 대역폭을 요구한다. 1 시간 분량의 이러한 비디오는 600 기가바이트(GB)를 초과하는 저장 공간을 필요로 한다.
- [0006] 비디오 코딩 및 디코딩의 한 가지 목적은, 압축을 통한, 입력 비디오 신호에서의 중복성(redundancy)의 감소일 수 있다. 압축은 앞서 설명한 대역폭 또는 저장 공간 요건들을, 일부 경우들에서는, 2 자릿수 이상 감소시키는데 도움이 될 수 있다. 무손실 및 손실 압축 양자 모두뿐만 아니라 이들의 조합이 이용될 수 있다. 무손실 압축은 압축된 원래 신호로부터 원래 신호의 정확한 사본이 재구성될 수 있는 기법들을 지칭한다. 손실 압축을 사용할 때, 재구성된 신호는 원래 신호와 동일하지 않을 수 있지만, 원래 신호와 재구성된 신호 사이의 왜곡은 재구성된 신호가 의도된 응용에 유용할 정도로 충분히 작다. 비디오의 경우, 손실 압축이 널리 이용된다. 용인되는 왜곡의 양은 응용에 의존하며; 예를 들어, 특정 소비자 스트리밍 응용들의 사용자들은 텔레비전 기여 응용들(television contribution applications)의 사용자들보다 더 높은 왜곡을 용인할 수 있다. 달성가능한 압축비는 더 높은 허용가능한/용인가능한 왜곡이 더 높은 압축비를 산출할 수 있다는 사실을 반영할 수 있다.
- [0007] 비디오 인코더 및 디코더는, 예를 들어, 모션 보상, 변환, 양자화, 및 엔트로피 코딩을 포함한, 몇가지 광범위한 카테고리로부터의 기법들을 이용할 수 있으며, 이들 중 일부는 아래에서 소개될 것이다.
- [0008] 역사적으로, 비디오 인코더들 및 디코더들은, 대부분의 경우에, 코딩된 비디오 시퀀스(coded video sequence, CVS), 픽처 그룹(Group of Pictures, GOP), 또는 유사한 멀티-픽처 타임프레임(multi-picture timeframe)에 대해 정의되고 일정하게 유지된 주어진 픽처 크기에서 동작하는 경향이 있었다. 예를 들어, MPEG-2에서, 시스템 설계들은 장면의 액티비티와 같은 팩터들에 따라, 그러나 I 픽처들에서만, 따라서 통상적으로 GOP에 대해 수평 해상도(및 그에 따라 픽처 크기)를 변경하는 것으로 알려져 있다. CVS 내의 상이한 해상도의 사용을 위한 참조 픽처들의 리샘플링은, 예를 들어, ITU-T Rec. H.263 Annex P로부터 알려져 있다. 그러나, 여기서 픽처 크기는 변경되지 않고, 참조 픽처들만이 리샘플링되므로, 잠재적으로 픽처 캔버스의 부분들만이 사용되거나(다운샘플링의 경우), 또는 장면의 부분들만이 캡처된다(업샘플링의 경우). 또한, H.263 Annex Q는(각각의 차원에서) 2배 만큼 상향 또는 하향으로 개별 매크로블록의 리샘플링을 허용한다. 다시, 픽처 크기는 동일하게 유지된다. 매크로블록의 크기는 H.263에서 고정되므로, 시그널링될 필요가 없다.
- [0009] 예측된 픽처들에서의 픽처 크기의 변경들은 현대의 비디오 코딩에서 더 주류가 되었다. 예를 들어, VP9는 참조 픽처 리샘플링 및 전체 픽처에 대한 해상도 변경을 허용한다. 유사하게, VVC에 대해 이루어진 특정 제안들(예

를 들어, 그 전체가 본 명세서에 포함된 [Hendry, et. al, "On adaptive resolution change (ARC) for VVC", Joint Video Team document JVET-M0135-v1, Jan 9-19, 2019]를 포함함)은 전체 참조 픽처들의 상이한 - 더 높거나 더 낮은 - 해상도로의 리샘플링을 허용한다. 그 문헌에서는, 상이한 후보 해상도들이 시퀀스 파라미터 세트에서 코딩되고 픽처 파라미터 세트에서 픽처별 선택스 요소들(per-picture syntax elements)에 의해 참조되도록 제안된다.

발명의 내용

[0010] 일 실시예에서, 적어도 하나의 프로세서를 사용하여 인코딩된 비디오 비트스트림을 디코딩하는 방법이 제공되며, 이 방법은, 현재 픽처의 적어도 하나의 서브픽처의 수가 1보다 큰 것에 기초하여, 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는지를 표시하는 플래그를 획득하는 단계; 플래그가 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는 것을 표시하는 것에 기초하여, 서브픽처 파티셔닝 정보를 획득하는 단계; 및 서브픽처 파티셔닝 정보에 기초하여 현재 픽처를 디코딩하는 단계를 포함한다.

[0011] 일 실시예에서, 인코딩된 비디오 비트스트림을 디코딩하기 위한 디바이스가 제공되며, 이 디바이스는, 프로그램 코드를 저장하도록 구성되는 적어도 하나의 메모리; 및 프로그램 코드를 관독하고 프로그램 코드에 의해 명령된 바와 같이 동작하도록 구성되는 적어도 하나의 프로세서를 포함하고, 프로그램 코드는: 적어도 하나의 프로세서로 하여금, 현재 픽처의 적어도 하나의 서브픽처의 수가 1보다 큰 것에 기초하여, 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는지를 표시하는 플래그를 획득하게 하도록 구성되는 제1 획득 코드; 적어도 하나의 프로세서로 하여금, 플래그가 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는 것을 표시하는 것에 기초하여, 서브픽처 파티셔닝 정보를 획득하게 하도록 구성되는 제2 획득 코드; 및 적어도 하나의 프로세서로 하여금 서브픽처 파티셔닝 정보에 기초하여 현재 픽처를 디코딩하게 하도록 구성되는 디코딩 코드를 포함한다.

[0012] 일 실시예에서, 명령어들이 저장된 비일시적 컴퓨터 관독가능 매체가 제공되며, 이 명령어들은: 인코딩된 비디오 비트스트림을 디코딩하기 위한 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 하나 이상의 프로세서로 하여금: 현재 픽처의 적어도 하나의 서브픽처의 수가 1보다 큰 것에 기초하여, 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는지를 표시하는 플래그를 획득하게 하고; 플래그가 서브픽처 파티셔닝 정보가 명시적으로 시그널링되는 것을 표시하는 것에 기초하여, 서브픽처 파티셔닝 정보를 획득하게 하고; 서브픽처 파티셔닝 정보에 기초하여 현재 픽처를 디코딩하게 하는 하나 이상의 명령어를 포함한다.

도면의 간단한 설명

[0013] 개시된 주제의 추가의 특징들, 본질 및 다양한 이점들이 다음의 상세한 설명 및 첨부 도면들로부터 더 명백할 것이다.

- 도 1은 일 실시예에 따른 통신 시스템의 단순화된 블록도의 개략도이다.
- 도 2는 일 실시예에 따른 통신 시스템의 단순화된 블록도의 개략도이다.
- 도 3은 일 실시예에 따른 디코더의 단순화된 블록도의 개략도이다.
- 도 4는 일 실시예에 따른 인코더의 단순화된 블록도의 개략도이다.
- 도 5a 내지 도 5e는 일 실시예에 따른 ARC 파라미터들을 시그널링하기 위한 옵션들의 개략도들이다.
- 도 6a 및 도 6b는 일 실시예에 따른 선택스 테이블들의 예들의 개략도이다.
- 도 7은 일 실시예에 따른 적응적 해상도 변경(adaptive resolution change)에 대한 스케일러빌리티(scalability)를 위한 예측 구조의 예이다.
- 도 8은 일 실시예에 따른 선택스 테이블의 예이다.
- 도 9는 일 실시예에 따른, 액세스 유닛당 POC 사이클 및 액세스 유닛 카운트 값을 파싱 및 디코딩하는 단순화된 블록도의 개략도이다.
- 도 10은 일 실시예에 따른, 다중-계층형 서브-픽처들(multi-layered sub-pictures)을 포함하는 비디오 비트스트림 구조의 개략도이다.
- 도 11은 일 실시예에 따른, 향상된 해상도를 갖는 선택된 서브-픽처의 디스플레이의 개략도이다.

도 12는 일 실시예에 따른, 다중-계층형 서브-픽처들을 포함하는 비디오 비트스트림에 대한 디코딩 및 디스플레이 프로세스의 블록도이다.

도 13은 일 실시예에 따른, 서브-픽처의 향상 계층(enhancement layer)을 갖는 360 비디오 디스플레이의 개략도이다.

도 14는 일 실시예에 따른, 서브-픽처들의 레이아웃 정보 및 그의 대응하는 계층 및 픽처 예측 구조의 예이다.

도 15는 일 실시예에 따른, 로컬 영역의 공간적 스케일러빌리티 모달리티(spatial scalability modality)를 갖는, 서브-픽처들의 레이아웃 정보 및 그의 대응하는 계층 및 픽처 예측 구조의 예이다.

도 16a 및 도 16b는 실시예들에 따른, 서브-픽처 레이아웃 정보에 대한 선택스 테이블들의 예들이다.

도 17은 일 실시예에 따른, 서브-픽처 레이아웃 정보에 대한 SEI 메시지의 선택스 테이블의 예이다.

도 18은 일 실시예에 따른, 각각의 출력 계층 세트에 대한 출력 계층들 및 프로파일/티어/레벨 정보를 표시하는 선택스 테이블의 예이다.

도 19는 일 실시예에 따른, 각각의 출력 계층 세트에 대한 출력 계층 모드를 표시하는 선택스 테이블의 예이다.

도 20은 일 실시예에 따른, 각각의 출력 계층 세트에 대한 각각의 계층의 현재 서브픽처를 표시하는 선택스 테이블의 예이다.

도 21은 일 실시예에 따른, 서브픽처 식별자를 표시하는 선택스 테이블의 예이다.

도 22는 일 실시예에 따른, 서브픽처 파티셔닝 정보를 표시하는 선택스 테이블의 예이다.

도 23은 일 실시예에 따른 인코딩된 비디오 비트스트림을 디코딩하기 위한 예시적인 프로세스의 흐름도이다.

도 24는 일 실시예에 따른 컴퓨터 시스템의 개략도이다.

발명을 실시하기 위한 구체적인 내용

[0014] 도 1은 본 개시내용의 일 실시예에 따른 통신 시스템(100)의 단순화된 블록도를 예시한다. 시스템(100)은 네트워크(150)를 통해 상호접속되는 적어도 2개의 단말(110-120)을 포함할 수 있다. 데이터의 단방향 송신을 위해, 제1 단말(110)은 네트워크(150)를 통해 다른 단말(120)로 송신하기 위해 로컬 위치에서 비디오 데이터를 코딩할 수 있다. 제2 단말(120)은 네트워크(150)로부터 다른 단말의 코딩된 비디오 데이터를 수신하고, 코딩된 데이터를 디코딩하고 복구된 비디오 데이터를 디스플레이할 수 있다. 단방향 데이터 송신은 미디어 서빙 응용들(media serving applications) 등에서 일반적일 수 있다.

[0015] 도 1은, 예를 들어, 영상 회의 동안 발생할 수 있는 코딩된 비디오의 양방향 송신을 지원하기 위해 제공되는 제 2 쌍의 단말들(130, 140)을 예시한다. 데이터의 양방향 송신을 위해, 각각의 단말(130, 140)은 네트워크(150)를 통해 다른 단말로 송신하기 위해 로컬 위치에서 캡처된 비디오 데이터를 코딩할 수 있다. 각각의 단말(130, 140)은 또한 다른 단말에 의해 송신된 코딩된 비디오 데이터를 수신할 수 있고, 코딩된 데이터를 디코딩할 수 있고, 복구된 비디오 데이터를 로컬 디스플레이 디바이스에서 디스플레이할 수 있다.

[0016] 도 1에서, 단말들(110-140)은 서버들, 개인용 컴퓨터들 및 스마트폰들로서 예시될 수 있지만, 본 개시내용의 원리들은 그렇게 제한되지 않을 수 있다. 본 개시내용의 실시예들은 랩톱 컴퓨터들, 태블릿 컴퓨터들, 미디어 플레이어들 및/또는 전용 영상 회의 장비에서 응용된다. 네트워크(150)는, 예를 들어, 유선 및/또는 무선 통신 네트워크들을 포함하여, 단말들(110-140) 사이에 코딩된 비디오 데이터를 전달하는 임의의 수의 네트워크들을 나타낸다. 통신 네트워크(150)는 회선 교환 및/또는 패킷 교환 채널들에서 데이터를 교환할 수 있다. 대표적인 네트워크들은 통신 네트워크들, 로컬 영역 네트워크들, 광역 네트워크들 및/또는 인터넷을 포함한다. 본 논의의 목적을 위해, 네트워크(150)의 아키텍처 및 토폴로지는 아래에서 본 명세서에서 설명되지 않는 한 본 개시내용의 동작에 중요하지 않을 수 있다.

[0017] 도 2는, 개시된 주제를 위한 응용의 예로서, 스트리밍 환경에서의 비디오 인코더 및 디코더의 배치를 예시한다. 개시된 주제는, 예를 들어, 영상 회의, 디지털 TV, CD, DVD, 메모리 스틱 등을 포함하는 디지털 미디어 상의 압축된 비디오의 저장 등을 포함하여, 다른 비디오 가능 응용들에 동등하게 적용가능할 수 있다.

[0018] 스트리밍 시스템은, 예를 들어, 압축되지 않은 비디오 샘플 스트림(202)을 생성하는 비디오 소스(201), 예를 들어, 디지털 카메라를 포함할 수 있는, 캡처 서브시스템(213)을 포함할 수 있다. 인코딩된 비디오 비트스트림들

과 비교할 때 많은 데이터 용량을 강조하기 위해 굵은 라인으로 묘사된 그 샘플 스트림(202)은 카메라(201)에 결합된 인코더(203)에 의해 처리될 수 있다. 인코더(203)는 아래에서 더 상세히 설명되는 바와 같이 개시된 주제의 양태들을 가능하게 하거나 구현하기 위해 하드웨어, 소프트웨어, 또는 이들의 조합을 포함할 수 있다. 샘플 스트림과 비교할 때 더 적은 데이터 용량을 강조하기 위해 얇은 라인으로 묘사된 인코딩된 비디오 비트스트림(204)은 장래의 사용을 위해 스트리밍 서버(205) 상에 저장될 수 있다. 하나 이상의 스트리밍 클라이언트(206, 208)는 스트리밍 서버(205)에 액세스하여 인코딩된 비디오 비트스트림(204)의 사본들(207, 209)을 검색할 수 있다. 클라이언트(206)는 인코딩된 비디오 비트스트림의 착신(incoming) 사본(207)을 디코딩하고 디스플레이(212) 또는 다른 렌더링 디바이스(묘사되지 않음) 상에 렌더링될 수 있는 발신(outgoing) 비디오 샘플 스트림(211)을 생성하는 비디오 디코더(210)를 포함할 수 있다. 일부 스트리밍 시스템들에서, 비디오 비트스트림들(204, 207, 209)은 특정 비디오 코딩/압축 표준들에 따라 인코딩될 수 있다. 그러한 표준들의 예들은 ITU-T 권고안(Recommendation) H.265를 포함한다. 다음도 비디오 코딩(Versatile Video Coding) 또는 VVC로서 비공식적으로 알려진 비디오 코딩 표준이 개발 중이다. 개시된 주제는 VVC의 맥락에서 사용될 수 있다.

[0019] 도 3은 본 개시내용의 실시예에 따른 비디오 디코더(210)의 기능 블록도일 수 있다.

[0020] 수신기(310)가 디코더(210)에 의해 디코딩될 하나 이상의 코덱 비디오 시퀀스를 수신할 수 있으며; 동일한 또는 다른 실시예에서는, 한 번에 하나의 코딩된 비디오 시퀀스를 수신할 수 있으며, 여기서 각각의 코딩된 비디오 시퀀스의 디코딩은 다른 코딩된 비디오 시퀀스들과 독립적이다. 코딩된 비디오 시퀀스는, 인코딩된 비디오 데이터를 저장하는 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는, 채널(312)로부터 수신될 수 있다. 수신기(310)는 인코딩된 비디오 데이터를 다른 데이터, 예를 들어, 코딩된 오디오 데이터 및/또는 보조 데이터 스트림들과 함께 수신할 수 있고, 이들은 그것들 각각의 사용 엔티티들(묘사되지 않음)에 포워딩될 수 있다. 수신기(310)는 코딩된 비디오 시퀀스를 다른 데이터로부터 분리할 수 있다. 네트워크 지터를 방지하기 위해, 수신기(310)와 엔트로피 디코더/파서(320)(이후 "파서") 사이에 버퍼 메모리(315)가 결합될 수 있다. 수신기(310)가 충분한 대역폭 및 제어가능성의 저장/포워드 디바이스로부터, 또는 등시 동기식 네트워크(isosynchronous network)로부터 데이터를 수신하고 있을 때, 버퍼(315)는 필요하지 않을 수 있거나, 작을 수 있다. 인터넷과 같은 베스트 에포트 패킷 네트워크들(best effort packet networks) 상에서의 사용을 위해, 버퍼(315)는 요구될 수 있고, 비교적 클 수 있으며, 유리하게는 적응적 크기일 수 있다.

[0021] 비디오 디코더(210)는 엔트로피 코딩된 비디오 시퀀스로부터 심벌들(321)을 재구성하기 위해 파서(320)를 포함할 수 있다. 그 심벌들의 카테고리들은 디코더(210)의 동작을 관리하기 위해 사용되는 정보, 및 잠재적으로, 도 3에 도시된 바와 같이, 디코더의 일체 부분(integral part)은 아니지만 디코더에 결합될 수 있는 디스플레이(212)와 같은 렌더링 디바이스를 제어하기 위한 정보를 포함한다. 렌더링 디바이스(들)에 대한 제어 정보는 SEI(Supplementary Enhancement Information) 메시지들 또는 VUI(Video Usability Information) 파라미터 세트 프레임먼트들(묘사되지 않음)의 형태로 될 수 있다. 파서(320)는 수신되는 코딩된 비디오 시퀀스를 파싱 및/또는 엔트로피 디코딩할 수 있다. 코딩된 비디오 시퀀스의 코딩은 비디오 코딩 기술 또는 표준에 따를 수 있고, 가변 길이 코딩, 허프만 코딩(Huffman coding), 맥락 민감성(context sensitivity)을 갖거나 갖지 않는 산술 코딩 등을 포함한 본 기술분야의 통상의 기술자에게 잘 알려진 원리들을 따를 수 있다. 파서(320)는, 코딩된 비디오 시퀀스로부터, 그룹에 대응하는 적어도 하나의 파라미터에 기초하여, 비디오 디코더 내의 픽셀들의 서브 그룹들 중 적어도 하나에 대한 서브그룹 파라미터들의 세트를 추출할 수 있다. 서브그룹들은 픽처 그룹(Group of Pictures, GOP)들, 픽처들, 서브-픽처들, 타일들, 슬라이스들, 브릭들, 매크로블록들, 코딩 트리 유닛(Coding Tree Unit, CTU)들, 코딩 유닛(Coding Unit, CU)들, 블록들, 변환 유닛(Transform Unit, TU)들, 예측 유닛(Prediction Unit, PU)들 등을 포함할 수 있다. 타일(tile)은 픽처의 특정 타일 열 및 행 내의 CU/CTU들의 직사각형 영역을 표시할 수 있다. 브릭(brick)은 특정 타일 내의 CU/CTU 행들의 직사각형 영역을 표시할 수 있다. 슬라이스(slice)는 NAL 유닛에 포함된 픽처의 하나 이상의 브릭을 표시할 수 있다. 서브-픽처(sub-picture)는 픽처 내의 하나 이상의 슬라이스의 직사각형 영역을 표시할 수 있다. 엔트로피 디코더/파서는 또한 코딩된 비디오 시퀀스로부터 변환 계수들, 양자화기 파라미터 값들, 모션 벡터들 등과 같은 정보를 추출할 수 있다.

[0022] 파서(320)는 버퍼(315)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩 및/또는 파싱 동작을 수행하여, 심벌들(321)을 생성할 수 있다.

[0023] 심벌들(321)의 재구성은 코딩된 비디오 픽처 또는 그의 부분들의 타입(예컨대: 인터 및 인트라 픽처, 인터 및 인트라 블록), 및 다른 팩터들에 따라 다수의 상이한 유닛들을 수반할 수 있다. 어느 유닛들이 수반되는지, 그리고 그 방식은 파서(320)에 의해 코딩된 비디오 시퀀스로부터 파싱된 서브그룹 제어 정보에 의해 제어될 수 있

다. 파서(320)와 아래의 다수의 유닛 사이의 이러한 서브그룹 제어 정보의 흐름은 명확성을 위해 묘사되어 있지 않다.

