



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2016-0072123
(43) 공개일자 2016년06월22일

- (51) 국제특허분류(Int. Cl.)
HO4N 19/30 (2014.01) *HO4N 19/187* (2014.01)
HO4N 19/423 (2014.01) *HO4N 19/52* (2014.01)
- (52) CPC특허분류
HO4N 19/30 (2015.01)
HO4N 19/187 (2015.01)
- (21) 출원번호 10-2016-7010352
- (22) 출원일자(국제) 2014년10월14일
 심사청구일자 없음
- (85) 번역문제출일자 2016년04월20일
- (86) 국제출원번호 PCT/US2014/060483
- (87) 국제공개번호 WO 2015/057704
 국제공개일자 2015년04월23일
- (30) 우선권주장
 61/890,782 2013년10월14일 미국(US)
 14/513,110 2014년10월13일 미국(US)

- (71) 출원인
 웰컴 인코포레이티드
 미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 발명자
 라마수브라모니안 아다르쉬 크리쉬난
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 왕 예-쿠이
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 천 잉
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
- (74) 대리인
 특허법인코리아나

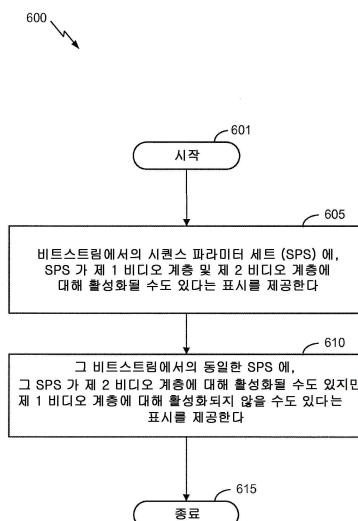
전체 청구항 수 : 총 25 항

(54) 발명의 명칭 **비디오 정보의 스케일러블 코딩을 위한 디바이스 및 방법**

(57) 요약

비디오 정보를 코딩하도록 구성된 장치는 메모리 유닛 및 메모리 유닛과 통신하는 프로세서를 포함한다. 메모리 유닛은 제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하도록 구성된다. 프로세서는 비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 것으로서, 파라미터 세트의 제 1 인스턴스는 제 1 비디오 계층 및 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하고, 그리고 비트스트림에서 파라미터 세트의 제 2 인스턴스를 프로세싱하는 것으로서, 파라미터 세트의 제 2 인스턴스는 : (1) 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있고, 그리고 (2) 제 1 비디오 계층이 파라미터 세트를 이용하지 않을 수도 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하도록 구성된다. 프로세서는 비디오 정보를 인코딩하거나 디코딩할 수도 있다.

대 표 도 - 도6



(52) CPC특허분류

H04N 19/423 (2015.01)

H04N 19/52 (2015.01)

명세서

청구범위

청구항 1

비디오 정보를 코딩하도록 구성된 장치로서,

제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하도록 구성된 메모리 유닛; 및
상기 메모리 유닛과 통신하는 프로세서

를 포함하며, 상기 프로세서는 :

비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 것으로서, 상기 파라미터 세트의 상기 제 1 인스턴스는 상기 제 1 비디오 계층 및 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하고; 그리고

상기 비트스트림에서의 상기 파라미터 세트의 제 2 인스턴스를 프로세싱하는 것으로서, 상기 파라미터 세트의 상기 제 2 인스턴스는 : (1) 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있고, 그리고 (2) 상기 제 1 비디오 계층이 상기 파라미터 세트를 이용하지 않을 수 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하도록

구성되는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 2

제 1 항에 있어서,

상기 파라미터 세트의 상기 제 2 인스턴스의 상기 제 2 표시는 상기 파라미터 세트의 상기 제 1 인스턴스에서 제공되지 않는 적어도 하나의 플래그 또는 신택스 엘리먼트를 포함하는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 3

제 1 항에 있어서,

상기 제 2 인스턴스의 상기 제 2 표시는 상기 제 1 인스턴스의 상기 제 1 표시에 의해 규정된 계층 ID 와는 상이한 계층 ID 를 규정하는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 4

제 1 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 공통 파라미터 세트 ID 를 규정하는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 5

제 1 항에 있어서,

상기 파라미터 세트는 비디오 파라미터 세트 (VPS), 시퀀스 파라미터 세트 (SPS), 팩쳐 파라미터 세트 (PPS), 또는 슬라이스 헤더 중 하나인, 비디오 정보를 코딩하도록 구성된 장치.

청구항 6

제 1 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 제 1 및 제 2 네트워크 추상화 계층 (NAL) 유닛들을 각각 포함하는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 7

제 6 항에 있어서,

상기 제 1 NAL 유닛은 제 1 미가공 바이트 시퀀스 페이로드 (RBSP) 콘텐츠 및 제 1 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 2 NAL 유닛은 제 2 RBSP 콘텐츠 및 제 2 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 1 및 제 2 RBSP 콘텐츠들은 동일하며, 상기 제 1 및 제 2 NAL 유닛 헤더들 콘텐츠들은 상이한, 비디오 정보를 코딩하도록 구성된 장치.

청구항 8

제 1 항에 있어서,

상기 비디오 정보를 코딩하도록 구성된 장치는 인코더를 포함하며, 상기 프로세서는 상기 비트스트림에서의 상기 제 1 및 제 2 인스턴스들을 제공함으로써 상기 비트스트림에서의 상기 제 1 및 제 2 인스턴스들을 프로세싱하도록 구성되며, 상기 프로세서는 또한, 상기 비트스트림에서의 상기 비디오 정보를 인코딩하도록 구성되는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 9

제 1 항에 있어서,

상기 비디오 정보를 코딩하도록 구성된 장치는 디코더를 포함하며, 상기 프로세서는 상기 비트스트림에서의 상기 제 1 및 제 2 인스턴스들을 수신함으로써 상기 비트스트림에서의 상기 제 1 및 제 2 인스턴스들을 프로세싱하도록 구성되며, 상기 프로세서는 또한, 상기 비트스트림에서의 상기 비디오 정보를 디코딩하도록 구성되는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 10

제 1 항에 있어서,

상기 비디오 정보를 코딩하도록 구성된 장치는 : 컴퓨터, 노트북, 랩탑 컴퓨터, 태블릿 컴퓨터, 셋-탑 박스, 전화기 핸드셋, 스마트 폰, 스마트 패드, 텔레비전, 카메라, 디스플레이 디바이스, 디지털 미디어 플레이어, 비디오 게이밍 콘솔, 및 자동차용 컴퓨터 중 하나 이상으로 이루어진 그룹으로부터 선택된 디바이스를 포함하는, 비디오 정보를 코딩하도록 구성된 장치.

청구항 11

비디오 정보를 인코딩하는 방법으로서,

비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 단계로서, 상기 파라미터 세트의 상기 제 1 인스턴스는 제 1 비디오 계층 및 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하는 단계; 및

상기 비트스트림에서의 상기 파라미터 세트의 제 2 인스턴스를 프로세싱하는 단계로서, 상기 파라미터 세트의 상기 제 2 인스턴스는 : (1) 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있고, 그리고 (2) 상기 제 1 비디오 계층이 상기 파라미터 세트를 이용하지 않을 수 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하는 단계

를 포함하는, 비디오 정보를 인코딩하는 방법.

청구항 12

제 11 항에 있어서,

상기 파라미터 세트의 상기 제 2 인스턴스의 상기 제 2 표시는 상기 파라미터 세트의 상기 제 1 인스턴스에서 제공되지 않는 적어도 하나의 플래그 또는 신택스 엘리먼트를 포함하는, 비디오 정보를 인코딩하는 방법.

청구항 13

제 11 항에 있어서,

상기 제 2 인스턴스의 상기 제 2 표시는 상기 제 1 인스턴스의 상기 제 1 표시에 의해 규정된 계층 ID 와는 상

이한 계층 ID 를 규정하는, 비디오 정보를 인코딩하는 방법.

청구항 14

제 11 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 공통 파라미터 세트 ID 를 규정하는, 비디오 정보를 인코딩하는 방법.

청구항 15

제 11 항에 있어서,

상기 파라미터 세트는 비디오 파라미터 세트 (VPS), 시퀀스 파라미터 세트 (SPS), 퍽처 파라미터 세트 (PPS), 또는 슬라이스 헤더 중 하나인, 비디오 정보를 인코딩하는 방법.

청구항 16

제 11 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 제 1 및 제 2 네트워크 추상화 계층 (NAL) 유닛들을 각각 포함하는, 비디오 정보를 인코딩하는 방법.

청구항 17

제 16 항에 있어서,

상기 제 1 NAL 유닛은 제 1 미가공 바이트 시퀀스 페이로드 (RBSP) 콘텐츠 및 제 1 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 2 NAL 유닛은 제 2 RBSP 콘텐츠 및 제 2 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 1 및 제 2 RBSP 콘텐츠들은 동일하며, 상기 제 1 및 제 2 NAL 유닛 헤더들 콘텐츠들은 상이한, 비디오 정보를 인코딩하는 방법.

청구항 18

코드를 포함하는 비-일시적 컴퓨터 판독가능 매체로서,

상기 코드는, 실행될 때, 장치로 하여금, 프로세스를 수행하게 하며, 상기 프로세스는 :

제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하는 단계;

비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 단계로서, 상기 파라미터 세트의 상기 제 1 인스턴스는 상기 제 1 비디오 계층 및 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하는 단계; 및

상기 비트스트림에서의 상기 파라미터 세트의 제 2 인스턴스를 프로세싱하는 단계로서, 상기 파라미터 세트의 상기 제 2 인스턴스는 : (1) 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있고, 그리고 (2) 상기 제 1 비디오 계층이 상기 파라미터 세트를 이용하지 않을 수 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하는 단계

를 포함하는, 비-일시적 컴퓨터 판독가능 매체.

청구항 19

제 18 항에 있어서,

상기 제 2 인스턴스의 상기 제 2 표시는 상기 제 1 인스턴스의 상기 제 1 표시에 의해 규정된 계층 ID 와는 상이한 계층 ID 를 규정하는, 비-일시적 컴퓨터 판독가능 매체.

청구항 20

제 18 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 공통 파라미터 세트 ID 를 규정하는, 비-일시적 컴퓨터 판독가능 매체.

청구항 21

제 18 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 제 1 및 제 2 네트워크 추상화 계층 (NAL) 유닛들을 각각 포함하며, 상기 제 1 NAL 유닛은 제 1 미가공 바이트 시퀀스 페이로드 (RBSP) 콘텐츠 및 제 1 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 2 NAL 유닛은 제 2 RBSP 콘텐츠 및 제 2 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 1 및 제 2 RBSP 콘텐츠들은 동일하며, 상기 제 1 및 제 2 NAL 유닛 헤더들 콘텐츠들은 상이한, 비-일시적 컴퓨터 판독가능 매체.

청구항 22

비디오 정보를 코딩하도록 구성된 비디오 코딩 디바이스로서,

제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하는 수단;

비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 수단으로서, 상기 파라미터 세트의 상기 제 1 인스턴스는 상기 제 1 비디오 계층 및 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하는 수단; 및

상기 비트스트림에서의 상기 파라미터 세트의 제 2 인스턴스를 프로세싱하는 수단으로서, 상기 파라미터 세트의 상기 제 2 인스턴스는 : (1) 상기 제 2 비디오 계층이 상기 파라미터 세트를 이용할 수 있고, 그리고 (2) 상기 제 1 비디오 계층이 상기 파라미터 세트를 이용하지 않을 수 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하는 수단

을 포함하는, 비디오 코딩 디바이스.

청구항 23

제 22 항에 있어서,

상기 제 2 인스턴스의 상기 제 2 표시는 상기 제 1 인스턴스의 상기 제 1 표시에 의해 규정된 계층 ID 와는 상이한 계층 ID 를 규정하는, 비디오 코딩 디바이스.

청구항 24

제 22 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 공통 파라미터 세트 ID 를 규정하는, 비디오 코딩 디바이스.

청구항 25

제 22 항에 있어서,

상기 제 1 및 제 2 인스턴스들은 제 1 및 제 2 네트워크 추상화 계층 (NAL) 유닛들을 각각 포함하며, 상기 제 1 NAL 유닛은 제 1 미가공 바이트 시퀀스 페이로드 (RBSP) 콘텐츠 및 제 1 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 2 NAL 유닛은 제 2 RBSP 콘텐츠 및 제 2 NAL 유닛 헤더 콘텐츠를 포함하며, 상기 제 1 및 제 2 RBSP 콘텐츠들은 동일하며, 상기 제 1 및 제 2 NAL 유닛 헤더들 콘텐츠들은 상이한, 비디오 코딩 디바이스.

발명의 설명**기술 분야**

[0001] 본 개시물은 비디오 코딩 및 압축의 분야, 특히 스케일러블 비디오 코딩 (SVC), 멀티뷰 비디오 코딩 (MVC), 또는 3D 비디오 코딩 (3DV) 에 관한 것이다.

배경 기술

[0002] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 휴대정보 단말기들 (PDA들), 랩탑 또는 데스크탑 컴퓨터들, 디지털 카메라들, 디지털 리코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 원격 화상회의 디바이스들 등을 포함한, 광범위한 디바이스들에 포함될 수 있다. 디지털 비디오 디바이스들

은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, AVC (Advanced Video Coding) 에 의해 정의된 표준들, 현재 개발중인 HEVC (High Efficiency Video Coding) 표준, 및 이런 표준들의 확장판들에서 설명되는 것들과 같은, 비디오 압축 기법들을 구현한다. 비디오 디바이스들은 이러한 비디오 코딩 기법들을 구현함으로써, 디지털 비디오 정보를 좀 더 효율적으로 송신하거나, 수신하거나, 인코딩하거나, 디코딩하거나, 및/또는 저장할 수도 있다.