- [0024] 이미 언급된 기능 블록들 이외에, 디코더(210)는 아래에 설명되는 바와 같이 개념적으로 다수의 기능 유닛으로 세분될 수 있다. 상업적 제약 하에서 동작하는 실제 구현에서, 이들 유닛 중 다수는 서로 밀접하게 상호작용하고, 적어도 부분적으로 서로 통합될 수 있다. 그러나, 개시된 주제를 설명하기 위해서는, 아래의 기능 유닛들로의 개념적 세분이 적절하다.
- [0025] 제1 유닛은 스케일러 및/또는 역변환 유닛(351)이다. 스케일러 및/또는 역변환 유닛(351)은, 파서(320)로부터의 심벌(들)(321)로서, 어느 변환을 사용할지, 블록 크기, 양자화 팩터, 양자화 스케일링 행렬들(quantization scaling matrices) 등을 포함한, 제어 정보뿐만 아니라 양자화된 변환 계수를 수신한다. 이는 집계기(aggregator)(355)에 입력될 수 있는 샘플 값들을 포함하는 블록들을 출력할 수 있다.
- [0026] 일부 경우들에서, 스케일러 및/또는 역변환(351)의 출력 샘플들은 인트라 코딩된 블록; 즉, 이전에 재구성된 픽처들로부터의 예측 정보를 사용하는 것이 아니고, 현재 픽처의 이전에 재구성된 부분들로부터의 예측 정보를 사용할 수 있는 블록에 관련될 수 있다. 그러한 예측 정보는 인트라 픽처 예측 유닛(352)에 의해 제공될 수 있다. 일부 경우들에서, 인트라 픽처 예측 유닛(352)은 현재 (부분적으로 재구성된) 픽처(358)로부터 폐치된 주위의 이미 재구성된 정보를 사용하여, 재구성 중인 블록과 동일한 크기 및 형상의 블록을 생성한다. 집계기(355)는, 일부 경우들에서, 샘플당 기준으로, 인트라 예측 유닛(352)이 생성한 예측 정보를 스케일러 및/또는 역변환 유닛(351)에 의해 제공된 출력 샘플 정보에 추가한다.
- [0027] 다른 경우들에서, 스케일러 및/또는 역변환 유닛(351)의 출력 샘플들은 인트라 코딩되고, 잠재적으로 모션 보상된 블록에 관련될 수 있다. 그러한 경우에, 모션 보상 예측 유닛(353)은 참조 픽처 메모리(357)에 액세스하여 예측에 사용되는 샘플들을 폐치할 수 있다. 블록에 관련된 심벌들(321)에 따라 폐치된 샘플들을 모션 보상한 후에, 이들 샘플은 집계기(355)에 의해 스케일러 및/또는 역변환 유닛의 출력(이 경우 잔차 샘플들 또는 잔차 신호라고 불림)에 추가되어 출력 샘플 정보를 생성할 수 있다. 모션 보상 유닛이 예측 샘플들을 폐치하는 참조 픽처 메모리 내의 어드레스들은, 예를 들어 X, Y, 및 참조 픽처 컴포넌트들을 가질 수 있는 심벌들(321)의 형태로 모션 보상 유닛에 이용가능한 모션 벡터들에 의해 제어될 수 있다. 모션 보상은 또한 서브-샘플 정확한 모션 벡터들이 사용중일 때 참조 픽처 메모리로부터 폐치된 샘플 값들의 보간, 모션 벡터 예측 메커니즘 등을 포함할 수 있다.
- [0028] 집계기(355)의 출력 샘플들에 대해 루프 필터 유닛(356) 내의 다양한 루프 필터링 기법들이 수행될 수 있다. 비디오 압축 기술들은, 파서(320)로부터의 심벌들(321)로서 루프 필터 유닛(356)에 이용가능하게 되고 코딩된 비디오 비트스트림에 포함된 파라미터들에 의해 제어되지만, 코딩된 픽처 또는 코딩된 비디오 시퀀스의 이전(디코딩 순서로) 부분들의 디코딩 동안 획득된 메타-정보에 응답할 뿐만 아니라, 이전에 재구성된 및 루프-필터링된 샘플 값들에 응답할 수도 있는 인-루프 필터(in-loop filter) 기술들을 포함할 수 있다.
- [0029] 루프 필터 유닛(356)의 출력은 렌더링 디바이스(212)에 출력될 뿐만 아니라 장래의 인트라-픽처 예측에서 사용하기 위해 참조 픽처 메모리에 저장될 수도 있는 샘플 스트림일 수 있다.
- [0030] 특정 코딩된 픽처들은, 완전히 재구성되면, 장래의 예측을 위한 참조 픽처들로서 사용될 수 있다. 코딩된 픽처가 완전히 재구성되고 코딩된 픽처가 참조 픽처로서 식별되면(예를 들어, 파서(320)에 의해), 현재 참조 픽처(358)는 참조 픽처 버퍼(357)의 일부가 될 수 있고, 다음 코딩된 픽처의 재구성에 착수하기 전에 새로운(fresh) 현재 픽처 메모리가 재할당될 수 있다.
- [0031] 비디오 디코더(210)는 ITU-T Rec. H.265와 같은 표준에서 문서화될 수 있는 미리 결정된 비디오 압축 기술에 따라 디코딩 동작들을 수행할 수 있다. 코딩된 비디오 시퀀스는, 비디오 압축 기술 문서 또는 표준에서 그리고 구체적으로 그 내부의 프로파일 문서에서 특정된 바와 같은, 비디오 압축 기술 또는 표준의 선택스를 고수한다는 점에서, 사용중인 비디오 압축 기술 또는 표준에 의해 특정된 선택스를 준수할 수 있다. 또한 준수를 위해, 코딩된 비디오 시퀀스의 복잡도가 비디오 압축 기술 또는 표준의 레벨에 의해 정의된 경계 내에 있는 것이 필요할 수 있다. 일부 경우들에서, 레벨들은 최대 픽처 크기, 최대 프레임 레이트, 최대 재구성 샘플 레이트(예를 들어, 초당 메가 샘플수로 측정됨), 최대 참조 픽처 크기 등을 제한한다. 레벨들에 의해 설정된 한계들은, 일부 경우들에서, HRD(Hypothetical Reference Decoder) 사양 및 코딩된 비디오 시퀀스에서 시그널링된 HRD 버퍼 관리를 위한 메타데이터를 통해 추가로 제한될 수 있다.
- [0032] 일 실시예에서, 수신기(310)는 인코딩된 비디오와 함께 추가적인(중복) 데이터를 수신할 수 있다. 이 추가적인

데이터는 코딩된 비디오 시퀀스(들)의 일부로서 포함될 수 있다. 이 추가적인 데이터는 데이터를 적절히 디코딩하고/하거나 원래의 비디오 데이터를 더 정확하게 재구성하기 위해 비디오 디코더(210)에 의해 사용될 수 있다. 추가적인 데이터는 예를 들어, 시간적, 공간적, 또는 SNR 향상 계층들, 중복 슬라이스들, 중복 픽처들, 순방향 오류 정정 코드들 등의 형태로 될 수 있다.

[0033] 도 4는 본 개시내용의 실시예에 따른 비디오 인코더(203)의 기능 블록도일 수 있다.

[0034] 인코더(203)는 인코더(203)에 의해 코딩될 비디오 이미지(들)를 캡처할 수 있는 비디오 소스(201)(인코더의 일부가 아님)로부터 비디오 샘플들을 수신할 수 있다.

[0035] 비디오 소스(201)는, 임의의 적합한 비트 심도(예를 들어: 8 비트, 10 비트, 12 비트, ...), 임의의 색 공간(예를 들어, BT.601 Y CrCb, RGB, ...), 및 임의의 적합한 샘플링 구조(예를 들어, Y CrCb 4:2:0, Y CrCb 4:4:4) 일 수 있는 디지털 비디오 샘플 스트림의 형태로 인코더(203)에 의해 코딩될 소스 비디오 시퀀스를 제공할 수 있다. 미디어 서빙 시스템에서, 비디오 소스(201)는 이전에 준비된 비디오를 저장하는 저장 디바이스일 수 있다. 영상 회의 시스템에서, 비디오 소스(203)는 비디오 시퀀스로서 로컬 이미지 정보를 캡처하는 카메라일 수 있다. 비디오 데이터는 순차적으로 볼 때 모션을 부여하는 복수의 개별 픽처로서 제공될 수 있다. 픽처들 자체는 픽셀들의 공간적 어레이로서 조직될 수 있고, 여기서 각각의 픽셀은 사용중인 샘플링 구조, 색 공간 등에 의존하여 하나 이상의 샘플을 포함할 수 있다. 본 기술분야의 통상의 기술자는 픽셀들과 샘플들 사이의 관계를 쉽게 이해할 수 있다. 아래의 설명은 샘플들에 초점을 맞춘다.

[0036] 일 실시예에 따르면, 인코더(203)는 소스 비디오 시퀀스의 픽처들을 실시간으로 또는 응용에 의해 요구되는 임의의 다른 시간 제약들 하에서 코딩된 비디오 시퀀스(443)로 코딩 및 압축할 수 있다. 적절한 코딩 속도를 시행하는 것이 제어기(450)의 하나의 기능이다. 제어기는 아래에 설명되는 바와 같이 다른 기능 유닛들을 제어하고 이들 유닛들에 기능적으로 결합된다. 결합은 명확성을 위해 묘사되어 있지 않다. 제어기에 의해 설정된 파라미터들은 레이트 제어 관련 파라미터들(픽처 스킵, 양자화기, 레이트-왜곡 최적화 기법들의 램다 값, ...), 픽처 크기, 픽처 그룹(GOP) 레이아웃, 최대 모션 벡터 검색 범위 등을 포함할 수 있다. 본 기술분야의 통상의 기술자는 제어기(450)의 다른 기능들을 쉽게 식별할 수 있는데 그 이유는 그것들이 특정 시스템 설계에 대해 최적화된 비디오 인코더(203)에 관련될 수 있기 때문이다.

[0037] 일부 비디오 인코더들은 본 기술분야의 통상의 기술자가 "코딩 루프"로서 쉽게 인식하는 것에서 동작한다. 과도하게 단순화된 설명으로서, 코딩 루프는, (코딩될 입력 픽처, 및 참조 픽처(들)에 기초하여 심벌들을 생성하는 것을 담당하는) 인코더(430)(이후 "소스 코더")의 인코딩 부분, 및 (심벌들과 코딩된 비디오 비트스트림 사이의 임의의 압축이 개시된 주제에서 고려되는 비디오 압축 기술들에서 무손실이기 때문에) (원격) 디코더가 또한 생성하는 샘플 데이터를 생성하기 위해 심벌들을 재구성하는 인코더(203)에 임베드된 (로컬) 디코더(433)로 구성될 수 있다. 그 재구성된 샘플 스트림은 참조 픽처 메모리(434)에 입력된다. 심벌 스트림의 디코딩이 디코더 위치(로컬 또는 원격)와는 독립적으로 비트-정확한 결과들(bit-exact results)을 야기하기 때문에, 참조 픽처 버퍼 콘텐츠도 또한 로컬 인코더와 원격 인코더 사이에서 비트 정확(bit exact)하다. 다시 말해서, 인코더의 예측 부분은 디코딩 동안 예측을 사용할 때 디코더가 "보는(see)" 것과 정확히 동일한 샘플 값들을 참조 픽처 샘플들로서 "본다". 참조 픽처 동기성(reference picture synchronicity)의 이러한 기본적인 원리(그리고, 예를 들어, 채널 오류들 때문에, 동기성이 유지될 수 없는 경우, 결과적인 드리프트)는 본 기술분야의 통상의 기술자에게 잘 알려져 있다.

[0038] "로컬" 디코더(433)의 동작은 도 3과 관련하여 위에서 이미 상세히 설명한 "원격" 디코더(210)의 동작과 동일할 수 있다. 그러나, 도 4를 또한 간단히 참조하면, 심벌들이 이용가능하고 엔트로피 코더(445) 및 파서(320)에 의한 코딩된 비디오 시퀀스로서의 심벌들의 인코딩 및/또는 디코딩이 무손실일 수 있기 때문에, 채널(312), 수신기(310), 버퍼(315), 및 파서(320)를 포함한, 디코더(210)의 엔트로피 디코딩 부분들은 로컬 디코더(433)에서 완전히 구현되지 않을 수 있다.

[0039] 이 시점에서 이루어질 수 있는 관찰은, 디코더에 존재하는 파싱 및/또는 엔트로피 디코딩을 제외한 임의의 디코더 기술이 또한 필연적으로, 대응하는 인코더에서, 실질적으로 동일한 기능 형태로 존재할 필요가 있다는 점이다. 이러한 이유로, 개시된 주제는 디코더 동작에 초점을 맞춘다. 인코더 기술들은 포괄적으로 설명된 디코더 기술들의 역(inverse)이기 때문에 그것들에 대한 설명은 축약될 수 있다. 특정 영역들에서만 더 상세한 설명이 요구되고 아래에 제공된다.

[0040] 동작의 일부로서, 소스 코더(430)는, "참조 프레임들"로서 지정된 비디오 시퀀스로부터의 하나 이상의 이전에

코딩된 프레임을 참조하여 예측적으로 입력 프레임을 코딩하는, 모션 보상된 예측 코딩을 수행할 수 있다. 이러한 방식으로, 코딩 엔진(432)은 입력 프레임의 픽셀 블록들과 입력 프레임에 대한 예측 참조(들)로서 선택될 수 있는 참조 프레임(들)의 픽셀 블록들 사이의 차이들을 코딩한다.

[0041] 로컬 비디오 디코더(433)는, 소스 코더(430)에 의해 생성된 심벌들에 기초하여, 참조 프레임들로서 지정될 수 있는 프레임들의 코딩된 비디오 데이터를 디코딩할 수 있다. 코딩 엔진(432)의 동작들은 유리하게는 손실 프로세스들일 수 있다. 코딩된 비디오 데이터가 비디오 디코더(도 4에 도시되지 않음)에서 디코딩될 수 있는 경우, 재구성된 비디오 시퀀스는 전형적으로 일부 오류들을 갖는 소스 비디오 시퀀스의 복제본(replica)일 수 있다. 로컬 비디오 디코더(433)는 참조 프레임들에 대해 비디오 디코더에 의해 수행될 수 있는 디코딩 프로세스들을 복제하고 재구성된 참조 프레임들이 참조 픽처 캐시(434)에 저장되게 할 수 있다. 이러한 방식으로, 인코더(203)는 (송신 오류들이 없이) 원단(far-end) 비디오 디코더에 의해 획득될 재구성된 참조 프레임들로서 공통 콘텐츠를 갖는 재구성된 참조 프레임들의 사본들을 로컬로 저장할 수 있다.

[0042] 예측자(435)는 코딩 엔진(432)에 대한 예측 검색들을 수행할 수 있다. 즉, 코딩될 새로운 프레임에 대해, 예측자(435)는 새로운 픽처들에 대한 적절한 예측 참조로서 역할을 할 수 있는 참조 픽처 모션 벡터들, 블록 형상들 등과 같은 특정 메타데이터 또는 샘플 데이터(후보 참조 픽셀 블록들로서)에 대해 참조 픽처 메모리(434)를 검색할 수 있다. 예측자(435)는 적절한 예측 참조들을 찾기 위해 샘플 블록-바이-픽셀 블록(sample block-by-pixel block) 기준으로 동작할 수 있다. 일부 경우들에서, 예측자(435)에 의해 획득된 검색 결과들에 의해 결정된 바와 같이, 입력 픽처는 참조 픽처 메모리(434)에 저장된 다수의 참조 픽처로부터 인출된 예측 참조들을 가질 수 있다.

[0043] 제어기(450)는, 예를 들어, 비디오 데이터를 인코딩하기 위해 사용되는 파라미터들 및 서브그룹 파라미터들의 설정을 포함하여, 비디오 코더(430)의 코딩 동작을 관리할 수 있다.

[0044] 전술한 모든 기능 유닛들의 출력은 엔트로피 코더(445)에서 엔트로피 코딩을 거칠 수 있다. 엔트로피 코더는, 예를 들어, 허프만 코딩, 가변 길이 코딩, 산술 코딩 등과 같은 본 기술분야의 통상의 기술자에게 알려진 기술들에 따라 심벌들을 무손실 압축함으로써, 다양한 기능 유닛들에 의해 생성된 심벌들을 코딩된 비디오 시퀀스로 변환한다.

[0045] 송신기(440)는, 인코딩된 비디오 데이터를 저장할 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는, 통신 채널(460)을 통한 송신을 준비하기 위해 엔트로피 코더(445)에 의해 생성된 코딩된 비디오 시퀀스(들)를 버퍼링할 수 있다. 송신기(440)는 비디오 코더(430)로부터의 코딩된 비디오 데이터를 송신될 다른 데이터, 예를 들어, 코딩된 오디오 데이터 및/또는 보조 데이터 스트림(소스들이 도시되지 않음)과 병합할 수 있다.

[0046] 제어기(450)는 인코더(203)의 동작을 관리할 수 있다. 코딩 동안, 제어기(450)는, 각자의 픽처에 적용될 수 있는 코딩 기법들에 영향을 미칠 수 있는, 특정 코딩된 픽처 타입을 각각의 코딩된 픽처에 할당할 수 있다. 예를 들어, 픽처들은 종종 다음의 프레임 타입들 중 하나로서 할당될 수 있다:

[0047] 인트라 픽처(Intra Picture)(I 픽처)는 예측의 소스로서 시퀀스 내의 임의의 다른 프레임을 사용하지 않고 코딩 및 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱들은, 예를 들어, 독립 디코더 리프레시(Independent Decoder Refresh) 픽처들을 포함한, 상이한 타입들의 인트라 픽처들을 허용한다. 본 기술분야의 통상의 기술자는 I 픽처들의 해당 변형들 및 그것들 각각의 응용들 및 특징들을 인식한다.

[0048] 예측 픽처(Predictive picture)(P 픽처)는 각각의 블록의 샘플 값들을 예측하기 위해 많아야 하나의 모션 벡터 및 참조 인덱스를 사용하여 인트라 예측(intra prediction) 또는 인터 예측(inter prediction)을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다.

[0049] 양방향 예측 픽처(Bi-directionally Predictive Picture)(B 픽처)는 각각의 블록의 샘플 값들을 예측하기 위해 많아야 2개의 모션 벡터 및 참조 인덱스를 사용하여 인트라 예측 또는 인터 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다. 유사하게, 다중-예측 픽처들은 단일 블록의 재구성을 위해 2개보다 많은 참조 픽처 및 연관된 메타데이터를 사용할 수 있다.

[0050] 소스 픽처들은 일반적으로 복수의 샘플 블록(예를 들어, 각각 4x4, 8x8, 4x8, 또는 16x16 샘플들의 블록들)으로 공간적으로 세분되고 블록-바이-블록(block-by-block) 기준으로 코딩될 수 있다. 블록들은 블록들의 각자의 픽처들에 적용되는 코딩 할당에 의해 결정된 다른(이미 코딩된) 블록들을 참조하여 예측적으로 코딩될 수 있다. 예를 들어, I 픽처들의 블록들은 비예측적으로 코딩될 수 있거나 그것들은 동일한 픽처의 이미 코딩된 블록들을 참조하여 예측적으로 코딩될 수 있다(공간적 예측 또는 인트라 예측). P 픽처들의 픽셀 블록들은, 하나의 이전

에 코딩된 참조 픽처를 참조하여 공간적 예측을 통해 또는 시간적 예측을 통해, 비예측적으로 코딩될 수 있다. B 픽처들의 블록들은, 1개 또는 2개의 이전에 코딩된 참조 픽처를 참조하여 공간적 예측을 통해 또는 시간적 예측을 통해, 비예측적으로 코딩될 수 있다.

- [0051] 비디오 코더(203)는 ITU-T Rec. H.265와 같은 미리 결정된 비디오 코딩 기술 또는 표준에 따라 코딩 동작들을 수행할 수 있다. 그 동작에서, 비디오 코더(203)는, 입력 비디오 시퀀스에서 시간적 및 공간적 중복성들을 활용하는 예측 코딩 동작들을 포함한, 다양한 압축 동작들을 수행할 수 있다. 따라서, 코딩된 비디오 데이터는 사용중인 비디오 코딩 기술 또는 표준에 의해 특정된 신택스를 준수할 수 있다.
- [0052] 일 실시예에서, 송신기(440)는 인코딩된 비디오와 함께 추가적인 데이터를 송신할 수 있다. 비디오 코더(430)는 코딩된 비디오 시퀀스의 일부로서 그러한 데이터를 포함할 수 있다. 추가적인 데이터는 시간적/공간적/SNR 향상 계층들, 중복 픽처들 및 슬라이스들과 같은 다른 형태들의 중복 데이터, SEI(Supplementary Enhancement Information) 메시지들, VUI(Visual Usability Information) 파라미터 세트 프래그먼트들 등을 포함할 수 있다.
- [0053] 최근, 다수의 시맨틱적으로 독립적인 픽처 부분들(multiple semantically independent picture parts)의 단일 비디오 픽처로의 압축된 도메인 집계(aggregation) 또는 추출이 일부 주목을 받고 있다. 특히, 예를 들어, 360 코딩 또는 특정 감시 응용들의 맥락에서, 다수의 시맨틱적으로 독립적인 소스 픽처(예를 들어, 큐브-투영된 360 장면의 6개 큐브 표면, 또는 멀티-카메라 감시 셋업의 경우 개개의 카메라 입력들)는 주어진 시점에서 상이한 장면별 활동에 대처하기 위해 별개의 적응적 해상도 설정을 요구할 수 있다. 즉, 인코더들은, 주어진 시점에서, 전체 360 또는 감시 장면을 구성하는 상이한 시맨틱적으로 독립적인 픽처들에 대해 상이한 리샘플링 팩터들을 사용하도록 선택할 수 있다. 단일 픽처로 조합될 때, 이는, 차례로, 참조 픽처 리샘플링이 수행되고, 코딩된 픽처의 부분들에 대해 적응적 해상도 코딩 시그널링이 이용가능한 것을 요구한다.
- [0054] 이하, 본 설명의 나머지에서 참조될 몇몇 용어들이 소개될 것이다.
- [0055] 서브-픽처는, 일부 경우들에서, 시맨틱적으로 그룹화되고, 변경된 해상도로 독립적으로 코딩될 수 있는, 샘플들, 블록들, 매크로블록들, 코딩 유닛들, 또는 유사한 엔티티들의 직사각형 배열을 지칭할 수 있다. 하나 이상의 서브-픽처는 픽처를 형성할 수 있다. 하나 이상의 코딩된 서브-픽처는 코딩된 픽처를 형성할 수 있다. 하나 이상의 서브-픽처가 픽처로 조립될 수 있고, 하나 이상의 서브-픽처가 픽처로부터 추출될 수 있다. 특정 환경들에서, 하나 이상의 코딩된 서브-픽처는 샘플 레벨로 트랜스코딩하지 않고 압축된 도메인에서 코딩된 픽처로 조립될 수 있고, 동일한 또는 다른 경우들에서, 하나 이상의 코딩된 서브-픽처는 압축된 도메인에서의 코딩된 픽처로부터 추출될 수 있다.
- [0056] 적응적 해상도 변경(Adaptive Resolution Change, ARC)은, 예를 들어, 참조 픽처 리샘플링에 의해, 코딩된 비디오 시퀀스 내의 픽처 또는 서브픽처의 해상도의 변경을 허용하는 메커니즘들을 지칭할 수 있다. 이후, ARC 파라미터들은, 예를 들어, 필터 파라미터들, 스케일링 팩터들, 출력 및/또는 참조 픽처들의 해상도들, 다양한 제어 플래그들 등을 포함할 수 있는, 적응적 해상도 변경을 수행하는데 요구되는 제어 정보를 지칭한다.
- [0057] 실시예들에서, 코딩 및 디코딩은 단일의 시맨틱적으로 독립적인 코딩된 비디오 픽처에 대해 수행될 수 있다. 독립적인 ARC 파라미터들을 갖는 다수의 서브-픽처의 코딩/디코딩의 함축적 의미(implication) 및 그의 함축된 추가적인 복잡성을 설명하기 전에, ARC 파라미터들을 시그널링하기 위한 옵션들이 설명될 것이다.
- [0058] 도 5a 내지 도 5e를 참조하면, ARC 파라미터들을 시그널링하기 위한 몇몇 실시예들이 도시되어 있다. 실시예들 각각에서 언급된 바와 같이, 이들은 코딩 효율, 복잡성, 및 아키텍처 관점에서 특정 장점들 및 특정 단점들을 가질 수 있다. 비디오 코딩 표준 또는 기술은 ARC 파라미터들을 시그널링하기 위해 이러한 실시예들 또는 관련 기술로부터 알려진 옵션들 중 하나 이상을 선택할 수 있다. 실시예들은 상호 배타적이지 않을 수 있고, 가능하게는 응용 요구, 수반된 표준 기술, 또는 인코더의 선택에 기초하여 상호 교환될 수 있다.
- [0059] ARC 파라미터들의 클래스들은 다음을 포함할 수 있다:
- [0060] - X 및 Y 차원에서 별개이거나 조합되는 업샘플 및/또는 다운샘플 팩터들.
- [0061] - 주어진 수의 픽처들에 대해 일정 속도 줌인/줌아웃을 나타내는, 시간적 차원이 추가된 업샘플 및/또는 다운샘플 팩터들.
- [0062] - 위의 2개 중 어느 하나는 팩터(들)를 포함하는 테이블을 가리킬 수 있는 하나 이상의 아마도 짧은 신택스 요

소의 코딩을 수반할 수 있다.