[0003] 비디오 압축 기법들은 비디오 시퀀스들에 내재하는 리던던시를 감소시키거나 또는 제거하기 위해 공간 (인트라-픽처) 예측 및/또는 시간 (인터-픽처) 예측을 수행한다. 블록-기반 비디오 코딩에 있어, 비디오 슬라이스 (예컨대, 비디오 프레임, 비디오 프레임의 부분 등) 는 비디오 블록들로 파티셔닝될 수도 있으며, 이 비디오 블록들은 또한 트리블록들, 코딩 유닛들 (CU들) 및/또는 코딩 노드들로서 지정될 수도 있다. 픽처의 인트라-코딩된 (I) 슬라이스에서의 비디오 블록들은 동일한 픽처의 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 픽처의 인터-코딩된 (P 또는 B) 슬라이스에서의 비디오 블록들은 동일한 픽처의 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측, 또는 다른 참조 픽처들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 픽처들은 프레임들로서 지정될 수 있으며, 참조 픽처들은 참조 프레임들로서 지정될 수도 있다.

[0004] 공간 또는 시간 예측은 코딩되는 블록에 대한 예측 블록을 초래한다. 잔차 데이터는 코딩되는 원래 블록과 예측 블록 사이의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 가리키는 모션 벡터, 및 코딩된 블록과 예측 블록 사이의 차이를 나타내는 잔차 데이터에 따라서 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라서 인코딩된다. 추가적인 압축을 위해, 잔차 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔차 변환 계수들을 초래할 수도 있으며, 이 잔차 변환 계수들은 그후 양자화될 수도 있다. 처음에 2차원 어레이로 배열된, 양자화된 변환 계수들은 변환 계수들의 1차원 벡터를 생성하기 위해 스캐닝될 수도 있으며, 엔트로피 인코딩이 더욱 더 많은 압축을 달성하기 위해 적용될 수도 있다.

발명의 내용

과제의 해결 수단

[0005] 스케일러블 비디오 코딩 (SVC) 은 참조 계층 (RL) 으로서 종종 지정되는 기초 계층 (BL) 및 하나 이상의 스케일러블 향상 계층들 (EL들) 이 사용되는 비디오 코딩을 지정한다. SVC 에서, 기초 계층은 베이스 레벨의 품질을 가진 비디오 데이터를 반송 (carry) 할 수 있다. 하나 이상의 향상 계층들은 예를 들어, 더 높은 공간, 시간, 및/또는 신호-대-잡음 (SNR) 레벨들을 지원하기 위해 추가적인 비디오 데이터를 반송할 수 있다. 향상 계층들은 이전에 인코딩된 계층에 대해 정의될 수도 있다. 예를 들어, 최하부 계층은 BL로서 기능할 수도 있으며, 반면 최상부 계층은 EL로서 기능할 수도 있다. 중간 계층들은 EL들 또는 RL들 중 어느 하나, 또는 양쪽으로서 기능할 수도 있다. 예를 들어, 중간 계층 (예컨대, 최하위 계층도 최상위 계층도 아닌 층) 은 기초 계층 또는 임의의 개체하는 (intervening) 향상 계층들과 같은, 중간 계층 아래의 계층들에 대해 EL 일 수도 있으며, 동시에 그 중간 계층 상부의 하나 이상의 향상 계층들에 대해 RL로서 기능한다. 이와 유사하게, HEVC 표준의 멀티뷰 또는 3D 확장판에서, 다수의 뷰들이 존재할 수도 있으며, 하나의 뷰의 정보가 또 다른 뷰의 정보 (예컨대, 모션 추정, 모션 벡터 예측 및/또는 다른 리던던시들) 를 코딩 (예컨대, 인코딩 또는 디코딩) 하는데 이용될 수도 있다.

[0006] SVC 에서, 인코더 또는 디코더에 의해 사용되는 파라미터들은 그들이 이용될 수도 있는 코딩 레벨 (예컨대, 비디오-레벨, 시퀀스-레벨, 픽처-레벨, 슬라이스 레벨, 등) 에 기초하여 파라미터 세트들로 그룹화된다. 예를 들어, 비트스트림에서의 하나 이상의 코딩된 비디오 시퀀스들에 의해 이용될 수도 있는 파라미터들은 비디오 파라미터 세트 (VPS) 에 포함될 수도 있으며, 코딩된 비디오 시퀀스에서의 하나 이상의 픽처들에 의해 이용될 수도 있는 파라미터들은 시퀀스 파라미터 세트 (SPS) 에 포함될 수도 있다. 이와 유사하게, 픽처에서의 하나 이상의 슬라이스들에 의해 이용되는 파라미터들은 픽처 파라미터 세트 (PPS) 에 포함될 수도 있으며, 단일 슬라이스에 고유한 다른 파라미터들은 슬라이스 헤더에 포함될 수도 있다. 이와 유사하게, 특정의 계층이 주어진 시간에 어느 파라미터 세트(들)를 이용하고 있는지의 표시가 여러 코딩 레벨들에서 제공될 수도 있다. 예를 들어, 특정의 계층에서의 슬라이스의 슬라이스 헤더는 PPS 를 참조할 수도 있으며, PPS 는 SPS 를 참조할 수도 있으며, SPS 는 VPS 를 참조할 수도 있다. 게다가, 동일한 파라미터 세트들이 (예컨대, 파라미터 세트들을 포함하는 비트스트림의 부분이 손실되면 파라미터 세트들이 복구될 수 있도록) 비트스트림을 통해서 다수회 제공될 수도 있다. 예를 들어, SPS 가 비트스트림 내에서 다수회 반복될 때, 이러한 SPS 는 반복된 SPS

로서 지칭될 수도 있다.

[0007] 그러나, 일부 코딩 방식들에서, 하나 이상의 비트스트림 제약들은 동일한 SPS 의 모든 인스턴스들이 동일하여야 한다는 것을 규정할 수도 있다. 예를 들어, 각각의 SPS 는 활성화될 SPS 의 SPS ID 에 추가하여, 계층 ID, 시간 서브-계층 ID, 등을 포함할 수도 있다. 따라서, 이러한 비트스트림 제약들을 만족시키기 위해, SPS 의 각각의 인스턴스 (예컨대, 동일한 SPS ID 를 가지는 다수의 인스턴스들) 는 계층 ID 와 같은 동일한 콘텐츠를 가져야 하며, 그 결과, 반복된 파라미터 세트들의 사용이, 특히, 파라미터 세트들이 다수의 계층들에 걸쳐서 공유될 때, 제한될 수도 있다. 예를 들어, 코더는 이러한 비트스트림 제약들이 적용가능하다고 결정하고, 비트스트림에서의 비디오 정보를 코딩할 때 비트스트림 제약들을 준수할 수도 있다.

[0008] 따라서, 반복된 파라미터 세트들의 더욱 비제한된 사용을 가능케 함으로써 에러 복원력이 향상될 수 있도록 하는 코딩 방식이 요망된다.