- [0063] - 입력 픽처, 출력 픽처, 참조 픽처, 코딩된 픽처의, 샘플들, 블록들, 매크로블록들, 코딩 유닛(CU)들의 단위의, 또는 임의의 다른 적절한 입도의, X 또는 Y 차원에서의, 조합되거나 별개의 해상도. 하나보다 많은 해상도(예를 들어, 입력 픽처에 대해 하나, 참조 픽처에 대해 하나 등)가 있으면, 특정 경우들에서, 값들의 하나의 세트는 값들의 다른 세트로부터 추론될 수 있다. 이는, 예를 들어, 플래그들의 사용에 의해 게이팅될 수 있다. 더 상세한 예에 대해서는, 아래를 참조한다.
- [0064] - 위에서 설명한 바와 같이 다시 적절한 입도로, H.263 Annex P에서 사용된 것과 유사한 "워핑(warping)" 좌표. H.263 Annex P는 이러한 워핑 좌표들을 코딩하기 위한 하나의 효율적인 방식을 정의하지만, 다른 잠재적으로 더 효율적인 방식들도 아마도 또한 고안될 수 있다. 예를 들어, Annex P의 워핑 좌표들의 가변 길이 가역적 "허프만"-스타일 코딩은 적합한 길이 이진 코딩에 의해 대체될 수 있고, 여기서 이진 코드 워드의 길이는, 예를 들어, 최대 픽처 크기의 경계들 밖에서의 "워핑"을 허용하기 위해, 가능하게는 특정 인수가 곱해지고 특정 값으로 오프셋된, 최대 픽처 크기로부터 도출될 수 있다.
- [0065] - 업샘플 및/또는 다운샘플 필터 파라미터들. 실시예들에서, 업샘플링 및/또는 다운샘플링을 위한 단일 필터만이 있을 수 있다. 그러나, 실시예들에서는, 필터 설계에 더 많은 유연성을 허용하는 것이 바람직할 수 있고, 이는 필터 파라미터들의 시그널링을 요구할 수 있다. 이러한 파라미터들은 가능한 필터 설계들의 리스트에서의 인덱스를 통해 선택될 수 있고, 필터는 (예를 들어, 적합한 엔트로피 코딩 기법들을 사용하여, 필터 계수들의 리스트를 통해) 완전히 특정될 수 있고, 필터는 위에서 언급한 메커니즘들 중 임의의 것에 따라 차례로 시그널링되는 업샘플 및/또는 다운샘플 비율들을 통해 암시적으로 선택될 수 있는 등이다.
- [0066] 이후, 본 설명은 코드워드를 통해 표시된, 유한 세트의 업샘플 및/또는 다운샘플 팩터들(X 및 Y 차원 양자 모두에서 사용될 동일한 팩터)의 코딩을 가정한다. 그 코드워드는, 예를 들어, H.264 및 H.265와 같은 비디오 코딩 사양들에서 특정 신택스 요소들에 공통인 Ext-Golomb 코드를 사용하여 가변 길이 코딩될 수 있다. 업샘플 및/또는 다운샘플 팩터들에 대한 값들의 하나의 적합한 맵핑은, 예를 들어, 표 1에 따를 수 있다.

표 1

코드워드	Ext-Golomb 코드	원래 / 타깃 해상도
0	1	1 / 1
1	010	1 / 1.5 (50% 업스케일)
2	011	1.5 / 1 (50% 다운스케일)
3	00100	1 / 2 (100% 업스케일)
4	00101	2 / 1 (100% 다운스케일)

- [0067]
- [0068] 비디오 압축 기술 또는 표준에서 이용가능한 업 및 다운스케일 메커니즘들의 능력들 및 응용 요구들에 따라 많은 유사한 맵핑이 고안될 수 있다. 표는 더 많은 값으로 확장될 수 있다. 값들은 또한, 예를 들어, 이진 코딩을 사용하여, Ext-Golomb 코드들 이외의 엔트로피 코딩 메커니즘들에 의해 표현될 수 있다. 이는, 리샘플링 팩터들이, 예를 들어, MANE에 의해, 비디오 처리 엔진들(주요한 인코더 및 디코더) 자체의 외부에서 관심 대상일 때 특정 이점들을 가질 수 있다. 해상도 변경이 요구되지 않는 상황들에 대해, 짧은; 위의 표에서, 단지 단일 비트인, Ext-Golomb 코드가 선택될 수 있다는 점에 유의해야 한다. 이는 가장 흔한 경우에 대해 이진 코드들을 사용하는 것에 비해 코딩 효율 이점을 가질 수 있다.
- [0069] 테이블 내의 엔트리들의 수뿐만 아니라 그것들의 시맨틱은 완전히 또는 부분적으로 구성가능할 수 있다. 예를 들어, 테이블의 기본 윤곽은 시퀀스 또는 디코더 파라미터 세트와 같은 "높은" 파라미터 세트로 전달될 수 있다. 실시예들에서, 하나 이상의 이러한 테이블은 비디오 코딩 기술 또는 표준에서 정의될 수 있고, 예를 들어, 디코더 또는 시퀀스 파라미터 세트를 통해 선택될 수 있다.
- [0070] 위에서 설명한 바와 같이 코딩된 업샘플 및/또는 다운샘플 팩터(ARC 정보)가 비디오 코딩 기술 또는 표준 신택스에 어떻게 포함될 수 있는지가 아래에서 설명된다. 유사한 고려사항들이 업샘플 및/또는 다운샘플 필터들을 제어하는 하나, 또는 몇 개의, 코드워드들에 적용될 수 있다. 필터 또는 다른 데이터 구조들에 대해 비교적 많

은 양의 데이터가 요구될 때의 논의에 대해서는 이하를 참조한다.

- [0071] 도 5a에 도시된 바와 같이, H.263 Annex P는 픽처 헤더(501)에, 구체적으로는 H.263 PLUSPTYPE(503) 헤더 확장에, 4개의 워핑 좌표의 형태로 ARC 정보(502)를 포함한다. 이것은 a) 이용가능한 픽처 헤더가 있고, b) ARC 정보의 빈번한 변경이 예상될 때 합리적인 설계 선택일 수 있다. 그러나, H.263-스타일 시그널링을 사용할 때의 오버헤드는 매우 높을 수 있고, 픽처 헤더가 과도 성질을 가질 수 있기 때문에 스케일링 팩터들은 픽처 경계들 사이에 관련되지 않을 수 있다.
- [0072] 도 5b에 도시된 바와 같이, JVCET-M135-v1은, 차례로 시퀀스 파라미터 세트(507) 내부에 위치하는 타깃 해상도들을 포함하는 테이블(506)을 인덱싱하는, 픽처 파라미터 세트(504)에 위치한 ARC 참조 정보(505)(인덱스)를 포함한다. 시퀀스 파라미터 세트(507)의 테이블(506) 내의 가능한 해상도의 배치는, 작성자에 의해 이루어진 구두 진술(verbal statement)에 따라, 능력 교환 동안에 상호운용성 협상 포인트(interoperability negotiation point)로서 SPS를 사용함으로써 정당화될 수 있다. 해상도는 적절한 픽처 파라미터 세트(504)를 참조함으로써 픽처마다 테이블(506) 내의 값들에 의해 설정된 한계들 내에서 변경할 수 있다.
- [0073] 도 5c 내지 도 5e를 참조하면, 비디오 비트스트림으로 ARC 정보를 전달하기 위해 다음의 실시예들이 존재할 수 있다. 이러한 옵션들 각각은 위에서 설명한 실시예들에 비해 특정 이점들을 갖는다. 실시예들은 동일한 비디오 코딩 기술 또는 표준에 동시에 존재할 수 있다.
- [0074] 실시예들, 예를 들어, 도 5c에 도시된 실시예에서, 리샘플링(줌) 팩터와 같은 ARC 정보(509)는 슬라이스 헤더, GOP 헤더, 타일 헤더, 또는 타일 그룹 헤더에 존재할 수 있다. 도 5c는 타일 그룹 헤더(508)가 사용되는 실시예를 예시한다. 이것은, 예를 들어, 위에서 나타낸 바와 같이, 단일 가변 길이 ue(v) 또는 몇 비트의 고정 길이 코드워드와 같이, ARC 정보가 작은 경우에 적절할 수 있다. 타일 그룹 헤더에 직접 ARC 정보를 갖는 것이 ARC 정보의 추가적인 이점을 갖는다는 것은, 전체 픽처보다는, 예를 들어, 그 타일 그룹에 의해 표현되는 서브픽처에 적용가능할 수 있다. 하기를 또한 참조한다. 또한, 비디오 압축 기술 또는 표준이 (예를 들어, 타일 그룹 기반 적응적 해상도 변경들과는 대조적으로) 전체 픽처 적응적 해상도 변경들만을 고려하더라도, ARC 정보를 H.263-스타일 픽처 헤더에 넣는 것에 비해 ARC 정보를 타일 그룹 헤더에 넣는 것은 여러 회복력 관점에서 특정 이점들을 갖는다.
- [0075] 실시예들, 예를 들어, 도 5d에 도시된 실시예에서, ARC 정보(512) 자체는, 예를 들어, 픽처 파라미터 세트, 헤더 파라미터 세트, 타일 파라미터 세트, 적응 파라미터 세트(adaptation parameter set) 등과 같은 적절한 파라미터 세트에 존재할 수 있다. 도 5d는 적응 파라미터 세트(511)가 사용되는 실시예를 예시한다. 그 파라미터 세트의 범위는 유리하게는 픽처, 예를 들어, 타일 그룹보다 크지 않을 수 있다. ARC 정보의 사용은 관련 파라미터 세트의 활성화를 통해 암시된다. 예를 들어, 비디오 코딩 기술 또는 표준이 픽처 기반 ARC만을 고려하면, 픽처 파라미터 세트 또는 등가물이 적절할 수 있다.
- [0076] 실시예들, 예를 들어, 도 5e에 도시된 실시예에서, ARC 참조 정보(513)는 타일 그룹 헤더(514) 또는 유사한 데이터 구조에 존재할 수 있다. 그 참조 정보(513)는, 단일 픽처를 초과하는 범위를 갖는 파라미터 세트(516), 예를 들어, 시퀀스 파라미터 세트, 또는 디코더 파라미터 세트에서 이용가능한 ARC 정보(515)의 서브세트를 지칭할 수 있다.
- [0077] 도 6a에 도시된 바와 같이, 픽처의 (가능하게는 직사각형) 부분에 적용가능한 헤더의 예시적인 신택스 구조로서의 타일 그룹 헤더(601)가 가변 길이, Exp-Golomb 코딩된 신택스 요소 dec_pic_size_idx(602)(볼드체로 묘사됨)를 조건부로 포함할 수 있다. 타일 그룹 헤더 내의 이 신택스 요소의 존재는 적응적 해상도(603)의 사용에 따라 게이팅될 수 있다 - 여기서, 플래그의 값은 볼드체로 묘사되지 않으며, 이는 플래그가 신택스 다이어그램에서 발생하는 포인트에서 비트스트림에 존재한다는 것을 의미한다. 이 픽처 또는 그의 부분들에 대해 적응적 해상도가 사용 중인지 여부는 비트스트림 내부 또는 외부의 임의의 고레벨 신택스 구조로 시그널링될 수 있다. 도시된 예에서, 그것은 아래에서 약술되는 바와 같이 시퀀스 파라미터 세트로 시그널링된다.
- [0078] 도 6b를 참조하면, 시퀀스 파라미터 세트(610)의 발췌 부분이 또한 도시되어 있다. 도시된 첫번째 신택스 요소는 adaptive_pic_resolution_change_flag(611)이다. 참(true)일 때, 그 플래그는 적응적 해상도의 사용을 나타낼 수 있고, 이는 결국 특정 제어 정보를 필요로 할 수 있다. 예에서, 이러한 제어 정보는 타일 그룹 헤더(601) 및 파라미터 세트(612) 내의 if() 문에 기초한 플래그의 값에 기초하여 조건부로 존재한다.
- [0079] 적응적 해상도가 사용 중일 때, 이 예에서, 코딩된 것은 샘플 단위의 출력 해상도(613)이다. 번호 613은 output_pic_width_in_luma_samples와 output_pic_height_in_luma_samples 양자 모두를 지칭하며, 이들은 함께

출력 픽처의 해상도를 정의할 수 있다. 비디오 코딩 기술 또는 표준의 다른 곳에서는, 어느 하나의 값에 대한 특정 제한들이 정의될 수 있다. 예를 들어, 레벨 정의는 총 출력 샘플의 수를 제한할 수 있으며, 이는 그러한 2개의 선택 요소의 값의 곱일 수 있다. 또한, 특정 비디오 코딩 기술들 또는 표준들, 또는, 예를 들어, 시스템 표준들과 같은 외부 기술들 또는 표준들은 넘버링 범위(예를 들어, 하나 또는 두 치수는 2의 거듭제곱인 수로 나눌 수 있어야 함), 또는纵横비(예를 들어, 폭과 높이는 4:3 또는 16:9와 같은 관계에 있어야 함)를 제한할 수 있다. 이러한 제한들은 하드웨어 구현들을 용이하게 하기 위해 또는 다른 이유들로 도입될 수 있고, 본 기술분야에 잘 알려져 있다.

[0080] 특정 응용들에서는, 인코더가 디코더에게 그 크기가 출력 픽처 크기인 것으로 암시적으로 가정하기보다는 특정 참조 픽처 크기를 사용하도록 지시하는 것이 권장될 수 있다. 이 예에서, 선택 요소 `reference_pic_size_present_flag(614)`는 참조 픽처 치수들(615)의 조건부 존재를 게이팅한다(다시, 번호는 폭과 높이 양자 모두를 지칭함).

[0081] 마지막으로, 가능한 디코딩 픽처 폭 및 높이의 테이블이 도시된다. 이러한 테이블은, 예를 들어, 테이블 표시(`num_dec_pic_size_in_luma_samples_minus1(616)`)에 의해 표현될 수 있다. "minus1"은 그 선택 요소의 값의 해석을 지칭할 수 있다. 예를 들어, 코딩된 값이 0이면, 하나의 테이블 엔트리가 존재한다. 값이 5이면, 6개의 테이블 엔트리가 존재한다. 테이블의 각각의 "라인"에 대해, 디코딩된 픽처 폭 및 높이가 이후 선택(617)에 포함된다.

[0082] 제시된 테이블 엔트리들(617)은 타일 그룹 헤더의 선택 요소 `dec_pic_size_idx(602)`를 사용하여 인덱싱될 수 있고, 그에 의해 타일 그룹당 상이한 디코딩된 크기들 - 실제로는, 줌 팩터들 - 을 허용한다.

[0083] 특정 비디오 코딩 기술들 또는 표준들, 예를 들어, VP9는, 공간적 스케일러빌리티(*spatial scalability*)를 가능하게 하기 위해, 시간적 스케일러빌리티(*temporal scalability*)와 함께 특정 형태들의 참조 픽처 리샘플링(개시된 주제와 매우 상이하게 시그널링됨)을 구현함으로써 공간적 스케일러빌리티를 지원한다. 특히, 특정 참조 픽처들은 공간적 향상 계층의 베이스를 형성하기 위해 ARC-스타일 기술들을 사용하여 더 높은 해상도로 업샘플링될 수 있다. 이러한 업샘플링된 픽처들은, 상세사항을 추가하기 위해, 고해상도에서 정상 예측 메커니즘들을 사용하여 리파인(refine)될 수 있다.

[0084] 본 명세서에서 논의된 실시예들은 이러한 환경에서 사용될 수 있다. 특정 경우들에서, 동일한 또는 다른 실시예에서, NAL 유닛 헤더 내의 값, 예를 들어, Temporal ID 필드는 시간적뿐만 아니라 공간적 계층도 나타내기 위해 사용될 수 있다. 그렇게 하는 것은 특정 시스템 설계들에 대해 특정 이점들을 가질 수 있으며; 예를 들어, NAL 유닛 헤더 Temporal ID 값에 기초하여 시간적 계층 선택된 포워딩에 대해 생성되고 최적화된 기존의 선택된 포워딩 유닛(Selected Forwarding Unit, SFU)은 스케일러블 환경들에 대해 수정 없이 사용될 수 있다. 이를 가능하게 하기 위해, 코딩된 픽처 크기와 시간적 계층 사이의 맵핑에 대해, NAL 유닛 헤더 내의 temporal ID 필드에 의해 나타내는 요건이 있을 수 있다.

[0085] 일부 비디오 코딩 기술들에서, 액세스 유닛(Access Unit, AU)은, 주어진 시간 인스턴스에서 캡처되어 각자의 픽처, 슬라이스, 타일, 및/또는 NAL 유닛 비트스트림으로 합성(compose)되는, 코딩된 픽처(들), 슬라이스(들), 타일(들), NAL 유닛(들) 등을 지칭할 수 있다. 그 시간 인스턴스는, 예를 들어, 합성 시간(*composition time*)일 수 있다.

[0086] HEVC 및 특정한 다른 비디오 코딩 기술들에서는, 디코딩된 픽처 버퍼(DPB)에 저장된 다수의 참조 픽처 중에서 선택된 참조 픽처를 표시하기 위해 픽처 순서 카운트(*picture order count*, POC) 값이 사용될 수 있다. 액세스 유닛(AU)이 하나 이상의 픽처, 슬라이스, 또는 타일을 포함할 때, 동일한 AU에 속하는 각각의 픽처, 슬라이스, 또는 타일은 동일한 POC 값을 운반(carry)할 수 있고, 이 값으로부터 이들이 동일한 합성 시간의 콘텐츠로부터 생성되었다는 것이 도출될 수 있다. 즉, 2개의 픽처/슬라이스/타일이 동일한 주어진 POC 값을 운반하는 시나리오에서, 이는 2개의 픽처/슬라이스/타일이 동일한 AU에 속하고 동일한 합성 시간을 갖는다는 것을 표시할 수 있다. 반대로, 2개의 픽처/타일/슬라이스가 상이한 POC 값들을 갖는다는 것은 그러한 픽처들/슬라이스들/타일들이 상이한 AU들에 속하고 상이한 합성 시간들을 갖는다는 것을 표시할 수 있다.

[0087] 실시예들에서, 이러한 엄격한 관계는 액세스 유닛이 상이한 POC 값들을 갖는 픽처들, 슬라이스들, 또는 타일들을 포함할 수 있다는 점에서 완화될 수 있다. AU 내에서 상이한 POC 값들을 허용함으로써, POC 값을 사용하여 동일한 제시 시간을 갖는 잠재적으로 독립적으로 디코딩가능한 픽처들/슬라이스들/타일들을 식별하는 것이 가능해진다. 이는, 결국, 아래에 더 상세히 설명되는 바와 같은, 참조 픽처 선택 시그널링, 예를 들어, 참조 픽처

세트 시그널링 또는 참조 픽처 리스트 시그널링의 변경 없이 다수의 스케일러블 계층들의 지원을 가능하게 할 수 있다.

- [0088] 그러나, POC 값 단독으로부터, 상이한 POC 값들을 갖는 다른 픽처들/슬라이스들/타일들에 대해, 픽처/슬라이스/타일이 속하는 AU를 식별할 수 있는 것이 여전히 바람직하다. 이는 아래에 설명되는 바와 같이 달성될 수 있다.
- [0089] 실시예들에서, NAL 유닛 헤더, 슬라이스 헤더, 타일 그룹 헤더, SEI 메시지, 파라미터 세트 또는 AU 구분자(delimiter)와 같은 고레벨 신택스 구조에서 액세스 유닛 카운트(access unit count, AUC)가 시그널링될 수 있다. AUC의 값은 어느 NAL 유닛, 픽처, 슬라이스, 또는 타일이 주어진 AU에 속하는지를 식별하는 데 사용될 수 있다. AUC의 값은 별개의 합성 시간 인스턴스에 대응할 수 있다. AUC 값은 POC 값의 배수와 동일할 수 있다. POC 값을 정수 값으로 나눔으로써, AUC 값이 계산될 수 있다. 특정 경우들에서, 나눗셈 연산들은 디코더 구현들에 특정 부담을 줄 수 있다. 이러한 경우에, AUC 값들의 넘버링 공간에서의 작은 제한들은 나눗셈 연산을 시프트 연산(shift operation)들로 대체하는 것을 허용할 수 있다. 예를 들어, AUC 값은 POC 값 범위의 최상위 비트(Most Significant Bit, MSB) 값과 동일할 수 있다.
- [0090] 실시예들에서, NAL 유닛 헤더, 슬라이스 헤더, 타일 그룹 헤더, SEI 메시지, 파라미터 세트 또는 AU 구분자와 같은 고레벨 신택스 구조에서 AU당 POC 사이클(poc_cycle_au)의 값이 시그널링될 수 있다. poc_cycle_au는 얼마나 많은 상이하고 연속적인 POC 값들이 동일한 AU와 연관될 수 있는지를 표시할 수 있다. 예를 들어, poc_cycle_au의 값이 4인 경우, POC 값이 0으로부터 3까지인 픽처들, 슬라이스들 또는 타일들은 AUC 값이 0인 AU와 연관될 수 있고, POC 값이 4로부터 7까지인 픽처들, 슬라이스들 또는 타일들은 AUC 값이 1인 AU와 연관될 수 있다. 따라서, AUC의 값은 POC 값을 poc_cycle_au의 값으로 나눔으로써 추론될 수 있다.
- [0091] 실시예들에서, poc_cyle_au의 값은 코딩된 비디오 시퀀스에서 공간적 또는 SNR 계층들의 수를 식별하는, 예를 들어, 비디오 파라미터 세트(video parameter set, VPS)에 위치된, 정보로부터 도출될 수 있다. 이러한 가능한 관계의 예에 대해 이하에서 간략히 설명한다. 위에서 설명한 바와 같은 도출은 VPS에서 몇 비트를 세이브할 수 있고 따라서 코딩 효율을 개선할 수 있지만, 일부 실시예들에서 poc_cycle_au는 계층적으로 비디오 파라미터 세트 아래의 적절한 고레벨 신택스 구조로 명시적으로 코딩될 수 있으므로, 픽처와 같은 비트스트림의 주어진 작은 부분에 대해 poc_cycle_au를 최소화할 수 있다. 이러한 최적화는 POC 값들, 및/또는 POC를 간접적으로 참조하는 신택스 요소들의 값들이 저레벨 신택스 구조들로 코딩될 수 있기 때문에 위의 도출 프로세스를 통해 세이브될 수 있는 것보다 더 많은 비트들을 세이브할 수 있다.
- [0092] 실시예들에서, 도 8은, 코딩된 비디오 시퀀스에서의 모든 픽처/슬라이스에 사용되는 poc_cycle_au를 표시하는 VPS(또는 SPS)에서의 vps_poc_cycle_au의 신택스 요소와, 슬라이스 헤더에서의, 현재 슬라이스의 poc_cycle_au를 표시하는 slice_poc_cycle_au의 신택스 요소를 시그널링하기 위한 신택스 테이블들의 예를 도시한다. AU당 POC 값이 균일하게 증가하는 경우, VPS에서의 vps_contant_poc_cycle_per_au는 1과 동일하게 설정될 수 있고 vps_poc_cycle_au는 VPS에서 시그널링될 수 있다. 이 경우, slice_poc_cycle_au는 명시적으로 시그널링되지 않을 수 있고, 각각의 AU에 대한 AUC의 값은 POC의 값을 vps_poc_cycle_au로 나눔으로써 계산될 수 있다. AU당 POC 값이 균일하게 증가하지 않는 경우, VPS에서의 vps_contant_poc_cycle_per_au는 0과 동일하게 설정될 수 있다. 이 경우, slice_access_unit_cnt는 각각의 슬라이스 또는 픽처에 대해 슬라이스 헤더에서 시그널링될 수 있지만, vps_access_unit_cnt는 시그널링되지 않을 수 있다. 각각의 슬라이스 또는 픽처는 slice_access_unit_cnt의 상이한 값을 가질 수 있다. 각각의 AU에 대한 AUC의 값은 POC의 값을 slice_poc_cycle_au로 나눔으로써 계산될 수 있다.
- [0093] 도 9는 위의 프로세스의 예를 예시하는 블록도를 도시한다. 예를 들어, 동작 S910에서, VPS(또는 SPS)가 파싱될 수 있고, 동작 S920에서 AU당 POC 사이클이 코딩된 비디오 시퀀스 내에서 일정한지가 결정될 수 있다. AU당 POC 사이클이 일정하면(동작 S920에서 예), 동작 S930에서 코딩된 비디오 시퀀스에 대해 시그널링된 poc_cycle_au 및 특정 액세스 유닛의 POC 값으로부터 특정 액세스 유닛에 대한 액세스 유닛 카운트의 값이 계산될 수 있다. AU당 POC 사이클이 일정하지 않으면(동작 S920에서 아니오), 동작 S940에서 픽처 레벨에서 시그널링된 poc_cycle_au 및 특정 액세스 유닛의 POC 값으로부터 특정 액세스 유닛에 대한 액세스 유닛 카운트의 값이 계산될 수 있다. 동작 S950에서, 새로운 VPS(또는 SPS)가 파싱될 수 있다.
- [0094] 실시예들에서, 픽처, 슬라이스, 또는 타일의 POC의 값이 상이할 수 있더라도, 동일한 AUC 값을 갖는 AU에 대응하는 픽처, 슬라이스, 또는 타일은 동일한 디코딩 또는 출력 시간 인스턴스와 연관될 수 있다. 따라서, 동일한 AU에서의 픽처들, 슬라이스들 또는 타일들에 걸친 임의의 인터-파싱/디코딩 종속성(inter-parsing/decoding

dependency) 없이, 동일한 AU와 연관된 픽처들, 슬라이스들 또는 타일들의 전부 또는 서브세트가 병렬로 디코딩 될 수 있고, 동일한 시간 인스턴스에서 출력될 수 있다.

- [0095] 실시예들에서, 픽처, 슬라이스, 또는 타일에 대한 POC의 값이 상이할 수 있더라도, 동일한 AUC 값을 갖는 AU에 대응하는 픽처, 슬라이스, 또는 타일은 동일한 합성/디스플레이 시간 인스턴스와 연관될 수 있다. 합성 시간이 컨테이너 포맷으로 포함될 때, 픽처들이 상이한 AU들에 대응하더라도, 픽처들이 동일한 합성 시간을 갖는다면, 픽처들은 동일한 시간 인스턴스에 디스플레이될 수 있다.
- [0096] 실시예들에서, 각각의 픽처, 슬라이스, 또는 타일은 동일한 AU에서 동일한 시간적 식별자(temporal_id)를 가질 수 있다. 시간 인스턴스에 대응하는 픽처들, 슬라이스들 또는 타일들의 전부 또는 서브세트는 동일한 시간적 서브-계층과 연관될 수 있다. 실시예들에서, 각각의 픽처, 슬라이스, 또는 타일은 동일한 AU에서 동일한 또는 상이한 공간적 계층 id(layer_id)를 가질 수 있다. 시간 인스턴스에 대응하는 픽처들, 슬라이스들 또는 타일들의 전부 또는 서브세트는 동일한 또는 상이한 공간적 계층과 연관될 수 있다.
- [0097] 도 7은 적응적 해상도 변경을 갖는 temporal_id, layer_id, POC 및 AUC 값들의 조합을 갖는 비디오 시퀀스 구조의 예를 도시한다. 이 예에서, AUC=0인 제1 AU에서의 픽처, 슬라이스 또는 타일은 temporal_id=0 및 layer_id=0 또는 1을 가질 수 있고, AUC=1인 제2 AU에서의 픽처, 슬라이스 또는 타일은 temporal_id=1 및 layer_id=0 또는 1을 각각 가질 수 있다. POC의 값은 temporal_id 및 layer_id의 값들에 관계없이 픽처당 1만큼 증가된다. 이 예에서, poc_cycle_au의 값은 2일 수 있다. 실시예들에서, poc_cycle_au의 값은 (공간적 스케일러빌리티) 계층들의 수와 동일하게 설정될 수 있다. 따라서, 이 예에서, POC의 값은 2만큼 증가되고, AUC의 값은 1만큼 증가된다.
- [0098] 위의 실시예들에서, 인터-픽처 또는 인터-계층 예측 구조 및 참조 픽처 표시의 전부 또는 서브-세트는 HEVC에서의 기존의 참조 픽처 세트(RPS) 시그널링 또는 참조 픽처 리스트(RPL) 시그널링을 사용하여 지원될 수 있다. RPS 또는 RPL에서, 선택된 참조 픽처는 현재 픽처와 선택된 참조 픽처 사이의 POC의 델타 값 또는 POC의 값을 시그널링함으로써 표시될 수 있다. 실시예들에서, RPS 및 RPL은 시그널링의 변경 없이 인터-픽처 또는 인터-계층 예측 구조를 표시하는 데 사용될 수 있지만, 다음과 같은 제한들이 있다. 참조 픽처의 temporal_id의 값이 현재 픽처의 temporal_id의 값보다 큰 경우, 현재 픽처는 모션 보상 또는 다른 예측들을 위해 참조 픽처를 사용하지 않을 수 있다. 참조 픽처의 layer_id의 값이 현재 픽처의 layer_id의 값보다 큰 경우, 현재 픽처는 모션 보상 또는 다른 예측들을 위해 참조 픽처를 사용하지 않을 수 있다.
- [0099] 실시예들에서, 시간적 모션 벡터 예측을 위한 POC 차이에 기초한 모션 벡터 스케일링은 액세스 유닛 내의 다수의 픽처들에 걸쳐 디스에이블될 수 있다. 따라서, 각각의 픽처가 액세스 유닛 내에서 상이한 POC 값을 가질 수 있지만, 모션 벡터는 액세스 유닛 내에서 시간적 모션 벡터 예측을 위해 스케일링 및 사용되지 않는다. 이는, 동일한 AU에서의 상이한 POC를 갖는 참조 픽처가 동일한 시간 인스턴스를 갖는 참조 픽처로 간주되기 때문이다. 따라서, 실시예에서, 참조 픽처가 현재 픽처와 연관된 AU에 속할 때, 모션 벡터 스케일링 함수는 1을 리턴할 수 있다.
- [0100] 실시예들에서, 참조 픽처의 공간적 해상도가 현재 픽처의 공간적 해상도와 상이할 때, 시간적 모션 벡터 예측을 위한 POC 차이에 기초한 모션 벡터 스케일링은 다수의 픽처들에 걸쳐 선택적으로 디스에이블될 수 있다. 모션 벡터 스케일링이 허용될 때, 모션 벡터는 현재 픽처와 참조 픽처 사이의 공간적 해상도 비율과 POC 차이 양자 모두에 기초하여 스케일링된다.
- [0101] 실시예들에서, 특히 poc_cycle_au가 불균일한 값을 가질 때(예를 들어, vps_contant_poc_cycle_per_au==0일 때), 시간적 모션 벡터 예측을 위해, 모션 벡터는 POC 차이 대신에 AUC 차이에 기초하여 스케일링될 수 있다. 그렇지 않으면(예를 들어, vps_contant_poc_cycle_per_au==1일 때), AUC 차이에 기초한 모션 벡터 스케일링은 POC 차이에 기초한 모션 벡터 스케일링과 동일할 수 있다.
- [0102] 실시예들에서, 모션 벡터가 AUC 차이에 기초하여 스케일링될 때, 현재 픽처와 동일한 AU(동일한 AUC 값을 가짐)에서의 참조 모션 벡터는 AUC 차이에 기초하여 스케일링되지 않고, 현재 픽처와 참조 픽처 사이의 공간적 해상도 비율에 기초한 스케일링과 함께 또는 스케일링 없이 모션 벡터 예측을 위해 사용된다.
- [0103] 실시예들에서, AUC 값은 AU의 경계를 식별하기 위해 사용될 수 있고, AU 입도를 갖는 입력 및 출력 타이밍을 필요로 하는 HRD(hypothetical reference decoder) 연산을 위해 사용될 수 있다. 실시예들에서, AU에서의 최고 계층을 갖는 디코딩된 픽처가 디스플레이를 위해 출력될 수 있다. AUC 값 및 layer_id 값은 출력 픽처를 식별하기 위해 사용될 수 있다.

- [0104] 실시예들에서, 픽처는 하나 이상의 서브-픽처를 포함할 수 있다. 각각의 서브-픽처는 픽처의 로컬 영역 또는 전체 영역을 커버할 수 있다. 서브-픽처에 의해 지원되는 영역은 다른 서브-픽처에 의해 지원되는 영역과 중첩될 수 있거나 중첩되지 않을 수 있다. 하나 이상의 서브-픽처에 의해 커버되는 영역은 픽처의 전체 영역을 커버할 수 있거나 커버하지 않을 수 있다. 픽처가 서브-픽처를 포함하는 경우, 서브-픽처에 의해 지원되는 영역은 픽처에 의해 지원되는 영역과 동일할 수 있다.
- [0105] 실시예들에서, 서브-픽처는 코딩된 픽처에 사용되는 코딩 방법과 유사한 코딩 방법에 의해 코딩될 수 있다. 서브-픽처는 독립적으로 코딩될 수 있거나 다른 서브-픽처 또는 코딩된 픽처에 따라 코딩될 수 있다. 서브-픽처는 다른 서브-픽처 또는 코딩된 픽처로부터 임의의 과성 종속성을 가질 수 있거나 갖지 않을 수 있다.
- [0106] 실시예들에서, 코딩된 서브-픽처는 하나 이상의 계층에 포함될 수 있다. 계층에서의 코딩된 서브-픽처는 상이한 공간적 해상도를 가질 수 있다. 원래의 서브-픽처는 공간적으로 리샘플링(예를 들어, 업샘플링 또는 다운샘플링)되고, 상이한 공간적 해상도 파라미터들로 코딩되고, 계층에 대응하는 비트스트림에 포함될 수 있다.
- [0107] 실시예들에서, (W, H)(여기서, W는 서브-픽처의 폭을 표시하고 H는 서브-픽처의 높이를 각각 표시함)를 갖는 서브-픽처는 코딩되어 계층 0에 대응하는 코딩된 비트스트림에 포함될 수 있고, (W*S_{w,k}, H*S_{h,k})를 갖는, 원래의 공간적 해상도를 갖는 서브-픽처로부터 업샘플링된(또는 다운샘플링된) 서브-픽처는 코딩되어 계층 k에 대응하는 코딩된 비트스트림에 포함될 수 있으며, 여기서 S_{w,k}, S_{h,k}는 수평으로 및 수직으로 리샘플링 비율들을 표시한다. S_{w,k}, S_{h,k}의 값들이 1보다 크면, 리샘플링은 업샘플링일 수 있다. 반면, S_{w,k}, S_{h,k}의 값들이 1보다 작으면, 리샘플링은 다운샘플링일 수 있다.
- [0108] 실시예들에서, 계층에서의 코딩된 서브-픽처는 동일한 서브-픽처 또는 상이한 서브-픽처의 다른 계층에서의 코딩된 서브-픽처와 상이한 시각적 품질을 가질 수 있다. 예를 들어, 계층 n에서의 서브-픽처 i는 양자화 파라미터 Q_{i,n}로 코딩될 수 있고, 계층 m에서의 서브-픽처 j는 양자화 파라미터 Q_{j,m}로 코딩될 수 있다.
- [0109] 실시예들에서, 계층에서의 코딩된 서브-픽처는 동일한 로컬 영역의 다른 계층에서의 코딩된 서브-픽처로부터의 임의의 과성 또는 디코딩 종속성 없이 독립적으로 디코딩가능할 수 있다. 동일한 로컬 영역의 다른 서브-픽처 계층을 참조하지 않고 독립적으로 디코딩가능할 수 있는 서브-픽처 계층은 독립적인 서브-픽처 계층일 수 있다. 독립적인 서브-픽처 계층에서의 코딩된 서브-픽처는 동일한 서브-픽처 계층에서의 이전에 코딩된 서브-픽처로부터의 디코딩 또는 과성 종속성을 가질 수 있거나 갖지 않을 수 있지만, 코딩된 서브-픽처는 다른 서브-픽처 계층에서의 코딩된 픽처로부터의 임의의 종속성을 갖지 않을 수 있다.
- [0110] 실시예들에서, 계층에서의 코딩된 서브-픽처는, 동일한 로컬 영역의 다른 계층에서의 코딩된 서브-픽처로부터의 임의의 과성 또는 디코딩 종속성을 갖고서, 종속적으로 디코딩가능할 수 있다. 동일한 로컬 영역의 다른 서브-픽처 계층을 참조하여 종속적으로 디코딩가능할 수 있는 서브-픽처 계층은 종속적인 서브-픽처 계층일 수 있다. 종속적인 서브-픽처에서의 코딩된 서브-픽처는 동일한 서브-픽처에 속하는 코딩된 서브-픽처를 참조할 수 있거나, 동일한 서브-픽처 계층에서의 이전에 코딩된 서브-픽처를 참조할 수 있거나, 또는 양자 모두의 서브-픽처들을 참조할 수 있다.
- [0111] 실시예들에서, 코딩된 서브-픽처는 하나 이상의 독립적인 서브-픽처 계층 및 하나 이상의 종속적인 서브-픽처 계층을 포함할 수 있다. 그러나, 코딩된 서브-픽처에 대해 적어도 하나의 독립적인 서브-픽처 계층이 존재할 수 있다. 독립적인 서브-픽처 계층의, NAL 유닛 헤더 또는 다른 고레벨 신텍스 구조에 존재할 수 있는, 계층 식별자(layer_id)의 값은 0일 수 있다. layer_id가 0인 서브-픽처 계층은 베이스 서브-픽처 계층일 수 있다.
- [0112] 실시예들에서, 픽처는 하나 이상의 전경 서브-픽처 및 하나의 배경 서브-픽처를 포함할 수 있다. 배경 서브-픽처에 의해 지원되는 영역은 픽처의 영역과 동일할 수 있다. 전경 서브-픽처에 의해 지원되는 영역은 배경 서브-픽처에 의해 지원되는 영역과 중첩될 수 있다. 배경 서브-픽처는 베이스 서브-픽처 계층일 수 있고, 전경 서브-픽처는 비-베이스(향상) 서브-픽처 계층일 수 있다. 하나 이상의 비-베이스 서브-픽처 계층은 디코딩을 위해 동일한 베이스 계층을 참조할 수 있다. layer_id가 a인 각각의 비-베이스 서브-픽처 계층은 layer_id가 b인 비-베이스 서브-픽처 계층을 참조할 수 있으며, 여기서 a는 b보다 크다.
- [0113] 실시예들에서, 픽처는 배경 서브-픽처를 갖거나 갖지 않고서 하나 이상의 전경 서브-픽처를 포함할 수 있다. 각각의 서브-픽처는 그 자신의 베이스 서브-픽처 계층 및 하나 이상의 비-베이스(향상) 계층을 가질 수 있다. 각각의 베이스 서브-픽처 계층은 하나 이상의 비-베이스 서브-픽처 계층에 의해 참조될 수 있다. layer_id가 a인 각각의 비-베이스 서브-픽처 계층은 layer_id가 b인 비-베이스 서브-픽처 계층을 참조할 수 있으며, 여기서

a는 b보다 크다.

- [0114] 실시예들에서, 픽처는 배경 서브-픽처를 갖거나 갖지 않고서 하나 이상의 전경 서브-픽처를 포함할 수 있다. (베이스 또는 비-베이스) 서브-픽처 계층에서의 각각의 코딩된 서브-픽처는 동일한 서브-픽처에 속하는 하나 이상의 비-베이스 계층 서브-픽처 및 동일한 서브-픽처에 속하지 않는 하나 이상의 비-베이스 계층 서브-픽처에 의해 참조될 수 있다.
- [0115] 실시예들에서, 픽처는 배경 서브-픽처를 갖거나 갖지 않고서 하나 이상의 전경 서브-픽처를 포함할 수 있다. 계층 a에서의 서브-픽처는 동일한 계층에서의 다수의 서브-픽처로 더 파티셔닝될 수 있다. 계층 b에서의 하나 이상의 코딩된 서브-픽처는 계층 a에서의 파티셔닝된 서브-픽처를 참조할 수 있다.
- [0116] 실시예들에서, 코딩된 비디오 시퀀스(CVS)는 코딩된 픽처들의 그룹일 수 있다. CVS는 하나 이상의 코딩된 서브-픽처 시퀀스(coded sub-picture sequence, CSPS)를 포함할 수 있고, 여기서 CSPS는 픽처의 동일한 로컬 영역을 커버하는 코딩된 서브-픽처 그룹일 수 있다. CSPS는 코딩된 비디오 시퀀스와 동일하거나 상이한 시간적 해상도를 가질 수 있다.
- [0117] 실시예들에서, CSPS는 코딩되어 하나 이상의 계층에 포함될 수 있다. CSPS는 하나 이상의 CSPS 계층을 포함할 수 있다. CSPS에 대응하는 하나 이상의 CSPS 계층을 디코딩하는 것은 동일한 로컬 영역에 대응하는 서브-픽처들의 시퀀스를 재구성할 수 있다.
- [0118] 실시예들에서, CSPS에 대응하는 CSPS 계층들의 수는 다른 CSPS에 대응하는 CSPS 계층들의 수와 동일하거나 상이할 수 있다.
- [0119] 실시예들에서, CSPS 계층은 다른 CSPS 계층과 상이한 시간적 해상도(예를 들어, 프레임 레이트)를 가질 수 있다. 원래의 (압축되지 않은) 서브-픽처 시퀀스는 시간적으로 리샘플링(예를 들어, 업샘플링 또는 다운샘플링)되고, 상이한 시간적 해상도 파라미터들로 코딩되고, 계층에 대응하는 비트스트림에 포함될 수 있다.
- [0120] 실시예들에서, 프레임 레이트 F를 갖는 서브-픽처 시퀀스는 코딩되어 계층 0에 대응하는 코딩된 비트스트림에 포함될 수 있고, $F \cdot S_{t,k}$ 를 갖는, 원래의 서브-픽처 시퀀스로부터의 시간적으로 업샘플링된(또는 다운샘플링된) 서브-픽처 시퀀스는 코딩되어 계층 k에 대응하는 코딩된 비트스트림에 포함될 수 있으며, 여기서 $S_{t,k}$ 는 계층 k에 대한 시간적 샘플링 비율을 표시한다. $S_{t,k}$ 의 값이 1보다 큰 경우, 시간적 리샘플링 프로세스는 프레임 레이트 상향 변환(frame rate up conversion)일 수 있다. 반면, $S_{t,k}$ 의 값이 1보다 작은 경우, 시간적 리샘플링 프로세스는 프레임 레이트 하향 변환(frame rate down conversion)일 수 있다.
- [0121] 실시예들에서, CSPS 계층 a를 갖는 서브-픽처가 모션 보상 또는 임의의 인터-계층 예측을 위해 CSPS 계층 b를 갖는 서브-픽처에 의해 참조될 때, CSPS 계층 a의 공간적 해상도가 CSPS 계층 b의 공간적 해상도와 상이한 경우, CSPS 계층 a에서의 디코딩된 픽셀들이 리샘플링되어 참조를 위해 사용된다. 리샘플링 프로세스는 업샘플링 필터링 또는 다운샘플링 필터링을 사용할 수 있다.
- [0122] 도 10은 layer_id가 0인 배경 비디오 CSPS 및 다수의 전경 CSPS 계층을 포함하는 예시적인 비디오 스트림을 도시한다. 코딩된 서브-픽처는 하나 이상의 CSPS 계층을 포함할 수 있지만, 임의의 전경 CSPS 계층에 속하지 않는 배경 영역은 베이스 계층을 포함할 수 있다. 베이스 계층은 배경 영역 및 전경 영역들을 포함할 수 있고, 항상 CSPS 계층은 전경 영역을 포함할 수 있다. 항상 CSPS 계층은 동일한 영역에서 베이스 계층보다 양호한 시각적 품질을 가질 수 있다. 항상 CSPS 계층은 동일한 영역에 대응하는 베이스 계층의 모션 벡터들 및 재구성된 픽셀들을 참조할 수 있다.
- [0123] 실시예들에서, 베이스 계층에 대응하는 비디오 비트스트림은 트랙(track)에 포함되고, 각각의 서브-픽처에 대응하는 CSPS 계층들은 비디오 파일에서 분리된 트랙에 포함된다.
- [0124] 실시예들에서, 베이스 계층에 대응하는 비디오 비트스트림은 트랙에 포함되고, 동일한 layer_id를 갖는 CSPS 계층들은 분리된 트랙에 포함된다. 이 예에서, 계층 k에 대응하는 트랙은 계층 k에 대응하는 CSPS 계층들만을 포함한다.
- [0125] 실시예들에서, 각각의 서브-픽처의 각각의 CSPS 계층은 분리된 트랙에 저장된다. 각각의 트랙은 하나 이상의 다른 트랙으로부터의 임의의 파싱 또는 디코딩 종속성을 가질 수 있거나 갖지 않을 수 있다.