[0009] 본 개시물의 시스템들, 방법들 및 디바이스들은 여러 혁신적인 양태들을 각각 가지며, 그 중 어떤 단 하나의 양태도 본원에서 개시된 바람직한 속성들을 단독으로 담당하지 않는다.

[0010] 일 양태에서, 비디오 정보를 코딩하도록 (예컨대, 인코딩 또는 디코딩하도록) 구성된 장치는 메모리 유닛 및 메모리 유닛과 통신하는 프로세서를 포함한다. 메모리 유닛은 제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하도록 구성된다. 프로세서는 비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 것으로서, 파라미터 세트의 제 1 인스턴스는 제 1 비디오 계층 및 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하고; 그리고 비트스트림에서의 파라미터 세트의 제 2 인스턴스를 프로세싱하는 것으로서, 파라미터 세트의 제 2 인스턴스는 : (1) 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있고, 그리고 (2) 제 1 비디오 계층이 파라미터 세트를 이용하지 않을 수도 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하도록 구성된다.

[0011] 다른 양태에서, 비디오 정보를 인코딩하는 방법은, 비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 단계로서, 파라미터 세트의 제 1 인스턴스는 제 1 비디오 계층 및 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하는 단계, 및 비트스트림에서의 파라미터 세트의 제 2 인스턴스를 프로세싱하는 단계로서, 파라미터 세트의 제 2 인스턴스는 : (1) 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있고, 그리고 (2) 제 1 비디오 계층이 파라미터 세트를 이용하지 않을 수도 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하는 단계를 포함한다.

[0012] 다른 양태에서, 비-일시적 컴퓨터 판독가능 매체는 실행될 때, 장치로 하여금, 프로세스를 수행하게 하는 코드를 포함한다. 프로세스는, 제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하는 단계, 비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 단계로서, 파라미터 세트의 제 1 인스턴스는 제 1 비디오 계층 및 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하는 단계, 및 비트스트림에서의 파라미터 세트의 제 2 인스턴스를 프로세싱하는 단계로서, 파라미터 세트의 제 2 인스턴스는 : (1) 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있고, 그리고 (2) 제 1 비디오 계층이 파라미터 세트를 이용하지 않을 수도 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하는 단계를 포함한다.

[0013] 다른 양태에서, 비디오 정보를 코딩하도록 구성된 비디오 코딩 디바이스는 제 1 비디오 계층 및 제 2 비디오 계층과 연관된 비디오 정보를 저장하는 수단, 비트스트림에서의 파라미터 세트의 제 1 인스턴스를 프로세싱하는 수단으로서, 파라미터 세트의 제 1 인스턴스는 제 1 비디오 계층 및 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있다는 제 1 표시를 포함하는, 상기 제 1 인스턴스를 프로세싱하는 수단, 및 비트스트림에서의 파라미터 세트의 제 2 인스턴스를 프로세싱하는 수단으로서, 파라미터 세트의 제 2 인스턴스는 : (1) 제 2 비디오 계층이 파라미터 세트를 이용할 수도 있고, 그리고 (2) 제 1 비디오 계층이 파라미터 세트를 이용하지 않을 수도 있다는 제 2 표시를 포함하는, 상기 제 2 인스턴스를 프로세싱하는 수단을 포함한다.

도면의 간단한 설명

[0014] 도 1a 는 본 개시물에서 설명하는 양태들에 따른 기법들을 이용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 1b 는 본 개시물에서 설명하는 양태들에 따른 기법들을 수행할 수도 있는 다른 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2a 는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 비디오 인코더의 예를 예시하는 블록도이다.

도 2b 는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 비디오 인코더의 예를 예시하는 블록도이다.

도 3a 는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록도이다.

도 3b 는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록도이다.

도 4 는 본 개시물의 일 실시형태에 따른, 상이한 계층들에서의 꾹처들의 예시적인 구성을 예시하는 블록도이다.

도 5 는 본 개시물의 일 실시형태에 따른, 비디오 정보를 코딩하는 방법을 예시하는 플로우 차트이다.

도 6 는 본 개시물의 일 실시형태에 따른, 비디오 정보를 코딩하는 방법을 예시하는 플로우 차트이다.

발명을 실시하기 위한 구체적인 내용

[0015]

본원에서 설명된 어떤 실시형태들은 HEVC (High Efficiency Video Coding) 와 같은, 진보된 비디오 코덱들의 맥락에서, 스케일러블 비디오 코딩을 위한 인터-계층 예측에 관한 것이다. 좀더 구체적으로는, 본 개시물은 HEVC 의 스케일러블 비디오 코딩 (SVC) 확장판에서 인터-계층 예측의 향상된 성능을 위한 시스템들 및 방법들에 관한 것이다.

[0016]

하기 설명에서, 어떤 실시형태들에 관련된 H.264/AVC 기법들이 설명되며; HEVC 표준 및 관련된 기법들이 또한 논의된다. 어떤 실시형태들은 본원에서 HEVC 및/또는 H.264 표준들의 맥락에서 설명되지만, 당업자는 본원에서 개시된 시스템들 및 방법들은 임의의 적합한 비디오 코딩 표준에 적용가능할 수도 있음을 인식할 수도 있다. 예를 들어, 본원에서 개시된 실시형태들은 다음 표준들 중 하나 이상에 적용가능할 수도 있다: ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 또는 ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual 및 그의 스케일러블 비디오 코딩 (SVC) 및 멀티뷰 비디오 코딩 (MVC) 확장판들을 포함한, (또한, ISO/IEC MPEG-4 AVC 로서 알려진) ITU-T H.264.

[0017]

HEVC 는 일반적으로 많은 측면들에서 이전 비디오 코딩 표준들의 프레임워크를 따른다. HEVC 에서 예측의 유닛은 어떤 이전 비디오 코딩 표준들에서의 유닛 (예컨대, 매크로블록) 과는 상이하다. 사실, 매크로블록의 컨셉은 어떤 이전 비디오 코딩 표준들에서 이해되는 바와 같이 HEVC 에 존재하지 않는다. 매크로블록은 다른 가능한 이익들 중, 높은 유연성을 제공할 수도 있는 퀘드트리 방식에 기초하여, 계층적 구조로 대체된다. 예를 들어, HEVC 방식 내에서, 블록들의 3개의 유형들, 즉, 코딩 유닛 (CU), 예측 유닛 (PU), 및 변환 유닛 (TU) 이 정의된다. CU 는 영역 분할 (splitting) 의 기본 유닛을 지칭할 수도 있다. CU 는 매크로블록의 컨셉과 유사한 것으로 간주될 수도 있으며, 그러나 HEVC 는 CU들의 최대 사이즈를 제한하지 않으며, 4개의 동일한 사이즈 CU들로의 회귀적인 분할을 가능하게 하여 콘텐츠 적응성을 향상시킬 수도 있다. PU 는 인터/인트라 예측의 기본 유닛으로서 간주될 수도 있으며, 단일 PU 는 불규칙적인 이미지 패턴들을 효과적으로 코딩하기 위해 다수의 임의의 형태 파티션들을 포함할 수도 있다. TU 는 변환의 기본 유닛으로서 간주될 수도 있다. TU 는 PU 와 독립적으로 정의될 수 있으며; 그러나, TU 의 사이즈는 TU 가 속하는 CU 의 사이즈에 한정될 수도 있다. 이 3개의 상이한 컨셉들로의 블록 구조의 분리는 각각의 유닛이 그 유닛의 개개의 역할에 따라서 각각 최적화될 수 있게 함으로써, 향상된 코딩 효율을 초래할 수도 있다.

[0018]

단지 예시의 목적을 위해, 어떤 본원에서 개시된 실시형태들은 단지 2개의 계층들 (예컨대, 기초 계층과 같은 하위 계층, 및 향상 계층과 같은 상위 계층) 을 포함하는 예들로 설명된다. 이러한 예들이 다수의 기초 및/또는 향상 계층들을 포함하는 구성들에 적용가능할 수도 있는 것으로 이해되어야 한다. 게다가, 설명의 용이성을 위해, 다음 개시물은 어떤 실시형태들과 관련하여 용어들 "프레임들" 또는 "블록들" 을 포함한다. 그러나, 이를 용어들은 한정하는 것으로 의도되지 않는다. 예를 들어, 아래에서 설명되는 기법들은 블록들 (예컨대, CU, PU, TU, 매크로블록들, 등), 슬라이스들, 프레임들, 등과 같은, 임의의 적합한 비디오 유닛들과 함께 사용될 수 있다.

비디오 코딩 표준들

[0019] 비디오 리코더 또는 컴퓨터에 의해 생성된 이미지, 비디오 이미지, TV 이미지, 또는 스틸 이미지와 같은, 디지털 이미지는 수평 및 수직 라인들에 배열된 픽셀들 또는 샘플들로 이루어질 수도 있다. 단일 이미지에서의 픽셀들의 개수는 일반적으로 수만개이다. 각각의 픽셀은 일반적으로 휘도 (luminance) 및 색차 (chrominance) 정보를 포함한다. 압축이 없다면, 이미지 인코더로부터 이미지 디코더로 운반되는 정보의 전적인 양은 실시간 이미지 송신을 불가능하게 만들 것이다. 송신되는 정보의 양을 감소시키기 위해, JPEG, MPEG 및 H.263 표준들과 같은, 다수의 상이한 압축 방법들이 개발되었다.

[0020] 비디오 코딩 표준들은 ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 또는 ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual 및 그의 스케일러블 비디오 코딩 (SVC) 및 멀티뷰 비디오 코딩 (MVC) 확장판들을 포함한, (또한, ISO/IEC MPEG-4 AVC로서 알려진) ITU-T H.264를 포함한다.

[0021] 게다가, 새로운 비디오 코딩 표준, 즉 고효율 비디오 코딩 (HEVC)이 ITU-T 비디오 코딩 전문가 그룹 (VCEG)과 ISO/IEC 동화상 전문가 그룹 (MPEG)의 비디오 코딩에 관한 합동 연구팀 (JCT-VC)에 의해 개발되고 있다. HEVC 초안 10에 대한 전체 인용은 ITU-T SG16 WP3과 ISO/IEC JTC1/SC29/WG11의 JCT-VC (Joint Collaborative Team on Video Coding), 12차 회의: 2013년 1월 14일 내지 2013년 1월 23일, 스위스, 제네바, 문서 JCTVC-L1003, Bross 등, "High Efficiency Video Coding (HEVC) Text Specification Draft 10"이다. HEVC에 대한 멀티뷰 확장판, 즉 MV-HEVC, 및 SHVC로 지칭되는, HEVC에 대한 스케일러블 확장판이 또한 JCT-3V (ITU-T/ISO/IEC 3D 비디오 코딩 확장판 개발을 위한 합동 작업팀) 및 JCT-VC에 의해 각각 개발 중에 있다.

[0022] 신규한 시스템들, 장치들, 및 방법들의 다양한 양태들은 이하에서 첨부 도면들을 참조하여 좀더 충분히 설명된다. 본 개시물은 그러나, 많은 상이한 형태들로 구현될 수도 있으며, 본 개시물을 통해서 제시되는 임의의 특징의 구조 또는 기능에 한정되는 것으로 해석되어서는 안된다. 대신, 이들 양태들은 본 개시물이 철저하고 완전하게 되도록, 그리고 본 개시물의 범위를 당업자들에게 충분히 전달하기 위해서 제공된다. 본원에서의 교시들에 기초하여, 당업자는 본 개시물의 범위가 본 개시물의 임의의 다른 양태와 독립적으로 구현되든 그와 결합되든, 본원에서 개시된 신규한 시스템들, 장치들, 및 방법들의 임의의 양태를 포괄하도록 의도된다는 것을 인식해야 한다. 예를 들어, 본원에서 개시된 임의의 개수의 양태들을 이용하여, 장치가 구현될 수도 있거나 또는 방법이 실시될 수도 있다. 게다가, 본 개시물의 범위는 본원에서 개시된 본 개시물의 여러 양태들에 추가해서 또는 이 이외에, 다른 구조, 기능, 또는 구조 및 기능을 이용하여 실행되는 장치 또는 방법을 포괄하도록 의도된다. 본원에서 개시된 임의의 양태는 청구항의 하나 이상의 엘리먼트들에 의해 구현될 수도 있는 것으로 이해되어야 한다.

[0023] 특정의 양태들이 본원에서 설명되지만, 이들 양태들의 많은 변형예들 및 치환들은 본 개시물의 범위 이내이다. 바람직한 양태들의 일부 이익들 및 이점들이 언급되지만, 본 개시물의 범위는 특유의 이익들, 용도들, 또는 목적들에 한정되는 것으로 의도되지 않는다. 대신, 본 개시물의 양태들은 상이한 무선 기술들, 시스템 구성들, 네트워크들, 및 송신 프로토콜들에 넓게 적용가능한 것으로 의도되며, 이들 중 일부가 일 예로서 도면들에 그리고 바람직한 양태들의 다음 설명에 예시된다. 상세한 설명 및 도면들은 한정하기 보다는 단지 본 개시물의 예시이며, 본 개시물의 범위는 첨부된 청구범위 및 이의 균등물들에 의해 정의된다.

[0024] 첨부 도면들은 예들을 예시한다. 첨부 도면들에서 참조 번호들로 표시된 엘리먼트들은 다음 설명에서 유사한 참조 번호들로 표시된 엘리먼트들에 대응한다. 본 개시물에서, 서수의 단어들 (예컨대, "제 1", "제 2", "제 3", 및 기타 등등)로 시작하는 이름들을 갖는 엘리먼트들은 엘리먼트들이 특정의 순서를 갖는다는 것을 반드시 암시하지는 않는다. 대신, 이러한 서수의 단어들은 동일한 또는 유사한 유형의 상이한 엘리먼트들을 지칭하기 위해 단지 사용된다.