- [0126] 실시예들에서, 각각의 트랙은 서브-픽처들의 전부 또는 서브세트의 CSPS 계층들의 계층 i 내지 계층 j 에 대응하는 비트스트림들을 포함할 수 있으며, 여기서 $0 < i \leq j \leq k$ 이고, k 는 CSPS의 최고 계층이다.
- [0127] 실시예들에서, 픽처는 깊이 맵, 알파 맵, 3D 지오메트리 데이터, 점유 맵 등을 포함하는 하나 이상의 연관된 미디어 데이터를 포함한다. 이러한 연관된 시간 설정형 미디어 데이터(timed media data)는 각각이 하나의 서브-픽처에 대응하는 하나 또는 다수의 데이터 서브스트림으로 나눌 수 있다.
- [0128] 도 11은 다중-계층형 서브-픽처 방법(multi-layered sub-picture method)에 기초한 영상 회의의 예를 도시한다. 비디오 스트림에는, 배경 픽처에 대응하는 하나의 베이스 계층 비디오 비트스트림 및 전경 서브-픽처들에 대응하는 하나 이상의 향상 계층 비디오 비트스트림이 포함된다. 각각의 향상 계층 비디오 비트스트림은 CSPS 계층에 대응할 수 있다. 디스플레이에서, 베이스 계층에 대응하는 픽처가 디폴트로 디스플레이된다. 이는 하나 이상의 사용자의 PIP(picture in a picture)를 포함한다. 특정 사용자가 클라이언트의 제어에 의해 선택될 때, 선택된 사용자에게 대응하는 향상 CSPS 계층은 향상된 품질 또는 공간적 해상도로 디코딩되고 디스플레이될 수 있다.
- [0129] 도 12는 위의 프로세스의 예를 예시하는 블록도를 도시한다. 예를 들어, 동작 S1210에서, 다수의 계층들을 갖는 비디오 비트스트림이 디코딩될 수 있다. 동작 S1220에서, 배경 영역 및 하나 이상의 전경 서브-픽처가 식별될 수 있다. 동작 S1230에서, 특정 서브-픽처 영역, 예를 들어, 전경 서브-픽처들 중 하나가 선택되는지가 결정될 수 있다. 특정 서브-픽처 영역이 선택되면(동작 S1240에서 예), 향상된 서브-픽처가 디코딩되고 디스플레이될 수 있다. 특정 서브-픽처 영역이 선택되지 않으면(동작 S1240에서 아니오), 배경 영역이 디코딩되고 디스플레이될 수 있다.
- [0130] 실시예들에서, 네트워크 미들 박스(예를 들어, 라우터)는 그 대역폭에 따라 사용자에게 전송할 계층들의 서브셋을 선택할 수 있다. 픽처/서브픽처 편성(picture/subpicture organization)은 대역폭 적응을 위해 사용될 수 있다. 예를 들어, 사용자가 대역폭을 갖지 않는 경우, 라우터는 중요도로 인해 또는 사용된 설정에 기초하여 계층들을 제거하거나 일부 서브픽처들을 선택하며 이것은 대역폭에 적응하도록 동적으로 행해질 수 있다.
- [0131] 도 13은 360 비디오의 사용 사례에 관한 실시예를 도시한다. 구형 360 픽처, 예를 들어, 픽처 1310이 평면 픽처 상에 투영될 때, 투영 360 픽처는 베이스 계층으로서 다수의 서브-픽처로 파티셔닝될 수 있다. 예를 들어, 다수의 서브-픽처는 뒤쪽 서브-픽처, 상단 서브-픽처, 우측 서브-픽처, 좌측 서브-픽처, 앞쪽 서브-픽처, 및 하단 서브-픽처를 포함할 수 있다. 특정 서브-픽처, 예를 들어, 앞쪽 서브-픽처의 향상 계층이 코딩되어 클라이언트에 송신될 수 있다. 디코더는 모든 서브-픽처들을 포함하는 베이스 계층과 선택된 서브-픽처의 향상 계층 양자 모두를 디코딩가능할 수 있다. 현재 뷰포트가 선택된 서브-픽처와 동일할 때, 디스플레이된 픽처는 향상 계층을 갖는 디코딩된 서브-픽처로 더 높은 품질을 가질 수 있다. 그렇지 않으면, 베이스 계층을 갖는 디코딩된 픽처가 더 낮은 품질로 디스플레이될 수 있다.
- [0132] 실시예들에서, 디스플레이를 위한 임의의 레이아웃 정보는 파일 내에, (SEI 메시지 또는 메타데이터와 같은) 보충 정보로서 존재할 수 있다. 하나 이상의 디코딩된 서브-픽처는 시그널링된 레이아웃 정보에 따라 재배치되어 디스플레이될 수 있다. 레이아웃 정보는 스트리밍 서버 또는 브로드캐스터(broadcaster)에 의해 시그널링될 수 있거나, 네트워크 엔티티 또는 클라우드 서버에 의해 재생성될 수 있거나, 사용자의 맞춤형된 설정에 의해 결정될 수 있다.
- [0133] 실시예들에서, 입력 픽처가 하나 이상의 (직사각형) 서브-영역(들)으로 분할될 때, 각각의 서브-영역은 독립적인 계층으로서 코딩될 수 있다. 로컬 영역에 대응하는 각각의 독립적인 계층은 고유 layer_id 값을 가질 수 있다. 각각의 독립적인 계층에 대해, 서브-픽처 크기 및 위치 정보가 시그널링될 수 있다. 예를 들어, 픽처 크기(폭, 높이), 좌측-상단 코너의 오프셋 정보(x_offset, y_offset). 도 14는 분할된 서브-픽처들의 레이아웃, 그의 서브-픽처 크기 및 위치 정보 및 그의 대응하는 픽처 예측 구조의 예를 도시한다. 서브-픽처 크기(들) 및 서브-픽처 위치(들)를 포함하는 레이아웃 정보는 파라미터 세트(들), 슬라이스 또는 타일 그룹의 헤더, 또는 SEI 메시지와 같은 고레벨 신택스 구조에서 시그널링될 수 있다.
- [0134] 실시예들에서, 독립적인 계층에 대응하는 각각의 서브-픽처는 AU 내의 그의 고유 POC 값을 가질 수 있다. DPB에 저장된 픽처들 중 참조 픽처가 RPS 또는 RPL 구조의 신택스 요소(들)를 사용하여 표시될 때, 계층에 대응하는 각각의 서브-픽처의 POC 값(들)이 사용될 수 있다.
- [0135] 실시예들에서, (인tra-계층) 예측 구조를 표시하기 위해, layer_id가 사용되지 않을 수 있고 POC(델타) 값이 사용될 수 있다.

- [0136] 실시예들에서, 계층(또는 로컬 영역)에 대응하는 POC 값이 N인 서브-픽처는 모션 보상된 예측을 위한 동일한 계층(또는 동일한 로컬 영역)에 대응하는 POC 값이 N+K인 서브-픽처의 참조 픽처로서 사용될 수 있거나 사용되지 않을 수 있다. 대부분의 경우에, 수 K의 값은 서브-영역들의 수와 동일할 수 있는 (독립적인) 계층들의 최대 수와 동일할 수 있다.
- [0137] 실시예들에서, 도 15는 도 14의 확장된 경우를 도시한다. 입력 픽처가 다수의(예를 들어, 4개) 서브-영역으로 분할될 때, 각각의 로컬 영역은 하나 이상의 계층으로 코딩될 수 있다. 이 경우, 독립적인 계층들의 수는 서브-영역들의 수와 동일할 수 있고, 하나 이상의 계층은 서브-영역에 대응할 수 있다. 따라서, 각각의 서브-영역은 하나 이상의 독립적인 계층(들) 및 0개 이상의 종속적인 계층(들)으로 코딩될 수 있다.
- [0138] 실시예들에서, 도 15에서는, 입력 픽처가 4개의 서브-영역으로 분할될 수 있다. 예로서, 우측-상단 서브-영역은 계층 1과 계층 4인 2개의 계층으로서 코딩될 수 있고, 우측-하단 서브-영역은 계층 3과 계층 5인 2개의 계층으로서 코딩될 수 있다. 이 경우, 계층 4는 모션 보상된 예측을 위해 계층 1을 참조할 수 있고, 계층 5는 모션 보상을 위해 계층 3을 참조할 수 있다.
- [0139] 실시예들에서, 계층 경계에 걸친 인-루프 필터링(디블록킹 필터링, 적응적 인-루프 필터링, 재성형기(reshaper), 쌍방 필터링 또는 임의의 딥 러닝 기반 필터링 등)이 (선택적으로) 디스에이블될 수 있다.
- [0140] 실시예들에서, 계층 경계에 걸친 모션 보상된 예측 또는 인트라-블록 카피가 (선택적으로) 디스에이블될 수 있다.
- [0141] 실시예들에서, 서브-픽처의 경계에서 모션 보상된 예측 또는 인-루프 필터링을 위한 경계 패딩은 선택적으로 처리될 수 있다. 경계 패딩이 처리되는지 여부를 표시하는 플래그가 파라미터 세트(들)(VPS, SPS, PPS, 또는 APS), 슬라이스 또는 타일 그룹 헤더, 또는 SEI 메시지와 같은 고레벨 신택스 구조에서 시그널링될 수 있다.
- [0142] 실시예들에서, 서브-영역(들)(또는 서브-픽처(들))의 레이아웃 정보는 VPS 또는 SPS에서 시그널링될 수 있다. 도 16a는 VPS에서의 신택스 요소들의 예를 도시하고, 도 16b는 SPS에서의 신택스 요소들의 예를 도시한다. 이 예에서, vps_sub_picture_dividing_flag는 VPS에서 시그널링된다. 플래그는 입력 픽처(들)가 다수의 서브-영역들로 분할되는지 여부를 표시할 수 있다. vps_sub_picture_dividing_flag의 값이 0일 때, 현재 VPS에 대응하는 코딩된 비디오 시퀀스(들)에서의 입력 픽처(들)는 다수의 서브-영역들로 분할되지 않을 수 있다. 이 경우, 입력 픽처 크기는 SPS에서 시그널링되는 코딩된 픽처 크기(pic_width_in_luma_samples, pic_height_in_luma_samples)와 동일할 수 있다. vps_sub_picture_dividing_flag의 값이 1일 때, 입력 픽처(들)는 다수의 서브-영역들로 분할될 수 있다. 이 경우, 신택스 요소들 vps_full_pic_width_in_luma_samples 및 vps_full_pic_height_in_luma_samples는 VPS에서 시그널링된다. vps_full_pic_width_in_luma_samples 및 vps_full_pic_height_in_luma_samples의 값들은 각각 입력 픽처(들)의 폭 및 높이와 동일할 수 있다.
- [0143] 실시예들에서, vps_full_pic_width_in_luma_samples 및 vps_full_pic_height_in_luma_samples의 값들은 디코딩을 위해 사용되지 않을 수 있지만, 합성 및 디스플레이를 위해 사용될 수 있다.
- [0144] 실시예들에서, vps_sub_picture_dividing_flag의 값이 1일 때, 신택스 요소들 pic_offset_x 및 pic_offset_y는 (하나의) 특정 계층(들)에 대응하는 SPS에서 시그널링될 수 있다. 이 경우, SPS에서 시그널링된 코딩된 픽처 크기(pic_width_in_luma_samples, pic_height_in_luma_samples)는 특정 계층에 대응하는 서브-영역의 폭 및 높이와 동일할 수 있다. 또한, 서브-영역의 좌측-상단 코너의 위치(pic_offset_x, pic_offset_y)는 SPS에서 시그널링될 수 있다.
- [0145] 실시예들에서, 서브-영역의 좌측-상단 코너의 위치 정보(pic_offset_x, pic_offset_y)는 디코딩을 위해 사용되지 않을 수 있지만, 합성 및 디스플레이를 위해 사용될 수 있다.
- [0146] 실시예들에서, (하나의) 입력 픽처(들)의 전부 또는 서브-세트 서브-영역(들)의 레이아웃 정보(크기 및 위치), 계층(들) 사이의 종속성 정보는 파라미터 세트 또는 SEI 메시지에서 시그널링될 수 있다. 도 17은 서브-영역들의 레이아웃의 정보, 계층들 사이의 종속성, 및 서브-영역과 하나 이상의 계층 사이의 관계를 표시하기 위한 신택스 요소들의 예를 도시한다. 이 예에서, 신택스 요소 num_sub_region은 현재 코딩된 비디오 시퀀스에서의 (직사각형) 서브-영역들의 수를 표시한다. 신택스 요소 num_layers는 현재 코딩된 비디오 시퀀스에서의 계층들의 수를 표시한다. num_layers의 값은 num_sub_region의 값과 동일하거나 그보다 클 수 있다. 임의의 서브-영역이 단일 계층으로서 코딩될 때, num_layers의 값은 num_sub_region의 값과 동일할 수 있다. 하나 이상의 서브-영역이 다수의 계층들로서 코딩될 때, num_layers의 값은 num_sub_region의 값보다 클 수 있다. 신택스 요

소 `direct_dependency_flag[i][j]`는 j 번째 계층으로부터 i 번째 계층으로의 종속성을 표시한다. `num_layers_for_region[i]`는 i 번째 서브-영역과 연관된 계층들의 수를 표시한다. `sub_region_layer_id[i][j]`는 i 번째 서브-영역과 연관된 j 번째 계층의 `layer_id`를 표시한다. `sub_region_offset_x[i]` 및 `sub_region_offset_y[i]`는 각각 i 번째 서브-영역의 좌측-상단 코너의 수평 및 수직 위치를 표시한다. `sub_region_width[i]` 및 `sub_region_height[i]`는 각각 i 번째 서브-영역의 폭 및 높이를 표시한다.

- [0147] 실시예들에서, 프로파일 티어 레벨 정보와 함께 또는 프로파일 티어 레벨 정보 없이 출력될 하나 이상의 계층을 표시하기 위해 출력 계층 세트를 지정하는 하나 이상의 선택스 요소는 고레벨 선택스 구조, 예를 들어, VPS, DPS, SPS, PPS, APS 또는 SEI 메시지에서 시그널링될 수 있다. 도 18을 참조하면, VPS를 참조하는 코딩된 비디오 시퀀스에서의 출력 계층 세트(output layer set, OLS)의 수를 표시하는 선택스 요소 `num_output_layer_sets`가 VPS에서 시그널링될 수 있다. 각각의 출력 계층 세트에 대해, `output_layer_flag`는 출력 계층들의 수만큼 시그널링될 수 있다.
- [0148] 실시예들에서, `output_layer_flag[i]`가 1인 것은 i 번째 계층이 출력된다는 것을 지정한다. `vps_output_layer_flag[i]`가 0인 것은 i 번째 계층이 출력되지 않는다는 것을 지정한다.
- [0149] 실시예들에서, 각각의 출력 계층 세트에 대한 프로파일 티어 레벨 정보를 지정하는 하나 이상의 선택스 요소는 고레벨 선택스 구조, 예를 들어, VPS, DPS, SPS, PPS, APS 또는 SEI 메시지에서 시그널링될 수 있다. 여전히 도 18을 참조하면, VPS를 참조하는 코딩된 비디오 시퀀스에서의 OLS당 프로파일 티어 레벨 정보의 수를 표시하는 선택스 요소 `num_profile_tile_level`은 VPS에서 시그널링될 수 있다. 각각의 출력 계층 세트에 대해, 프로파일 티어 레벨 정보에 대한 선택스 요소들의 세트 또는 프로파일 티어 레벨 정보에서의 엔트리들 중에서 특정 프로파일 티어 레벨 정보를 표시하는 인덱스가 출력 계층들의 수만큼 시그널링될 수 있다.
- [0150] 실시예들에서, `profile_tier_level_idx[i][j]`는 i 번째 OLS의 j 번째 계층에 적용되는 `profile_tier_level()` 선택스 구조의, VPS에서의 `profile_tier_level()` 선택스 구조들의 리스트로의, 인덱스를 지정한다.
- [0151] 실시예들에서, 도 19를 참조하면, 선택스 요소들 `num_profile_tile_level` 및/또는 `num_output_layer_sets`는 최대 계층 수가 1보다 클 때(`vps_max_layers_minus1 > 0`) 시그널링될 수 있다.
- [0152] 실시예들에서, 도 19를 참조하면, i 번째 출력 계층 세트에 대한 출력 계층 시그널링의 모드를 표시하는 선택스 요소 `vps_output_layers_mode[i]`는 VPS에 존재할 수 있다.
- [0153] 실시예들에서, `vps_output_layers_mode[i]`가 0인 것은 i 번째 출력 계층 세트이면서 최고 계층만이 출력된다는 것을 지정한다. `vps_output_layer_mode[i]`가 1인 것은 i 번째 출력 계층 세트인 모든 계층이 출력된다는 것을 지정한다. `vps_output_layer_mode[i]`가 2인 것은 출력되는 계층들이 i 번째 출력 계층 세트이면서 `vps_output_layer_flag[i][j]`가 1인 계층들인 것을 지정한다. 더 많은 값들이 예약될 수 있다.
- [0154] 실시예들에서, `output_layer_flag[i][j]`는 i 번째 출력 계층 세트에 대한 `vps_output_layers_mode[i]`의 값에 따라 시그널링될 수 있거나 시그널링되지 않을 수 있다.
- [0155] 실시예들에서, 도 19를 참조하면, 플래그 `vps_ptl_signal_flag[i]`는 i 번째 출력 계층 세트에 대해 존재할 수 있다. `vps_ptl_signal_flag[i]`의 값에 따라, i 번째 출력 계층 세트에 대한 프로파일 티어 레벨 정보가 시그널링될 수 있거나 시그널링되지 않을 수 있다.
- [0156] 실시예들에서, 도 20을 참조하면, 현재 CVS에서의 서브픽처의 수 `max_subpics_minus1`은 고레벨 선택스 구조, 예를 들어, VPS, DPS, SPS, PPS, APS 또는 SEI 메시지에서 시그널링될 수 있다.
- [0157] 실시예들에서, 도 20을 참조하면, 서브픽처들의 수가 1보다 클 때(`max_subpics_minus1 > 0`), i 번째 서브픽처에 대한 서브픽처 식별자 `sub_pic_id[i]`가 시그널링될 수 있다.
- [0158] 실시예들에서, 각각의 출력 계층 세트의 각각의 계층에 속하는 서브픽처 식별자를 표시하는 하나 이상의 선택스 요소가 VPS에서 시그널링될 수 있다. 도 20을 참조하면, `sub_pic_id_layer[i][j][k]`는 i 번째 출력 계층 세트의 j 번째 계층에 존재하는 k 번째 서브픽처를 표시한다. 이 정보를 사용하여, 디코더는 특정 출력 계층 세트의 각각의 계층에 대해 어느 서브-픽처가 디코딩되고 출력될 수 있는지를 인식할 수 있다.
- [0159] 일 실시예에서, 픽처 헤더(PH)는 코딩된 픽처의 모든 슬라이스들에 적용되는 선택스 요소들을 포함하는 선택스 구조이다. 픽처 유닛(PU)은 특정된 분류 규칙에 따라 서로 연관되고, 디코딩 순서에서 연속적이며, 정확히 하나의 코딩된 픽처를 포함하는 NAL 유닛들의 세트이다. PU는 픽처 헤더(PH) 및 코딩된 픽처를 합성하는 하나 이

상의 VCL NAL 유닛을 포함할 수 있다.

- [0160] 일 실시예에서, SPS(RBSP)는, 그것이 참조되기 전에 디코딩 프로세스에 이용가능하거나, TemporalId가 0인 적어도 하나의 AU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0161] 일 실시예에서, SPS(RBSP)는, 그것이 참조되기 전에 디코딩 프로세스에 이용가능하거나, SPS를 참조하는 하나 이상의 PPS를 포함하는, CVS에서의 TemporalId가 0인 적어도 하나의 AU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0162] 일 실시예에서, SPS(RBSP)는, 그것이 하나 이상의 PPS에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, SPS를 참조하는 하나 이상의 PPS를 포함하는, CVS에서의 SPS NAL 유닛을 참조하는 PPS NAL 유닛들의 최저 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0163] 일 실시예에서, SPS(RBSP)는, 그것이 하나 이상의 PPS에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, 0과 동일한 TemporalId 및 SPS NAL 유닛을 참조하는 PPS NAL 유닛들의 최저 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0164] 일 실시예에서, SPS(RBSP)는, 그것이 하나 이상의 PPS에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, 0과 동일한 TemporalId 및 SPS를 참조하는 하나 이상의 PPS를 포함하는, CVS에서의 SPS NAL 유닛을 참조하는 PPS NAL 유닛의 최저 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0165] 동일한 또는 다른 실시예에서, pps_seq_parameter_set_id는 참조된 SPS에 대한 sps_seq_parameter_set_id의 값을 지정한다. pps_seq_parameter_set_id의 값은 CLVS에서의 코딩된 픽처들에 의해 참조되는 모든 PPS들에서 동일할 수 있다.
- [0166] 동일한 또는 다른 실시예에서, CVS에서의 sps_seq_parameter_set_id의 특정한 값을 갖는 모든 SPS NAL 유닛들은 동일한 내용을 가질 수 있다.
- [0167] 동일한 또는 다른 실시예에서, nuh_layer_id 값들에 관계없이, SPS NAL 유닛들은 sps_seq_parameter_set_id의 동일한 값 공간을 공유할 수 있다.
- [0168] 동일한 또는 다른 실시예에서, SPS NAL 유닛의 nuh_layer_id 값은 SPS NAL 유닛을 참조하는 PPS NAL 유닛들의 최저 nuh_layer_id 값과 동일할 수 있다.
- [0169] 일 실시예에서, m과 동일한 nuh_layer_id를 갖는 SPS가 n과 동일한 nuh_layer_id를 갖는 하나 이상의 PPS에 의해 참조된다. m과 동등한 nuh_layer_id를 갖는 계층은 n과 동등한 nuh_layer_id를 갖는 계층 또는 m과 동등한 nuh_layer_id를 갖는 계층의 (직접 또는 간접) 참조 계층과 동일할 수 있다.
- [0170] 일 실시예에서, PPS(RBSP)는, 그것이 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS NAL 유닛의 TemporalId와 동일한 TemporalId를 갖는 적어도 하나의 AU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0171] 일 실시예에서, PPS(RBSP)는, 그것이 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS를 참조하는 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)를 포함하는, CVS에서의 PPS NAL 유닛의 TemporalId와 동일한 TemporalId를 갖는 적어도 하나의 AU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0172] 일 실시예에서, PPS(RBSP)는, 그것이 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS를 참조하는 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)를 포함하는, CVS에서의 PPS NAL 유닛을 참조하는 코딩된 슬라이스 NAL 유닛들의 최저 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0173] 일 실시예에서, PPS(RBSP)는, 그것이 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS NAL 유닛의 TemporalId와 동일한 TemporalId 및 PPS를 참조하는 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)를 포함하는, CVS에서의 PPS NAL 유닛을 참조하는 코딩된 슬라이스 NAL 유닛들의 최저 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.

- [0174] 동일한 또는 다른 실시예에서, PH에서의 `ph_pic_parameter_set_id`는 사용 중인 참조된 PPS에 대한 `pps_pic_parameter_set_id`의 값을 지정할 수 있다. `pps_seq_parameter_set_id`의 값은 CLVS에서의 코딩된 픽처들에 의해 참조되는 모든 PPS들에서 동일할 수 있다.
- [0175] 동일한 또는 다른 실시예에서, PU 내의 `pps_pic_parameter_set_id`의 특정한 값을 갖는 모든 PPS NAL 유닛들은 동일한 내용을 가질 수 있다.
- [0176] 동일한 또는 다른 실시예에서, `nuh_layer_id` 값들에 관계없이, PPS NAL 유닛들은 `pps_pic_parameter_set_id`의 동일한 값 공간을 공유할 수 있다.
- [0177] 동일한 또는 다른 실시예에서, PPS NAL 유닛의 `nuh_layer_id` 값은 PPS NAL 유닛을 참조하는 NAL 유닛을 참조하는 코딩된 슬라이스 NAL 유닛들의 최저 `nuh_layer_id` 값과 동일할 수 있다.
- [0178] 일 실시예에서, `m`과 동일한 `nuh_layer_id`를 갖는 PPS가 `n`과 동일한 `nuh_layer_id`를 갖는 하나 이상의 코딩된 슬라이스 NAL 유닛에 의해 참조된다. `m`과 동등한 `nuh_layer_id`를 갖는 계층은 `n`과 동등한 `nuh_layer_id`를 갖는 계층 또는 `m`과 동등한 `nuh_layer_id`를 갖는 계층의 (직접 또는 간접) 참조 계층과 동일할 수 있다.
- [0179] 일 실시예에서, PPS(RBSP)는, 그것이 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS NAL 유닛의 `TemporalId`와 동일한 `TemporalId`를 갖는 적어도 하나의 AU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0180] 일 실시예에서, PPS(RBSP)는, 그것이 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS를 참조하는 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)를 포함하는, CVS에서의 PPS NAL 유닛의 `TemporalId`와 동일한 `TemporalId`를 갖는 적어도 하나의 AU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0181] 일 실시예에서, PPS(RBSP)는, 그것이 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS를 참조하는 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)를 포함하는, CVS에서의 PPS NAL 유닛을 참조하는 코딩된 슬라이스 NAL 유닛들의 최저 `nuh_layer_id` 값과 동일한 `nuh_layer_id`를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0182] 일 실시예에서, PPS(RBSP)는, 그것이 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)에 의해 참조되기 전에 디코딩 프로세스에 이용가능하거나, PPS NAL 유닛의 `TemporalId`와 동일한 `TemporalId` 및 PPS를 참조하는 하나 이상의 PH(또는 코딩된 슬라이스 NAL 유닛)를 포함하는, CVS에서의 PPS NAL 유닛을 참조하는 코딩된 슬라이스 NAL 유닛들의 최저 `nuh_layer_id` 값과 동일한 `nuh_layer_id`를 갖는 적어도 하나의 PU에 포함되거나, 또는 외부 수단을 통해 제공될 수 있다.
- [0183] 동일한 또는 다른 실시예에서, PH에서의 `ph_pic_parameter_set_id`는 사용 중인 참조된 PPS에 대한 `pps_pic_parameter_set_id`의 값을 지정할 수 있다. `pps_seq_parameter_set_id`의 값은 CLVS에서의 코딩된 픽처들에 의해 참조되는 모든 PPS들에서 동일할 수 있다.
- [0184] 동일한 또는 다른 실시예에서, PU 내의 `pps_pic_parameter_set_id`의 특정한 값을 갖는 모든 PPS NAL 유닛들은 동일한 내용을 가질 수 있다.
- [0185] 동일한 또는 다른 실시예에서, `nuh_layer_id` 값들에 관계없이, PPS NAL 유닛들은 `pps_pic_parameter_set_id`의 동일한 값 공간을 공유할 수 있다.
- [0186] 동일한 또는 다른 실시예에서, PPS NAL 유닛의 `nuh_layer_id` 값은 PPS NAL 유닛을 참조하는 NAL 유닛을 참조하는 코딩된 슬라이스 NAL 유닛들의 최저 `nuh_layer_id` 값과 동일할 수 있다.
- [0187] 일 실시예에서, `m`과 동일한 `nuh_layer_id`를 갖는 PPS가 `n`과 동일한 `nuh_layer_id`를 갖는 하나 이상의 코딩된 슬라이스 NAL 유닛에 의해 참조된다. `m`과 동등한 `nuh_layer_id`를 갖는 계층은 `n`과 동등한 `nuh_layer_id`를 갖는 계층 또는 `m`과 동등한 `nuh_layer_id`를 갖는 계층의 (직접 또는 간접) 참조 계층과 동일할 수 있다.
- [0188] 일 실시예에서, 도 21에 도시된 바와 같이, 픽처 파라미터 세트 내의 `pps_subpic_id[i]`는 i 번째 서브픽처의 서브픽처 ID를 지정할 수 있다. `pps_subpic_id[i]` 신텍스 요소의 길이는 `pps_subpic_id_len_minus1 + 1` 비트이다.
- [0189] 변수 `SubpicIdVal[i]`는, 0으로부터 `sps_num_subpics_minus1`까지의 범위에 있는 i 의 각각의 값에 대해, 다음과 같이 도출될 수 있다:

```

for( i = 0; i <= sps_num_subpics_minus1; i++ )
if( subpic_id_mapping_explicitly_signalled_flag )
    SubpicIdVal[ i ] = subpic_id_mapping_in_pps_flag ? pps_subpic_id[ i ] :
sps_subpic_id[ i ]
else

```

[0190] SubpicIdVal[i] = i

[0191] 동일한 또는 다른 실시예에서, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i 및 j의 임의의 2개의 상이한 값에 대해, SubpicIdVal[i]는 SubpicIdVal[j]와 동일하지 않을 수 있다.