비디오 코딩 시스템

[0025] 도 1a는 본 개시물에서 설명된 양태들에 따른 기법들을 이용할 수도 있는 예시적인 비디오 코딩 시스템 (10)을 예시하는 블록도이다. 설명되는 본원에서 사용될 때, 용어 "비디오 코더"는 비디오 인코더들 및 비디오 디코더들 양쪽을 포괄적으로 지칭한다. 본 개시물에서, 용어들 "비디오 코딩" 또는 "코딩"은 포괄적으로 비디오 인코딩 및 비디오 디코딩을 지칭할 수도 있다. 비디오 인코더들 및 비디오 디코더들에 대해서, 본 출원에서 설명된 양태들은 트랜스코더들 (예컨대, 비트스트림을 디코딩하고 다른 비트스트림을 재-인코딩할 수 있는 디바이스들) 및 미들박스들 (middleboxes) (예컨대, 비트스트림을 수정하고, 변환하고, 및/또는 아니면 조

작할 수 있는 디바이스들) 과 같은, 다른 관련된 디바이스들로 확장될 수도 있다.

[0028] 도 1a 에 나타낸 바와 같이, 비디오 코딩 시스템 (10) 은 목적지 모듈 (14) 에 의해 추후 디코딩될 인코딩된 비디오 데이터를 생성하는 소스 모듈 (12) 을 포함한다. 도 1a 의 예에서, 소스 모듈 (12) 및 목적지 모듈 (14) 은 별개의 디바이스들 상에 있으며 - 구체적으로 말하면, 소스 모듈 (12) 은 소스 디바이스의 일부이고, 목적지 모듈 (14) 은 목적지 디바이스의 일부이다. 그러나, 소스 및 목적지 모듈들 (12, 14) 이 도 1b 의 예에 나타낸 바와 같이 동일한 디바이스 상에 있거나 또는 그 부분일 수도 있다는 점에 유의한다.

[0029] 또 다시 도 1a 를 참조하면, 소스 모듈 (12) 및 목적지 모듈 (14) 은 데스크탑 컴퓨터들, 노트북 (예컨대, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 세-탑 박스들, 소위 "스마트" 폰들과 같은 전화기 핸드셋들, 소위 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함한, 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 일부 경우들에서, 소스 모듈 (12) 및 목적지 모듈 (14) 은 무선 통신용으로 탑재될 수도 있다.

[0030] 목적지 모듈 (14) 은 디코딩될 인코딩된 비디오 데이터를 링크 (16) 를 통해서 수신할 수도 있다. 링크 (16) 는 인코딩된 비디오 데이터를 소스 모듈 (12) 로부터 목적지 모듈 (14) 로 이동시킬 수 있는 임의 종류의 매체 또는 디바이스를 포함할 수도 있다. 도 1a 의 예에서, 링크 (16) 는 소스 모듈 (12) 로 하여금 인코딩된 비디오 데이터를 직접 목적지 모듈 (14) 로 실시간으로 송신할 수 있게 하는 통신 매체를 포함할 수도 있다.

인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라서 변조되어 목적지 모듈 (14) 로 송신될 수도 있다. 통신 매체는 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적인 송신 라인들과 같은, 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 근거리 네트워크, 광역 네트워크, 또는 클로벌 네트워크, 예컨대 인터넷과 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 모듈 (12) 로부터 목적지 모듈 (14) 로 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0031] 대안적으로, 인코딩된 데이터는 출력 인터페이스 (22) 로부터 옵션적인 저장 디바이스 (31) 로 출력될 수도 있다. 이와 유사하게, 인코딩된 데이터는 입력 인터페이스 (28) 에 의해 저장 디바이스 (31) 로부터 액세스될 수도 있다. 저장 디바이스 (31) 는 하드 드라이브, 플래시 메모리, 휴발성 또는 비-휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 로컬 액세스되는 데이터 저장 매체들 중 임의의 데이터 저장 매체를 포함할 수도 있다. 추가 예에서, 저장 디바이스 (31) 는 소스 모듈 (12) 에 의해 생성된 인코딩된 비디오를 유지할 수도 있는 파일 서버 또는 또 다른 중간 저장 디바이스에 대응할 수도 있다. 목적지 모듈 (14) 은 스트리밍 또는 다운로드를 통해서, 저장 디바이스 (31) 로부터의 저장된 비디오 데이터에 액세스할 수도 있다. 파일 서버는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 모듈 (14) 로 송신하는 것이 가능한 임의 종류의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예컨대, 웹사이트용), FTP 서버, NAS (network attached storage) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 모듈 (14) 은 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서, 인코딩된 비디오 데이터에 액세스할 수도 있다. 이것은 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하는데 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양쪽의 조합을 포함할 수도 있다. 저장 디바이스 (31) 로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이 양쪽의 조합일 수도 있다.

[0032] 본 개시물의 기법들은 무선 애플리케이션들 또는 설정들에 한정되지 않는다. 이 기법들은 오버-디-에어 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예컨대, 인터넷을 통한 스트리밍 비디오 송신들 (예컨대, HTTP 를 통한 동적 적응 스트리밍 (DASH) 등), 데이터 저장 매체 상에의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은, 다양한 멀티미디어 애플리케이션들 중 임의의 애플리케이션의 지원 하에 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 비디오 코딩 시스템 (10) 은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화 통신과 같은, 지원 애플리케이션들로의 1-방향 또는 2-방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0033] 도 1a 의 예에서, 소스 모듈 (12) 은 비디오 소스 (18), 비디오 인코더 (20), 및 출력 인터페이스 (22) 를 포함한다. 일부의 경우, 출력 인터페이스 (22) 는 변조기/복조기 (모뎀) 및/또는 송신기를 포함할 수도 있다.

소스 모듈 (12) 에서, 비디오 소스 (18) 는 비디오 캡쳐 디바이스, 예컨대, 비디오 카메라, 이전에 캡쳐된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 공급 인터페이스,

및/또는 컴퓨터 그래픽스 데이터를 소스 비디오로서 생성하는 컴퓨터 그래픽스 시스템과 같은 소스, 또는 이런 소스들의 조합을 포함할 수도 있다. 일 예로서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 모듈 (12) 및 목적지 모듈 (14) 은 도 1b 의 예에 예시된 바와 같이, 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 그러나, 본 개시물에서 설명하는 기법들은 비디오 코딩에 일반적으로 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다.

[0034] 캡쳐되거나, 사전-캡쳐되거나, 또는 컴퓨터-생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다.

인코딩된 비디오 데이터는 소스 모듈 (12) 의 출력 인터페이스 (22) 를 통해서 목적지 모듈 (14) 에 직접 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 (또한, 대안적으로) 디코딩 및/또는 플레이백을 위해, 목적지 모듈 (14) 또는 다른 디바이스들에 의한 추후 액세스를 위해, 저장 디바이스 (31) 상에 저장될 수도 있다.

도 1a 및 도 1b 에 예시된 비디오 인코더 (20) 는 도 2a 에 예시된 비디오 인코더 (20), 도 2b 에 예시된 비디오 인코더 (23), 또는 본원에서 설명되는 임의의 다른 비디오 인코더를 포함할 수도 있다.