[0192] 동일한 또는 다른 실시예에서, 현재 픽처가 CLVS의 제1 픽처가 아닐 때, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i의 각각의 값에 대해, SubpicIdVal[i]의 값이 동일한 계층에서의 디코딩 순서로 이전 픽처의 SubpicIdVal[i]의 값과 동일하지 않으면, 서브픽처 인덱스 i를 갖는 현재 픽처 내의 서브픽처의 모든 코딩된 슬라이스 NAL 유닛들에 대한 nal_unit_type는 IDR_W_RADL로부터 CRA_NUT까지의 범위에 있는 특정한 값과 동일할 수 있다.

[0193] 동일한 또는 다른 실시예에서, 현재 픽처가 CLVS의 제1 픽처가 아닐 때, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i의 각각의 값에 대해, SubpicIdVal[i]의 값이 동일한 계층에서의 디코딩 순서로 이전 픽처의 SubpicIdVal[i]의 값과 동일하지 않으면, sps_independent_subpics_flag는 1과 동일할 수 있다.

[0194] 동일한 또는 다른 실시예에서, 현재 픽처가 CLVS의 제1 픽처가 아닐 때, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i의 각각의 값에 대해, SubpicIdVal[i]의 값이 동일한 계층에서의 디코딩 순서로 이전 픽처의 SubpicIdVal[i]의 값과 동일하지 않으면, subpic_treated_as_pic_flag[i] 및 loop_filter_across_subpic_enabled_flag[i]는 1과 동일할 수 있다.

[0195] 동일한 또는 다른 실시예에서, 현재 픽처가 CLVS의 제1 픽처가 아닐 때, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i의 각각의 값에 대해, SubpicIdVal[i]의 값이 동일한 계층에서의 디코딩 순서로 이전 픽처의 SubpicIdVal[i]의 값과 동일하지 않으면, sps_independent_subpics_flag는 1과 동일할 수 있거나 또는 subpic_treated_as_pic_flag[i] 및 loop_filter_across_subpic_enabled_flag[i]는 1과 동일할 수 있다.

[0196] 동일한 또는 다른 실시예에서, 서브픽처가 다른 서브픽처에 대한 어떠한 참조도 없이 독립적으로 인코딩될 때, 영역의 서브픽처 식별자의 값은 코딩된 비디오 시퀀스 내에서 변경될 수 있다.

[0197] 실시예들에서, 픽처는 하나 이상의 서브픽처로 파티셔닝될 수 있다. 각각의 서브픽처는 하나 이상의 슬라이스로 더 파티셔닝될 수 있다. 서브픽처 파티셔닝 정보를 시그널링하기 위한 효율적인 시그널링 방법이 요구된다.

[0198] 샘플들은 CTB들의 단위로 처리될 수 있다. 폭과 높이 양자 모두에서의 각각의 루마 CTB에 대한 어레이 크기는 샘플들의 단위로 CtbSizeY이다. 각각의 크로마 CTB에 대한 어레이의 폭 및 높이는 샘플들의 단위로 각각 CtbWidthC 및 CtbHeightC이다. 각각의 CTB에는 인트라 또는 인터 예측을 위한 그리고 변환 코딩을 위한 블록 크기들을 식별하기 위한 파티션 시그널링이 할당된다. 파티셔닝은 재귀적 쿼드트리 파티셔닝(recursive quadtree partitioning)이다. 쿼드트리의 루트(root)는 CTB와 연관된다. 쿼드트리는 쿼드트리 리프(quadtree leaf)로 지칭되는 리프에 도달할 때까지 분할된다. 컴포넌트 폭이 CTB 크기의 정수가 아닐 때, 우측 컴포넌트 경계에서의 CTB들은 불완전하다. 컴포넌트 높이가 CTB 크기의 정수배가 아닐 때, 하단 컴포넌트 경계에서의 CTB들은 불완전하다.

[0199] 각각의 서브픽처의 폭 및 높이 정보는 CtbSizeY의 단위로 SPS에서 시그널링될 수 있다. 도 22에서, 예를 들어, subpic_width_minus1[i] + 1은 CtbSizeY의 단위로 i번째 서브픽처의 폭을 지정한다. 신텍스 요소의 길이는 $\lceil \log_2((\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}) \rceil$ 비트이다. 존재하지 않을 때, subpic_width_minus1[i]의 값은 $((\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}) - \text{subpic_ctu_top_left_x}[i] - 1$ 과 동일한 것으로 추론된다. subpic_height_minus1[i] + 1은 CtbSizeY의 단위로 i번째 서브픽처의 높이를 지정한다. 신텍스 요소의 길이는 $\lceil \log_2((\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}) \rceil$ 비트이다. 존재하지 않을 때, subpic_height_minus1[i]의 값은 $((\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}) - \text{subpic_ctu_top_left_y}[i] - 1$ 과 동일한 것으로 추론된다.

- [0200] 픽처 폭이 CtbSizeY 이상일 때, 각각의 서브픽처의 폭은 CtbSizeY 이상일 수 있다. 픽처 높이가 CtbSizeY 이상일 때, 각각의 서브픽처의 높이는 CtbSizeY 이상일 수 있다.
- [0201] 픽처 폭이 CtbSizeY보다 크지 않고 픽처 높이가 CtbSizeY보다 크지 않은 경우, 픽처는 하나보다 많은 서브픽처로 파티셔닝되지 않을 수 있다. 이 경우, 서브픽처들의 수는 1일 수 있다.
- [0202] pic_width_max_in_luma_samples가 CtbSizeY보다 크지 않고 pic_height_max_in_luma_samples가 CtbSizeY보다 크지 않을 때, subpic_info_present_flag의 값은 0일 수 있다. subpic_info_present_flag가 0일 때, 서브픽처 파티셔닝 정보에 대해 명시적 시그널링이 존재하지 않고, 픽처 내의 서브픽처의 수는 1이다.
- [0203] 동일한 또는 다른 실시예에서, sps_subpic_id_len_minus1 + 1은 선택스 요소 sps_subpic_id[i], 선택스 요소들 pps_subpic_id[i](존재하는 경우), 및 선택스 요소 slice_subpic_id(존재하는 경우)를 표현하는 데 사용되는 비트들의 수를 지정한다. sps_subpic_id_len_minus1의 값은 0으로부터 15까지의 범위에 있을 수 있다. $1 \ll (\text{sps_subpic_id_len_minus1})$ 의 값은 sps_num_subpics_minus1 + 1 이상일 수 있다.
- [0204] 동일한 또는 다른 실시예에서, 서브픽처들의 수가 1일 때, subpic_info_present_flag는 0일 수 있고 서브픽처 파티셔닝 정보는 명시적으로 시그널링되지 않을 수 있는데, 이는 그 경우에 서브픽처 폭 및 높이 정보가 픽처 폭 및 높이 정보와 동일하고 서브픽처의 좌측-상단 위치가 픽처의 좌측-상단 위치와 동일하기 때문이다.
- [0205] 예를 들어, subpic_ctu_top_left_x[i]는 CtbSizeY의 단위로 i번째 서브픽처의 상단 좌측 CTU의 수평 위치를 지정한다. 선택스 요소의 길이는 $\text{Ceil}(\text{Log2}(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY})$ 비트이다. 존재하지 않을 때, subpic_ctu_top_left_x[i]의 값은 0인 것으로 추론된다. subpic_ctu_top_left_y[i]는 CtbSizeY의 단위로 i번째 서브픽처의 상단 좌측 CTU의 수직 위치를 지정한다. 선택스 요소의 길이는 $\text{Ceil}(\text{Log2}(\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY})$ 비트이다. 존재하지 않을 때, subpic_ctu_top_left_y[i]의 값은 0인 것으로 추론된다. subpic_width_minus1[i] + 1은 CtbSizeY의 단위로 i번째 서브픽처의 폭을 지정한다. 선택스 요소의 길이는 $\text{Ceil}(\text{Log2}(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY})$ 비트이다. 존재하지 않을 때, subpic_width_minus1[i]의 값은 $(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY} - \text{subpic_ctu_top_left_x}[i] - 1$ 과 동일한 것으로 추론된다. subpic_height_minus1[i] + 1은 CtbSizeY의 단위로 i번째 서브픽처의 높이를 지정한다. 선택스 요소의 길이는 $\text{Ceil}(\text{Log2}(\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY})$ 비트이다. 존재하지 않을 때, subpic_height_minus1[i]의 값은 $(\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY} - \text{subpic_ctu_top_left_y}[i] - 1$ 과 동일한 것으로 추론된다.
- [0206] 동일한 또는 다른 실시예에서, 서브픽처들의 수가 1보다 클 때, 도 22에 도시된 바와 같이, subpic_info_present_flag는 1일 수 있고 서브픽처 파티셔닝 정보는 파라미터 세트에서 명시적으로 시그널링될 수 있다.
- [0207] 예를 들어, 도 22에서, sps_num_subpics_minus2 + 2는 CLVS에서의 각각의 픽처 내의 서브픽처들의 수를 지정한다. sps_num_subpics_minus2의 값은 0으로부터 $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}) * \text{Ceil}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 2$ 까지의 범위에 있을 수 있다. 존재하지 않을 때, sps_num_subpics_minus2의 값은 0인 것으로 추론된다.
- [0208] 동일한 실시예에서, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i에 대한, 각각 타일 열들 및 행들에서의 i번째 서브픽처의 폭 및 높이를 지정하는 리스트들 SubpicWidthInTiles[i] 및 SubpicHeightInTiles[i]과, 0으로부터 sps_num_subpics_minus1까지의 범위에 있는 i에 대한, i번째 서브픽처의 높이가 하나의 타일 행보다 작은지를 지정하는 리스트 subpicHeightLessThanOneTileFlag[i]가 다음과 같이 도출될 수 있다:

```

for( i = 0; i <= sps_num_subpics_minus2 + 1; i++ ) {
    leftX = subpic_ctu_top_left_x[ i ]
    rightX = leftX + subpic_width_minus1[ i ]
    SubpicWidthInTiles[ i ] = ctbToTileColIdx[ rightX ] + 1 - ctbToTileColIdx[ leftX ]
    topY = subpic_ctu_top_left_y[ i ]
    bottomY = topY + subpic_height_minus1[ i ]
    SubpicHeightInTiles[ i ] =
ctbToTileRowIdx[ bottomY ] + 1 - ctbToTileRowIdx[ topY ]
    if( SubpicHeightInTiles[ i ] == 1 &&
        subpic_height_minus1[ i ] + 1 < RowHeight[ ctbToTileRowIdx[ topY ] ] )
        subpicHeightLessThanOneTileFlag[ i ] = 1
    else
        subpicHeightLessThanOneTileFlag[ i ] = 0
}

```

[0209]

[0210]

rect_slice_flag가 1일 때, i번째 슬라이스 내의 CTU들의 수를 지정하는 0으로부터 num_slices_in_pic_minus1까지의 범위에 있는 i에 대한 리스트 NumCtusInSlice[i], 슬라이스 내의 제1 CTU를 포함하는 타일의 타일 인덱스를 지정하는 0으로부터 num_slices_in_pic_minus1까지의 범위에 있는 i에 대한 리스트 SliceTopLeftTileIdx[i], 및 i번째 슬라이스 내의 j번째 CTB의 픽처 래스터 스캔 어드레스를 지정하는 0으로부터 num_slices_in_pic_minus1까지의 범위에 있는 i, 및 0으로부터 NumCtusInSlice[i]-1까지의 범위에 있는 j에 대한 행렬 CtbAddrInSlice[i][j], 및 i번째 슬라이스를 포함하는 타일 내의 슬라이스들의 수를 지정하는 변수 NumSlicesInTile[i]는 다음과 같이 도출될 수 있다:

```

if( single_slice_per_subpic_flag ) {
    for( i = 0; i <= sps_num_subpics_minus2 + 1; i++ ) {
        NumCtusInSlice[ i ] = 0
        if( subpicHeightLessThanOneTileFlag[ i ] ) /* 슬라이스는 타일 내의 다수의 CTU
행들로 구성된다. */
            AddCtbsToSlice( i, subpic_ctu_top_left_x[ i ],
                subpic_ctu_top_left_x[ i ] + subpic_width_minus1[ i ] + 1,
subpic_ctu_top_left_y[ i ],
                subpic_ctu_top_left_y[ i ] + subpic_height_minus1[ i ] + 1 )
        else { /* 슬라이스는 직사각형 영역을 커버하는 다수의 완전한 타일들로 구성된다.
*/
            tileX = CtbToTileColBd[ subpic_ctu_top_left_x[ i ] ]
            tileY = CtbToTileRowBd[ subpic_ctu_top_left_y[ i ] ]
            for( j = 0; j < SubpicHeightInTiles[ i ]; j++ )
                for( k = 0; k < SubpicWidthInTiles[ i ]; k++ )
                    AddCtbsToSlice( i, tileColBd[ tileX + k ], tileColBd[ tileX + k + 1 ],
tileRowBd[ tileY + j ],
                            tileRowBd[ tileY + j + 1 ] )
                }
            }
        } else {
            tileIdx = 0
            for( i = 0; i <= num_slices_in_pic_minus1; i++ )
                NumCtusInSlice[ i ] = 0
            for( i = 0; i <= num_slices_in_pic_minus1; i++ ) {
                SliceTopLeftTileIdx[ i ] = tileIdx
                tileX = tileIdx % NumTileColumns
                tileY = tileIdx / NumTileColumns
                if( i < num_slices_in_pic_minus1 ) {
                    sliceWidthInTiles[ i ] = slice_width_in_tiles_minus1[ i ] + 1
                    sliceHeightInTiles[ i ] = slice_height_in_tiles_minus1[ i ] + 1
                } else {
                    sliceWidthInTiles[ i ] = NumTileColumns - tileX
                    sliceHeightInTiles[ i ] = NumTileRows - tileY
                }
            }
        }
    }
}

```

[0211]

```

    NumSlicesInTile[ i ] = 1
}
if( slicWidthInTiles[ i ] == 1 && sliceHeightInTiles[ i ] == 1 ) {
    if( num_exp_slices_in_tile[ i ] == 0 ) {
        NumSlicesInTile[ i ] = 1
        sliceHeightInCtus[ i ] =
RowHeight[ SliceTopLeftTileIdx[ i ] / NumTileColumns ]
    } else {
        remainingHeightInCtbsY =
RowHeight[ SliceTopLeftTileIdx[ i ] / NumTileColumns ]
        for( j = 0; j < num_exp_slices_in_tile[ i ] - 1; j++ ) {
            sliceHeightInCtus[ i + j ] = exp_slice_height_in_ctus_minus1[ i ][ j ] + 1
            remainingHeightInCtbsY -= sliceHeightInCtus[ i + j ]
        }
        uniformSliceHeight = exp_slice_height_in_ctus_minus1[ i ][ j ] + 1
        while( remainingHeightInCtbsY >= uniformSliceHeight ) {
            sliceHeightInCtus[ i + j ] = uniformSliceHeight
            remainingHeightInCtbsY -= uniformSliceHeight
            j++
        }
        if( remainingHeightInCtbsY > 0 ) {
            sliceHeightInCtus[ i + j ] = remainingHeightInCtbsY
            j++
        }
        NumSlicesInTile[ i ] = j
    }
}
ctbY = tileRowBd[ tileY ]
for( j = 0; j < NumSlicesInTile[ i ]; j++ ) {
    AddCtbsToSlice( i + j, tileColBd[ tileX ], tileColBd[ tileX + 1 ],
        ctbY, ctbY + sliceHeightInCtus[ i + j ] )
    ctbY += sliceHeightInCtus[ i + j ]
}
i += NumSlicesInTile[ i ] - 1
} else

```

[0212]

될 수 있다. 일 예에서, 하나 이상의 프로세서는 제안된 방법들 중 하나 이상을 수행하기 위해 비일시적 컴퓨터 판독가능 매체에 저장되는 프로그램을 실행한다.

- [0228] 위에서 설명된 기법들은 컴퓨터 판독가능 명령어들을 사용하여 컴퓨터 소프트웨어로서 구현되고 하나 이상의 컴퓨터 판독가능 매체에 물리적으로 저장될 수 있다. 예를 들어, 도 24는 개시된 주제의 특정 실시예들을 구현하기에 적합한 컴퓨터 시스템(2400)을 도시한다.
- [0229] 컴퓨터 소프트웨어는, 컴퓨터 중앙 처리 유닛(CPU), 그래픽 처리 유닛(GPU) 등에 의해, 직접, 또는 해석, 마이크로-코드 실행 등을 통해 실행될 수 있는 명령어들을 포함하는 코드를 생성하기 위해 어셈블리, 컴파일(compilation), 링킹(linking), 또는 유사한 메커니즘들이 수행될 수 있는 임의의 적합한 머신 코드 또는 컴퓨터 언어를 사용하여 코딩될 수 있다.
- [0230] 명령어들은, 예를 들어, 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트폰, 게이밍 디바이스, 사물 인터넷 디바이스 등을 포함하여, 다양한 타입의 컴퓨터들 또는 그것의 컴포넌트들 상에서 실행될 수 있다.
- [0231] 컴퓨터 시스템(2400)에 대한 도 24에 도시된 컴포넌트들은 사실상 예시적인 것이고, 본 개시내용의 실시예들을 구현하는 컴퓨터 소프트웨어의 사용 또는 기능성의 범위에 대한 임의의 제한을 암시하도록 의도되지 않는다. 컴포넌트들의 구성이 컴퓨터 시스템(2400)의 예시적인 실시예에서 예시된 컴포넌트들 중 임의의 하나 또는 이들의 조합과 관련하여 임의의 종속성 또는 요건을 갖는 것으로 해석되어서도 안 된다.
- [0232] 컴퓨터 시스템(2400)은 특정 휴먼 인터페이스 입력 디바이스들을 포함할 수 있다. 그러한 휴먼 인터페이스 입력 디바이스는, 예를 들어, 촉각 입력(예컨대: 키스트로크, 스와이프, 데이터 글러브 움직임), 오디오 입력(예컨대: 음성, 손뼉), 시각적 입력(예컨대, 제스처), 후각적 입력(묘사되지 않음)을 통한 하나 이상의 인간 사용자에 의한 입력에 응답할 수 있다. 휴먼 인터페이스 디바이스들은 또한 오디오(예컨대: 음성, 음악, 주변 사운드), 이미지들(예컨대: 스캐닝된 이미지들, 스틸 이미지 카메라로부터 획득된 사진 이미지들), 비디오(예컨대 2차원 비디오, 입체적 비디오를 포함하는 3차원 비디오)와 같은, 인간에 의한 의식적인 입력과 반드시 직접적으로 관련되지 않는 특정 미디어를 캡처하기 위해 사용될 수 있다.
- [0233] 입력 휴먼 인터페이스 디바이스들은: 키보드(2401), 마우스(2402), 트랙패드(2403), 터치 스크린(2410) 및 연관된 그래픽 어댑터(2450), 데이터-글러브, 조이스틱(2405), 마이크로폰(2406), 스캐너(2407), 카메라(2408) 중 하나 이상(각각의 하나만이 묘사됨)을 포함할 수 있다.
- [0234] 컴퓨터 시스템(2400)은 특정 휴먼 인터페이스 출력 디바이스들을 또한 포함할 수 있다. 그러한 휴먼 인터페이스 출력 디바이스들은, 예를 들어, 촉각 출력, 사운드, 광, 및 냄새/맛을 통해 하나 이상의 인간 사용자의 감각들을 자극하고 있을 수 있다. 그러한 휴먼 인터페이스 출력 디바이스들은 촉각 출력 디바이스들(예를 들어 터치-스크린(2410), 데이터-글러브, 또는 조이스틱(2405)에 의한 촉각 피드백, 그러나 입력 디바이스들로서 역할을 하지 않는 촉각 피드백 디바이스들도 있을 수 있음), 오디오 출력 디바이스들(예컨대: 스피커들(2409), 헤드폰들(묘사되지 않음)), 시각적 출력 디바이스들(예컨대, 각각 터치-스크린 입력 능력이 있거나 없고, 각각 촉각 피드백 능력이 있거나 없는, CRT(cathode ray tube) 스크린들, LCD(liquid-crystal display) 스크린들, 플라즈마 스크린들, OLED(organic light-emitting diode) 스크린들을 포함하는 스크린들(2410) - 이들 중 일부는 스테레오그래픽 출력과 같은 수단을 통해 2차원 시각적 출력 또는 3차원을 초과한 출력을 출력할 수 있음 - ; 가상 현실 안경(묘사되지 않음), 홀로그래픽 디스플레이들 및 스모크 탱크들(묘사되지 않음)), 및 프린터들(묘사되지 않음)을 포함할 수 있다.
- [0235] 컴퓨터 시스템(2400)은 인간 액세스가능한 저장 디바이스들 및 그의 연관된 매체들, 예컨대 CD/DVD 등의 매체(2421)를 갖는 CD/DVD ROM/RW(2420)를 포함하는 광학 매체, 썸-드라이브(thumb-drive)(2422), 이동식 하드 드라이브 또는 솔리드 스테이트 드라이브(2423), 테이프 및 플로피 디스크(묘사되지 않음)와 같은 레거시 자기 매체, 보안 동글(묘사되지 않음)과 같은 특수화된 ROM/ASIC/PLD 기반 디바이스들 등을 또한 포함할 수 있다.
- [0236] 본 기술분야의 통상의 기술자들은 또한, 현재 개시된 주제와 관련하여 사용되는 용어 "컴퓨터 판독가능 매체"가 송신 매체, 반송파들, 또는 다른 일시적 신호들을 포함하지 않는다는 것을 이해해야 한다.
- [0237] 컴퓨터 시스템(2400)은 하나 이상의 통신 네트워크(955)에 대한 인터페이스(들)를 또한 포함할 수 있다. 네트워크들은 예를 들어 무선, 유선, 광학일 수 있다. 네트워크들은 추가로 로컬, 광역, 대도시, 차량 및 산업, 실시간, 지연-허용(delay-tolerant) 등일 수 있다. 네트워크들의 예들은 로컬 영역 네트워크들, 예컨대 이더넷, 무선 LAN들, GSM(global systems for mobile communications), 제3 세대(3G), 제4 세대(4G), 제5 세대(5G), 롱-텀 에볼루션(LTE) 등을 포함하는 셀룰러 네트워크들, 케이블 TV, 위성 TV 및 지상파 브로드캐스트 TV를 포함하

는 TV 유선 또는 무선 광역 디지털 네트워크들, CANBus를 포함하는 차량 및 산업 등을 포함한다. 특정 네트워크들은 일반적으로 특정 범용 데이터 포트들 또는 주변 버스들(949)(예컨대, 예를 들어, 컴퓨터 시스템(2400)의 USB(universal serial bus) 포트들)에 부착되는 외부 네트워크 인터페이스 어댑터들(954)을 요구하며; 다른 것들은 일반적으로 아래에 설명되는 바와 같은 시스템 버스(예를 들어, PC 컴퓨터 시스템으로의 이더넷 인터페이스 또는 스마트폰 컴퓨터 시스템으로의 셀룰러 네트워크 인터페이스)로의 부착에 의해 컴퓨터 시스템(2400)의 코어에 통합된다. 예로서, 네트워크(2455)는 네트워크 인터페이스(2454)를 사용하여 주변 버스(2449)에 접속될 수 있다. 이들 네트워크들 중 임의의 것을 사용하여, 컴퓨터 시스템(2400)은 다른 엔티티들과 통신할 수 있다. 그러한 통신은 단방향성 수신 전용(예를 들어, 브로드캐스트 TV), 단방향성 송신 전용(예를 들어, CANbus 대 특정 CANbus 디바이스들), 또는 예를 들어 로컬 영역 또는 광역 디지털 네트워크들을 사용하는 다른 컴퓨터 시스템들과의 양방향성일 수 있다. 위에서 설명한 바와 같은 네트워크들 및 네트워크 인터페이스들(954) 각각에 대해 특정 프로토콜들 및 프로토콜 스택들이 사용될 수 있다.