[0035] 도 1a 의 예에서, 목적지 모듈 (14) 은 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 일부의 경우, 입력 인터페이스 (28) 는 수신기 및/또는 모뎀을 포함할 수도 있다. 목적지 모듈 (14) 의 입력 인터페이스 (28) 는 링크 (16) 를 통해서 인코딩된 비디오 데이터를 수신할 수도 있다. 링크 (16) 를 통해서 통신되거나, 또는 저장 디바이스 (31) 상에 제공되는 인코딩된 비디오 데이터는, 비디오 데이터를 디코딩할 때에, 비디오 디코더 (30) 와 같은 비디오 디코더에 의해 사용하기 위한, 비디오 인코더 (20) 에 의해 생성되는 다양한 신택스 엘리먼트들을 포함할 수도 있다. 이런 신택스 엘리먼트들은 통신 매체 상으로 송신되거나, 저장 매체 상에 저장되거나, 또는 파일 서버에 저장된 인코딩된 비디오 데이터와 함께 포함될 수도 있다. 도 1a 및 도 1b 에 예시된 비디오 디코더 (30) 는 도 3a 에 예시된 비디오 디코더 (30), 도 3b 에 예시된 비디오 디코더 (33), 또는 본원에서 설명되는 임의의 다른 비디오 디코더를 포함할 수도 있다.

[0036] 디스플레이 디바이스 (32) 는 목적지 모듈 (14) 과 통합되거나 또는 그 외부에 있을 수도 있다. 일부 예들에서, 목적지 모듈 (14) 은 통합된 디스플레이 디바이스를 포함하며, 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 모듈 (14) 은 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 그 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 또 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0037] 관련된 양태들에서, 도 1b 는 예시적인 비디오 인코딩 및 디코딩 시스템 (10') 을 나타내며, 여기서 소스 및 목적지 모듈들 (12, 14) 은 디바이스 또는 사용자 디바이스 (11) 상에 있거나 또는 그 부분이다. 디바이스 (11) 는 "스마트" 폰 또는 기타 등등과 같은, 전화기 핸드셋일 수도 있다. 디바이스 (11) 는 소스 및 목적지 모듈들 (12, 14) 과 동작적 통신하는 옵션적인 제어기/프로세서 모듈 (13) 을 포함할 수도 있다. 도 1b 의 시스템 (10') 은 비디오 인코더 (20) 와 출력 인터페이스 (22) 사이의 비디오 프로세싱 유닛 (21) 을 더 포함할 수 있다. 일부 구현예들에서, 비디오 프로세싱 유닛 (21) 은 도 1b 에 예시된 바와 같이, 별개의 유닛이며; 그러나, 다른 구현예들에서, 비디오 프로세싱 유닛 (21) 은 비디오 인코더 (20) 의 일부 및/또는 프로세서/제어기 모듈 (13) 로서 구현될 수 있다. 시스템 (10') 은 또한 비디오 시퀀스에서 관심 오브젝트를 추적할 수 있는 옵션적인 추적기 (29) 를 포함할 수도 있다. 추적될 관심 오브젝트는 하나 이상의 본 개시물의 양태들과 관련하여 설명된 기법에 의해 세그먼트될 수도 있다. 관련된 양태들에서, 추적 (tracking) 은 디스플레이 디바이스 (32) 에 의해, 단독으로 또는 추적기 (29) 와 함께 수행될 수도 있다. 도 1b 의 시스템 (10'), 및 그의 구성요소들은, 그렇지 않으면, 도 1a 의 시스템 (10), 및 그의 구성요소들과 유사하다.

[0038] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 현재 개발중인 HEVC (High Efficiency Video Coding) 표준과 같은 비디오 압축 표준에 따라서 동작할 수도 있으며, HEVC 테스트 모델 (HM) 을 따를 수도 있다. 이의 대안으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG 4, 파트 10, AVC (Advanced Video Coding) 로서 대안적으로 지정되는 ITU-T H.264 표준, 또는 이런 표준들의 확장판들과 같은 다른 사유 (proprietary) 또는 산업 표준들에 따라서 동작할 수도 있다. 본 개시물의 기법들은, 그러나, 임의의 특정의 코딩 표준에 한정되지 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 을 포함한다.

[0039] 도 1a 및 도 1b 의 예들에 나타내지는 않지만, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양쪽의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기에 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다.

적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대, 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.

[0040] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 임의의 이들의 조합들과 같은, 다양한 적합한 인코더 회로 중 임의의 회로로 구현될 수도 있다. 이 기법들이 소프트웨어로 부분적으로 구현되는 경우, 디바이스는 본 개시물의 기법들을 수행하기 위해 소프트웨어용 명령들을 적합한 비일시적 컴퓨터 판독가능 매체에 저장하고, 그 명령들을 하드웨어에서 하나 이상의 프로세서들을 이용하여 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 쪽이든 각각 디바이스에서 결합된 인코더/디코더 (CODEC) 의 부분으로서 통합될 수도 있다.

비디오 코딩 프로세스

[0042] 위에서 간단히 언급한 바와 같이, 비디오 인코더 (20) 는 비디오 데이터를 인코딩한다. 비디오 데이터는 하나 이상의 픽처들을 포함할 수도 있다. 픽처들의 각각은 비디오의 일부를 형성하는 스틸 이미지이다. 일부의 경우, 픽처는 비디오 "프레임" 으로서 지칭될 수도 있다. 비디오 인코더 (20) 가 비디오 데이터를 인코딩할 때, 비디오 인코더 (20) 는 비트스트림을 생성할 수도 있다. 비트스트림은 비디오 데이터의 코딩된 표현을 형성하는 비트들의 시퀀스를 포함할 수도 있다. 비트스트림은 코딩된 픽처들 및 연관된 데이터를 포함할 수도 있다. 코딩된 픽처는 픽처의 코딩된 표현이다.

[0043] 비트스트림을 생성하기 위해, 비디오 인코더 (20) 는 비디오 데이터에서 각각의 픽처에 관해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 픽처들에 관해 인코딩 동작들을 수행할 때, 비디오 인코더 (20) 는 일련의 코딩된 픽처들 및 연관되는 데이터를 생성할 수도 있다. 연관되는 데이터는 비디오 파라미터 세트들 (VPS), 시퀀스 파라미터 세트들, 픽처 파라미터 세트들, 적응 파라미터 세트들, 및 다른 신택스 구조들을 포함할 수도 있다. 시퀀스 파라미터 세트 (SPS) 는 0개 이상의 픽처들의 시퀀스들에 적용가능한 파라미터들을 포함할 수도 있다. 픽처 파라미터 세트 (PPS) 는 0개 이상의 픽처들에 적용가능한 파라미터들을 포함할 수도 있다. 적응 파라미터 세트 (APS) 는 0개 이상의 픽처들에 적용가능한 파라미터들을 포함할 수도 있다. APS 에서의 파라미터들은 PPS 에서의 파라미터들보다 변할 가능성이 더 많은 파라미터들일 수도 있다.