[0238] 전술한 휴먼 인터페이스 디바이스들, 인간-액세스가능한 저장 디바이스들, 및 네트워크 인터페이스들은 컴퓨터 시스템(2400)의 코어(2440)에 부착될 수 있다.

[0239] 코어(2440)는 하나 이상의 중앙 처리 유닛(CPU)(2441), 그래픽 처리 유닛(GPU)(2442), 필드 프로그래머블 게이트 영역(FPGA)(2443)의 형태로 된 특수화된 프로그래머블 처리 유닛, 특정 태스크들에 대한 하드웨어 가속기(2444) 등을 포함할 수 있다. 이들 디바이스는, 판독 전용 메모리(ROM)(2445), 랜덤 액세스 메모리(RAM)(2446), 내부 비-사용자 액세스가능 하드 드라이브들, SSD(solid-state drive)들 등과 같은 내부 대용량 스토리지(2447)와 함께, 시스템 버스(2448)를 통해 접속될 수 있다. 일부 컴퓨터 시스템들에서, 시스템 버스(2448)는 추가적인 CPU들, GPU들 등에 의한 확장을 가능하게 하기 위해 하나 이상의 물리적 플러그의 형태로 액세스가능할 수 있다. 주변 디바이스들은 코어의 시스템 버스(2448)에 직접, 또는 주변 버스(2449)를 통해 부착될 수 있다. 주변 버스를 위한 아키텍처들은 PCI(peripheral component interconnect), USB 등을 포함한다.

[0240] CPU들(2441), GPU들(2442), FPGA들(2443), 및 가속기들(2444)은, 조합하여, 전술한 컴퓨터 코드를 구성할 수 있는 특정 명령어들을 실행할 수 있다. 그 컴퓨터 코드는 ROM(2445) 또는 RAM(2446)에 저장될 수 있다. 과도적인 데이터가 또한 RAM(2446)에 저장될 수 있는 반면, 영구 데이터는, 예를 들어, 내부 대용량 스토리지(2447)에 저장될 수 있다. 메모리 디바이스들 중 임의의 것에 대한 고속 저장 및 검색은, 하나 이상의 CPU(2441), GPU(2442), 대용량 스토리지(2447), ROM(2445), RAM(2446) 등과 밀접하게 연관될 수 있는, 캐시 메모리의 사용을 통해 가능하게 될 수 있다.

[0241] 컴퓨터 판독가능 매체는 다양한 컴퓨터 구현 동작들을 수행하기 위한 컴퓨터 코드를 가질 수 있다. 매체 및 컴퓨터 코드는 본 개시내용의 목적을 위해 특별히 설계되고 구성된 것들일 수 있거나, 또는 그것들은 컴퓨터 소프트웨어 기술분야의 기술자들에게 잘 알려져 있고 이용가능한 종류의 것일 수 있다.

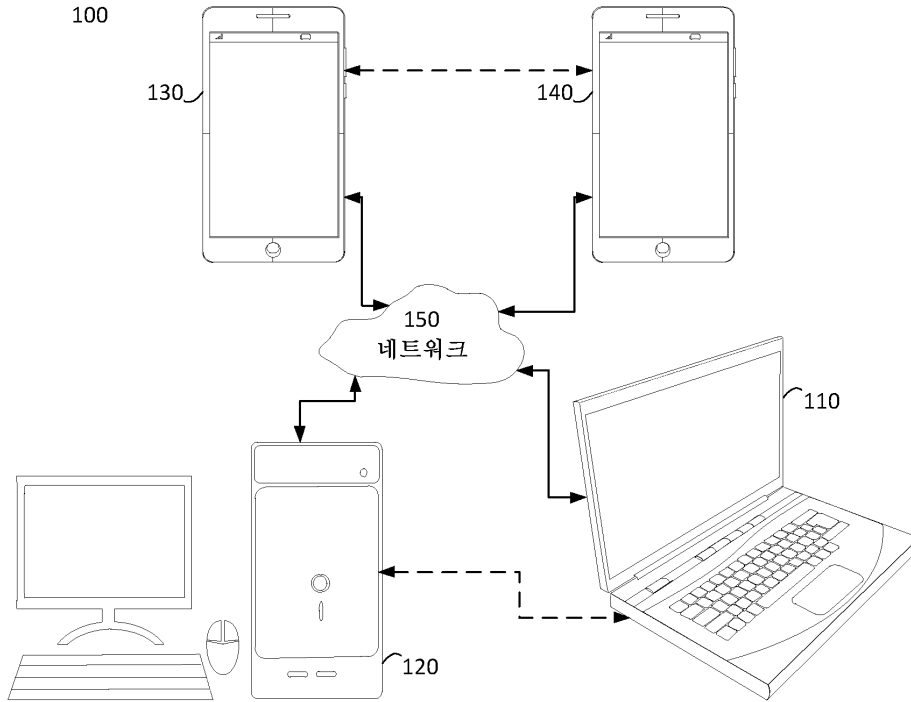
[0242] 제한이 아니라 예로서, 아키텍처를 갖는 컴퓨터 시스템(2400), 및 구체적으로 코어(2440)는 프로세서(들)(CPU들, GPU들, FPGA, 가속기들 등을 포함함)가 하나 이상의 유형의(tangible) 컴퓨터 판독가능 매체에 구현된 소프트웨어를 실행하는 결과로서 기능성을 제공할 수 있다. 그러한 컴퓨터 판독가능 매체는 위에 소개된 바와 같은 사용자-액세스가능한 대용량 스토리지뿐만 아니라, 코어-내부 대용량 스토리지(2447) 또는 ROM(2445)과 같은 비일시적인 본질의 것인 코어(2440)의 특정 스토리지와 연관된 매체일 수 있다. 본 개시내용의 다양한 실시예들을 구현하는 소프트웨어가 그러한 디바이스들에 저장되고 코어(2440)에 의해 실행될 수 있다. 컴퓨터 판독가능 매체는 특정 필요에 따라 하나 이상의 메모리 디바이스 또는 칩을 포함할 수 있다. 소프트웨어는 코어(2440) 및 구체적으로 그 내부의 프로세서들(CPU, GPU, FPGA 등을 포함함)로 하여금, RAM(2446)에 저장된 데이터 구조들을 정의하는 것 및 소프트웨어에 의해 정의된 프로세스들에 따라 그러한 데이터 구조들을 수정하는 것을 포함하여, 본 명세서에 설명된 특정 프로세스들 또는 특정 프로세스들의 특정 부분들을 실행하게 할 수 있다. 추가로 또는 대안으로서, 컴퓨터 시스템은, 본 명세서에 설명된 특정 프로세스들 또는 특정 프로세스들의 특정 부분들을 실행하기 위해 소프트웨어 대신에 또는 그와 함께 동작할 수 있는, 회로(예를 들어: 가속기(2444))에 하드와이어링되거나 다른 방식으로 구현된 로직의 결과로서 기능성을 제공할 수 있다. 소프트웨어에 대한 참조는, 적절한 경우, 로직을 포함할 수 있고, 그 반대도 가능하다. 컴퓨터 판독가능 매체에 대한 참조는, 적절한 경우, 실행을 위한 소프트웨어를 저장하는 회로(예컨대 집적 회로(IC)), 또는 실행을 위한 로직을 구현하는 회로, 또는 둘 다를 포함할 수 있다. 본 개시내용은 하드웨어 및 소프트웨어의 임의의 적합한 조합을 포함한다.

[0243] 본 개시내용이 여러 예시적인 실시예들을 설명하였지만, 본 개시내용의 범위 내에 속하는 변경들, 치환들, 및

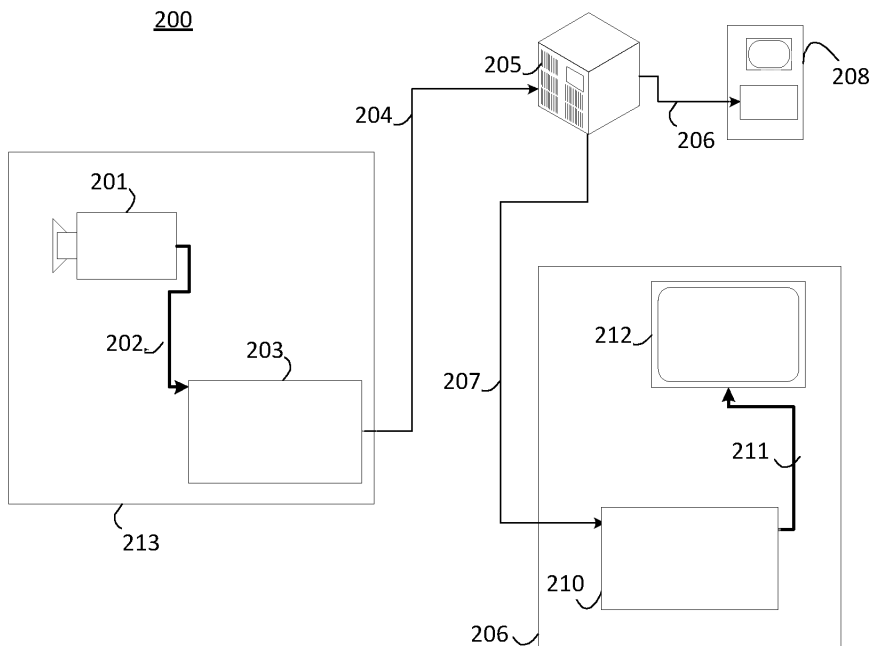
다양한 대체 균등물들이 존재한다. 따라서, 본 기술분야의 통상의 기술자들은, 비록 본 명세서에 명시적으로 도시되거나 설명되지는 않았지만, 본 개시내용의 원리들을 구현하고 따라서 그것의 진의 및 범위 내에 있는, 다수의 시스템들 및 방법들을 고안할 수 있을 것이라는 점이 인정될 것이다.

도면

도면1

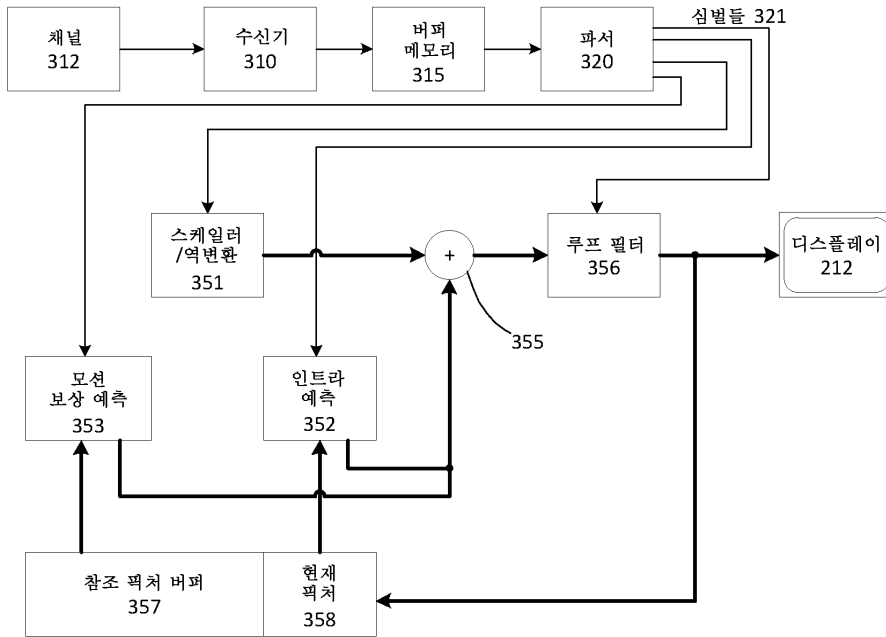


도면2



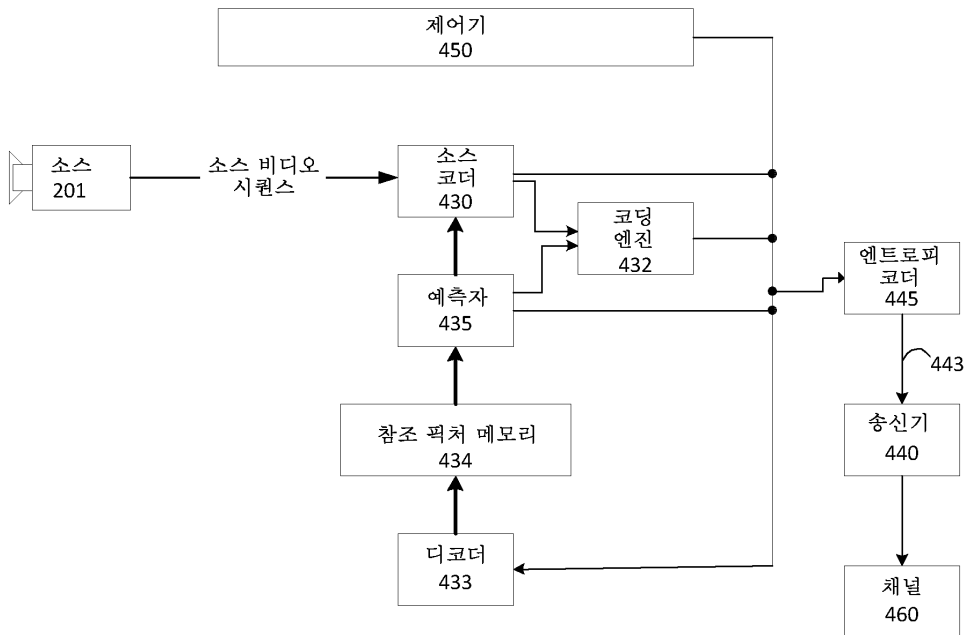
도면3

디코더 210

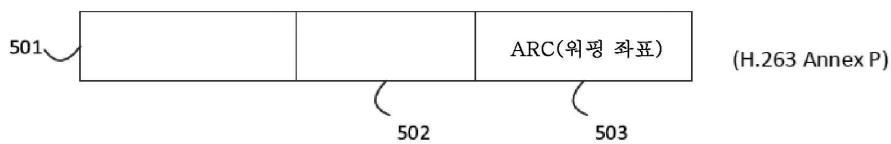


도면4

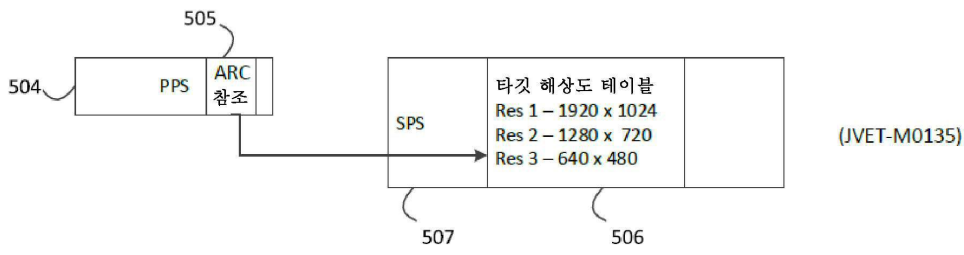
인코더 203



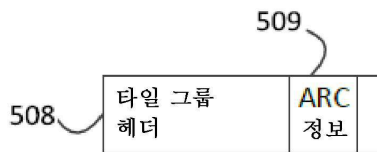
도면5a



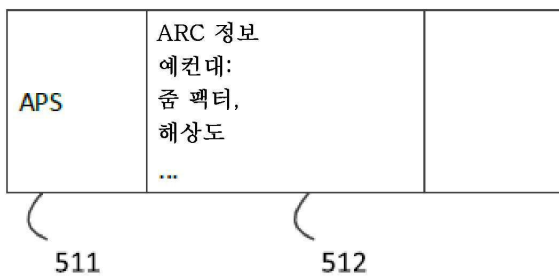
도면5b



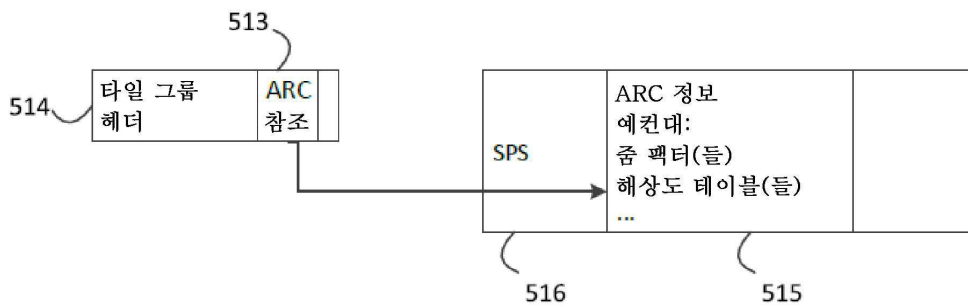
도면5c



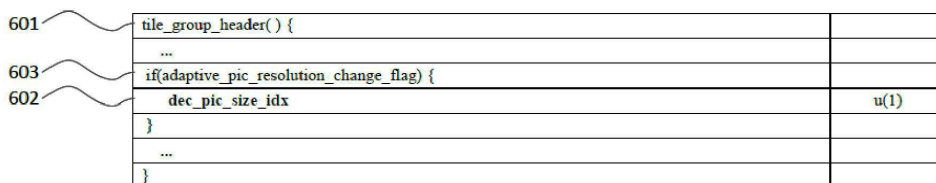
도면5d



도면5e



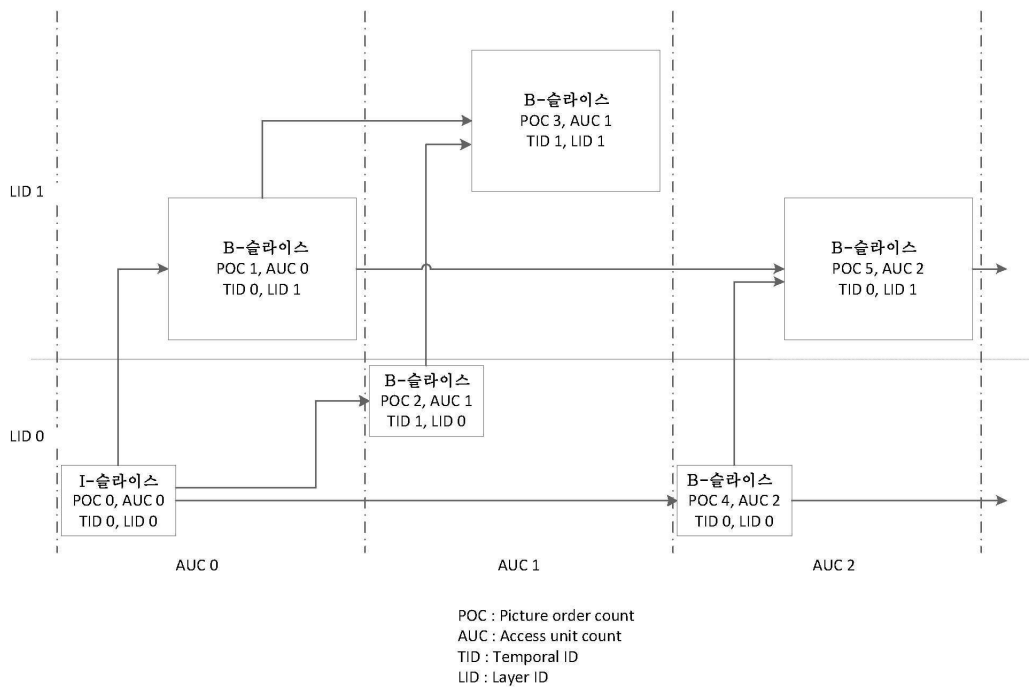
도면6a



도면6b

610	seq_parameter_set_rbsp() {	디스크립터
	...	
611	adaptive_pic_resolution_change_flag	u(1)
612	if(adaptive_pic_resolution_change_flag) {	
613	output_pic_width_in_luma_samples	ue(v)
	output_pic_height_in_luma_samples	ue(v)
614	reference_pic_size_present_flag	u(1)
	if(reference_pic_size_present_flag)	
	{	
615	reference_pic_width_in_luma_samples	ue(v)
	reference_pic_height_in_luma_samples	ue(v)
	}	
616	num_dec_pic_size_in_luma_samples_minus1	ue(v)
	for(i = 0; i <= num_dec_pic_size_in_luma_samples_minus1; i++) {	
617	dec_pic_width_in_luma_samples[i]	ue(v)
	dec_pic_height_in_luma_samples[i]	ue(v)
	}	
	}	
	...	
	}	

도면7

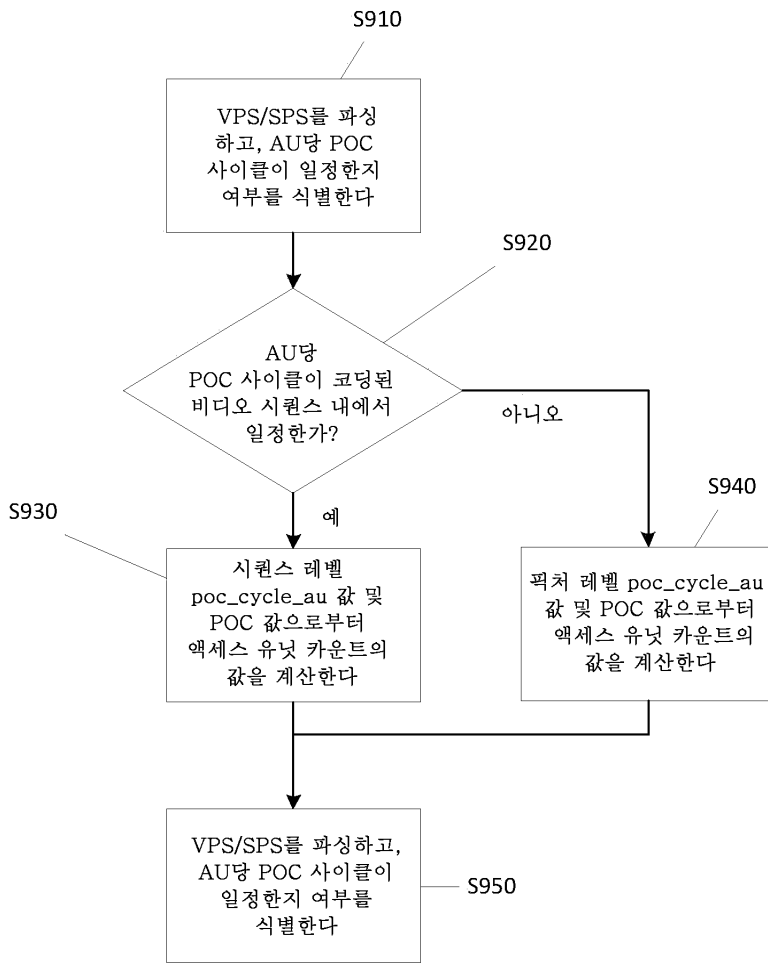


도면8

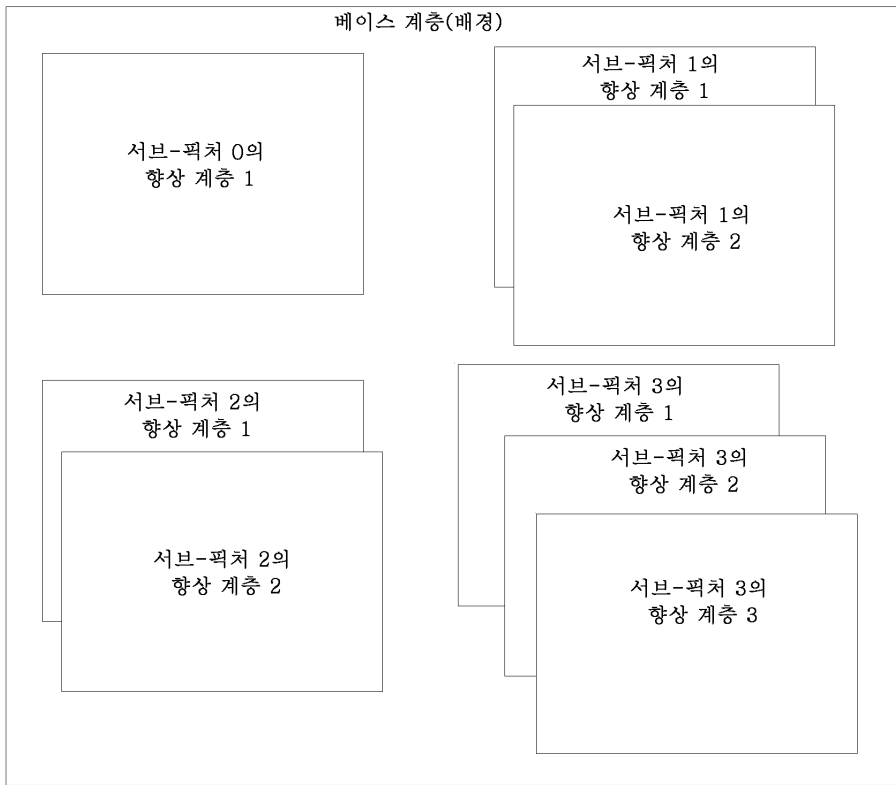
video_parameter_set_rbsp() { ^o	디스크립터
vps_video_parameter_set_id ^o	u(4) ^o
vps_max_layers_minus1 ^o	u(8) ^o
for(i = 0; i <= vps_max_layers_minus1; i++) { ^o	^o
vps_included_layer_id[i] ^o	u(7) ^o
vps_reserved_zero_bit ^o	u(1) ^o
} ^o	^o
vps_constraint_info_present_flag ^o	u(1) ^o
vps_constant_poc_cycle_per_au ^o	u(1) ^o
if(vps_constant_poc_cycle_per_au) ^o	^o
vps_poc_cycle_au ^o	u(8) ^o
... ^o	^o
} ^o	^o

slice_header() { ^o	디스크립터
slice_pic_parameter_set_id ^o	ue(v) ^o
if(rect_slice_flag NumBricksInPic > 1) ^o	^o
slice_address ^o	u(v) ^o
if(!rect_slice_flag && !single_brick_per_slice_flag) ^o	^o
num_bricks_in_slice_minus1 ^o	ue(v) ^o
slice_type ^o	ue(v) ^o
if(NalUnitType == GRA_NUT) ^o	^o
recovery_poc_cnt ^o	se(v) ^o
slice_pic_order_cnt_lsb ^o	u(v) ^o
... ^o	^o
if(!vps_constant_poc_cycle_per_au) ^o	^o
slice_poc_cycle_au ^o	u(8) ^o
^o	^o
^o	^o
} ^o	^o

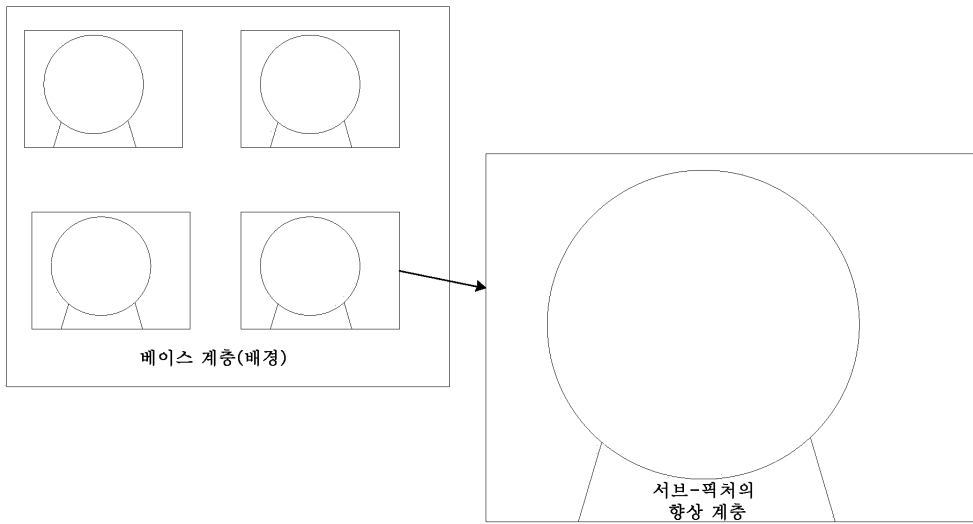
도면9



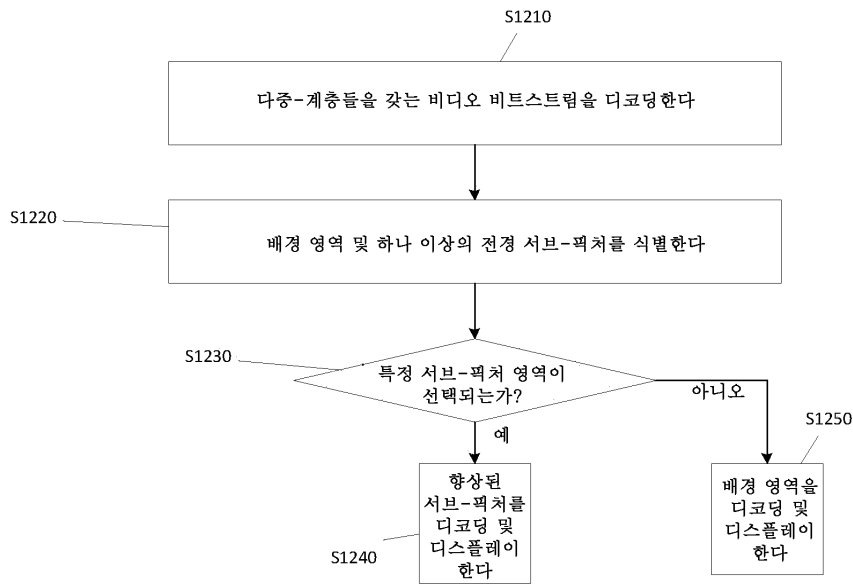
도면10



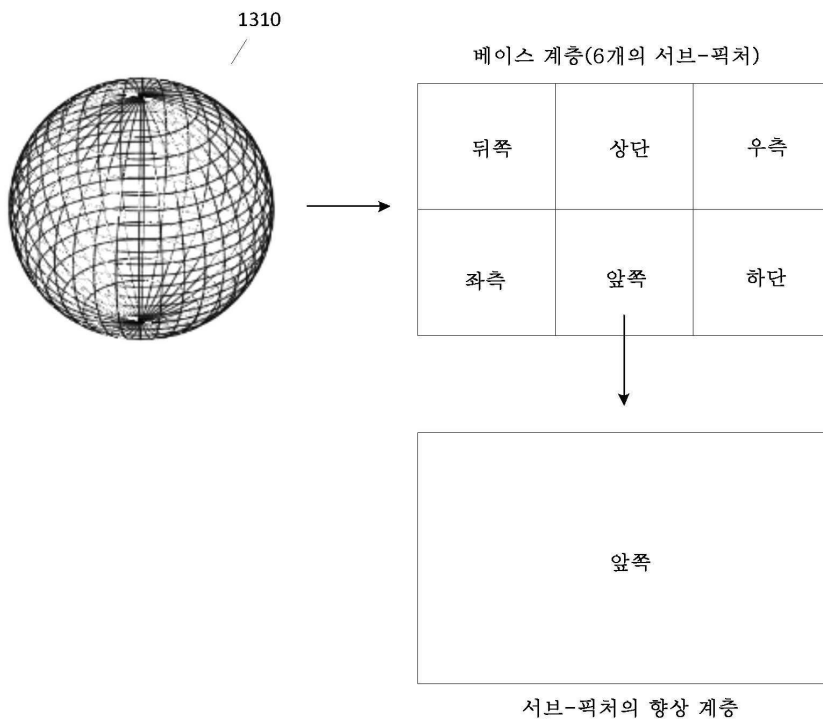
도면11



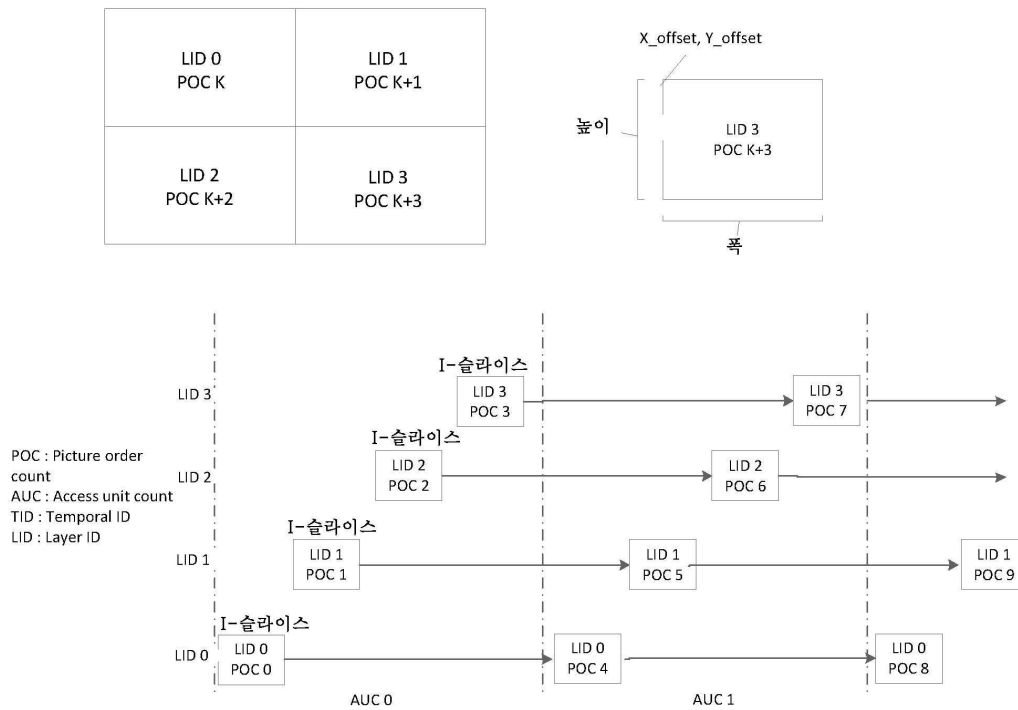
도면12



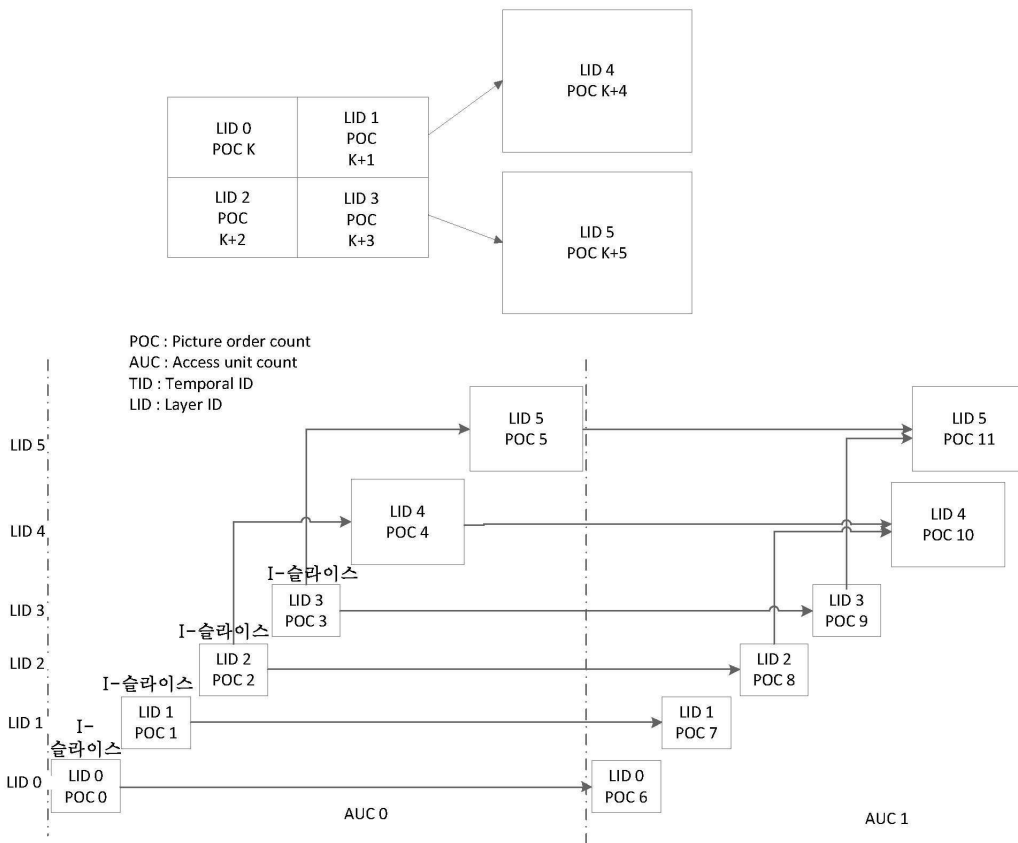
도면13



도면14



도면15



도면16a

video_parameter_set_rbsp() { ^o	디스크립터
vps_video_parameter_set_id ^o	u(4) ^o
vps_max_layers_minus1 ^o	u(8) ^o
for(i = 0; i <= vps_max_layers_minus1; i++) { ^o	^o
vps_included_layer_id[i] ^o	u(7) ^o
vps_reserved_zero_bit ^o	u(1) ^o
} ^o	^o
... ^o	^o
vps_sub_picture_dividing_flag ^o	u(1) ^o
if(vps_sub_picture_dividing_flag) { ^o	^o
vps_full_pic_width_in_luma_samples ^o	ue(v) ^o
vps_full_pic_height_in_luma_samples ^o	ue(v) ^o
} ^o	^o
... ^o	^o
} ^o	^o

도면16b

seq_parameter_set_rbsp() { ^o	디스크립터
sps_decoding_parameter_set_id ^o	u(4) ^o
sps_video_parameter_set_id ^o	u(4) ^o
sps_max_sub_layers_minus1 ^o	u(3) ^o
... ^o	^o
pic_width_in_luma_samples ^o	ue(v) ^o
pic_height_in_luma_samples ^o	u<(v)< ^o
if(vps_sub_picture_dividing_flag) { ^o	^o
pic_offset_x ^o	ue(v) ^o
pic_offset_y ^o	ue(v) ^o
} ^o	^o
... ^o	^o
} ^o	^o

도면17

sub_region_partitioning_info(payloadSize) { ^o	디스크립터
num_sub_region ^o	ue(v) ^o
num_layers ^o	ue(v) ^o
for(i = 0; i <= num_layers; i++) ^o	^o
layer_id[i] ^o	ue(v) ^o
for(i = 1; i < num_layers; i++) ^o	^o
for(j = 0; j < i; j++) ^o	^o
direct_dependency_flag[i][j] ^o	u(1) ^o
for(i = 0; i < num_sub_region; i++) { ^o	^o
num_layers_for_region[i] ^o	ue(v) ^o
for(i = 0; j < num_layers_for_region[i]; j++) ^o	^o
sub_region_layer_id[i][j] ^o	ue(v) ^o
sub_region_offset_x[i] ^o	ue(v) ^o
sub_region_offset_y[i] ^o	ue(v) ^o
sub_region_width[i] ^o	ue(v) ^o
sub_region_height[i] ^o	ue(v) ^o
} ^o	^o
... ^o	^o
} ^o	^o

도면18

video parameter set rbsp() {	디스크립터
...	
vps max layers minus1	u(6)
num output layer sets	ue(v)
num profile tier level	ue(v)
for(i = 0; i < num profile tier level; i++)	
profile tier level(vps max sub layers minus1)	
for(i = 0; i < num output layer sets; i++)	
for(j = 0; j < NumLayersInIdList[i]; j++) {	
output layer flag[i][j]	u(1)
profile tier level idx[i][j]	u(v)
}	
...	
}	

도면19

video parameter set rbsp() {	디스크립터
...	
vps max layers minus1	u(6)
if(vps max layers minus1 > 0) {	
num output layer sets	ue(v)
num profile tier level	ue(v)
}	
for(i = 0; i < num profile tier level; i++)	
profile tier level(vps max sub layers minus1)	
for(i = 0; i < num output layer sets; i++) {	
vps output layers mode[i]	u(2)
vps ptl signal flag[i]	u(1)
for(j = 0; j < NumLayersInIdList[i]; j++) {	
if(vps output layers mode[i] == 2)	
output layer flag[i][j]	u(1)
if(vps ptl signal flag[i])	
profile tier level idx[i][j]	u(v)
}	
}	
...	
}	

도면20

video parameter set rbsp() {	디스크립터
...	
vps max layers minus1	u(6)
if(vps max layers minus1 > 0) {	
num output layer sets	ue(v)
num profile tier level	ue(v)
}	
max subpics minus1	u(8)
for(i = 0; i < max subpics minus1; i++) {	
sub pic id[i]	u(8)
}	
for(i = 0; i < num profile tier level; i++)	
profile tier level(vps max sub layers minus1)	
for(i = 0; i < num output layer sets; i++) {	
vps output layers mode[i]	u(2)
vps ptl signal flag[i]	u(1)
for(j = 0; j < NumLayersInIdList[i]; j++) {	
num output subpic layer[i][j]	ue(v)
for(k = 0; k < num output subpic layer[i][j]; k++)	
sub pic id layer[i][j][k]	u(8)
if(vps output layers mode[i] == 2)	
output layer flag[i][j]	u(1)
if(vps ptl signal flag[i])	
profile tier level idx[i][j]	u(v)
}	
}	
...	
}	

도면21

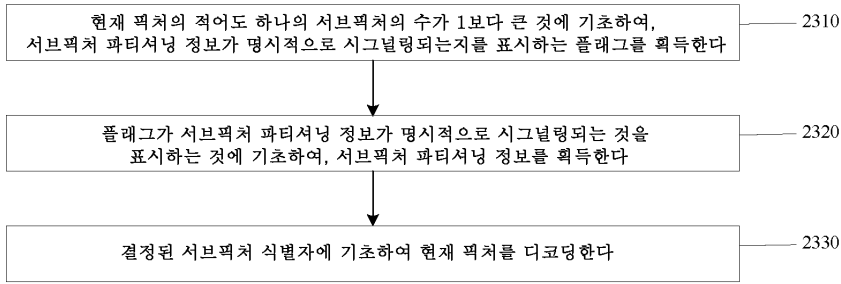
pic parameter set rbsp() {	디스크립터
pps pic parameter set id	ue(v)
pps seq parameter set id	u(4)
mixed_nalu_types_in_pic_flag	u(1)
pic width in luma samples	ue(v)
pic height in luma samples	ue(v)
pps conformance window flag	u(1)
if(pps conformance window flag) {	
pps conf win left offset	ue(v)
pps conf win right offset	ue(v)
pps conf win top offset	ue(v)
pps conf win bottom offset	ue(v)
}	
scaling window explicit signalling flag	u(1)
if(scaling window explicit signalling flag) {	
scaling win left offset	ue(v)
scaling win right offset	ue(v)
scaling win top offset	ue(v)
scaling win bottom offset	ue(v)
}	
output flag present flag	u(1)
subpic id mapping in pps flag	u(1)
if(subpic id mapping in pps flag) {	
pps num subpics minus1	ue(v)
pps subpic id len minus1	ue(v)
for(i = 0; i <= pps num subpic minus1; i++)	
pps subpic id[i]	u(v)
}	
...	
}	

도면22

seq_parameter_set_rbsp() {	디스크립터
...	
subpic_info_present_flag	u(1)
if(subpic_info_present_flag) {	
sps_num_subpics_minus2	ue(v)
sps_independent_subpics_flag	u(1)
for(i = 0; i <= sps_num_subpics_minus2 + 1; i++) {	
if(i > 0 && pic_width_max_in_luma_samples > CtbSizeY)	
subpic_ctu_top_left_x[i]	u(v)
if(i > 0 && pic_height_max_in_luma_samples > CtbSizeY) {	
subpic_ctu_top_left_y[i]	u(v)
if(i < sps_num_subpics_minus2 + 1 && pic_width_max_in_luma_samples > CtbSizeY)	
subpic_width_minus1[i]	u(v)
if(i < sps_num_subpics_minus2 + 1 && pic_height_max_in_luma_samples > CtbSizeY)	
subpic_height_minus1[i]	u(v)
if(!sps_independent_subpics_flag) {	
subpic_treated_as_pic_flag[i]	u(1)
loop_filter_across_subpic_enabled_flag[i]	u(1)
}	
}	
sps_subpic_id_len_minus1	ue(v)
subpic_id_mapping_explicitly_signalled_flag	u(1)
if(subpic_id_mapping_explicitly_signalled_flag) {	
subpic_id_mapping_in_sps_flag	u(1)
if(subpic_id_mapping_in_sps_flag)	
for(i = 0; i <= sps_num_subpics_minus2 + 1; i++)	
sps_subpic_id[i]	u(v)
}	
}	
...	
}	

도면23

2300



도면24