[0044] 코딩된 픽처를 생성하기 위해, 비디오 인코더 (20) 는 픽처를 동일-사이즈로된 비디오 블록들로 파티셔닝할 수도 있다. 비디오 블록은 샘플들의 2차원 어레이일 수도 있다. 비디오 블록들의 각각은 트리블록과 연관된다. 일부의 경우, 트리블록은 최대 코딩 유닛 (LCU) 으로서 지칭될 수도 있다. HEVC 의 트리블록들은 H.264/AVC 와 같은 이전 표준들의 매크로블록들과 대략적으로 유사할 수도 있다. 그러나, 트리블록은 특정의 사이즈에 반드시 한정되지 않으며, 하나 이상의 코딩 유닛들 (CU들) 을 포함할 수도 있다. 비디오 인코더 (20) 는 큐드트리 파티셔닝을 이용하여, 트리블록들의 비디오 블록들을 CU들과 연관되는 비디오 블록들, 따라서 이를 "트리블록들" 로 파티셔닝할 수도 있다.

[0045] 일부 예들에서, 비디오 인코더 (20) 는 픽처를 복수의 슬라이스들로 파티셔닝할 수도 있다. 슬라이스들의 각각은 정수의 CU들을 포함할 수도 있다. 일부의 경우, 슬라이스는 정수의 트리블록들을 포함한다. 다른 경우, 슬라이스의 경계는 트리블록 내에 있을 수도 있다.

[0046] 픽처에 대해 인코딩 동작을 수행하는 것의 일부로서, 비디오 인코더 (20) 는 그 픽처의 각각의 슬라이스에 대해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 슬라이스에 대해 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 그 슬라이스와 연관되는 인코딩된 데이터를 생성할 수도 있다. 슬라이스와 연관되는 인코딩된 데이터는 "코딩된 슬라이스" 로서 지칭될 수도 있다.

[0047] 코딩된 슬라이스를 생성하기 위해, 비디오 인코더 (20) 는 슬라이스에서의 각각의 트리블록에 관해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 트리블록에 대해 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 코딩된 트리블록을 생성할 수도 있다. 그 코딩된 트리블록은 트리블록의 인코딩된 버전을 나타내는 데이터를 포함할 수도 있다.

[0048] 비디오 인코더 (20) 가 코딩된 슬라이스를 생성할 때, 비디오 인코더 (20) 는 래스터 스캔 순서에 따라서 슬라이스에서의 트리블록들에 관해 인코딩 동작들을 수행할 수도 있다 (예컨대, 인코딩 할 수도 있다). 예를 들어, 비디오 인코더 (20) 는 비디오 인코더 (20) 가 슬라이스에서 트리블록들의 각각을 인코딩 완료할 때까지,

슬라이스에서 트리블록들의 최상부 로우 (row) 를 가로질러 좌에서 우로, 그후 트리블록들의 다음 하위 로우를 가로질러 좌에서 우로, 그리고 기타 등등으로 진행하는 순서로, 슬라이스의 트리블록들을 인코딩할 수도 있다.

[0049] 래스터 스캔 순서에 따라서 트리블록들을 인코딩하는 것의 결과로서, 주어진 트리블록의 상부 및 좌측에 있는 트리블록들은 인코딩되었을 수도 있지만, 주어진 트리블록의 하부 및 우측에 있는 트리블록들은 아직 인코딩되지 않았다. 그 결과, 비디오 인코더 (20) 는 주어진 트리블록을 인코딩할 때, 주어진 트리블록의 상부 및 좌측에 있는 트리블록들을 인코딩함으로써 생성되는 정보에 액세스가능할 수도 있다. 그러나, 비디오 인코더 (20) 는 주어진 트리블록을 인코딩할 때, 주어진 트리블록의 하부 및 우측에 있는 트리블록들을 인코딩함으로써 생성되는 정보에 액세스불가능할 수도 있다.

[0050] 코딩된 트리블록을 생성하기 위해, 비디오 인코더 (20) 는 비디오 블록을 계속해서 더 작은 비디오 블록들로 분할하기 위해 트리블록의 비디오 블록에 관해 큐드트리 파티셔닝을 회귀적으로 수행할 수도 있다. 더 작은 비디오 블록들의 각각은 상이한 CU 와 연관될 수도 있다. 예를 들어, 비디오 인코더 (20) 는 트리블록의 비디오 블록을 4개의 동일-사이즈로된 서브-블록들로 파티셔닝하고, 서브-블록들 중 하나 이상을 4개의 동일-사이즈로된 서브-서브-블록들로 파티셔닝하며, 기타등등으로 파티셔닝할 수도 있다. 파티셔닝된 CU 는 그의 비디오 블록이 다른 CU들과 연관되는 비디오 블록들로 파티셔닝되는 CU 일 수도 있다. 비-파티셔닝된 CU 는 비디오 블록이 다른 CU들과 연관되는 비디오 블록들로 파티셔닝되지 않는 CU 일 수도 있다.

[0051] 비트스트림에서의 하나 이상의 선택스 엘리먼트들은 비디오 인코더 (20) 가 트리블록의 비디오 블록을 파티셔닝 할 수도 있는 최대 횟수를 나타낼 수도 있다. CU 의 비디오 블록은 형태가 정사각형일 수도 있다. CU 의 비디오 블록의 사이즈 (예컨대, CU 의 사이즈) 는 8x8 픽셀들로부터 최고 64x64 픽셀들 이상의 최대치를 갖는 트리블록의 비디오 블록의 사이즈 (예컨대, 트리블록의 사이즈) 까지의 범위에 이를 수도 있다.

[0052] 비디오 인코더 (20) 는 z-스캔 순서에 따라서 트리블록의 각각의 CU 에 관해 인코딩 동작들을 수행할 수도 있다 (예컨대, 인코딩할 수도 있다). 즉, 비디오 인코더 (20) 는 최상부-좌측 CU, 최상부-우측 CU, 최하부-좌측 CU, 그리고 그후 최하부-우측 CU 를, 그 순서로 인코딩할 수도 있다. 비디오 인코더 (20) 가 파티셔닝된 CU 에 관해 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 z-스캔 순서에 따라서, 파티셔닝된 CU 의 비디오 블록의 서브-블록들과 연관되는 CU들을 인코딩 할 수도 있다. 다시 말해서, 비디오 인코더 (20) 는 좌상단 서브-블록과 연관되는 CU, 우상단 서브-블록과 연관되는 CU, 좌하단 서브-블록과 연관되는 CU, 그리고, 그후 우하단 서브-블록과 연관되는 CU 를, 그 순서로 인코딩할 수도 있다.

[0053] z-스캔 순서에 따라서 트리블록의 CU들을 인코딩하는 것의 결과로서, 주어진 CU 의 상부, 좌상부, 우상부, 좌측, 및 좌하부에 있는 CU들이 인코딩되었을 수도 있다. 주어진 CU 의 하부 및 우측에 있는 CU들은 아직 인코딩되지 않았다. 그 결과, 비디오 인코더 (20) 는 주어진 CU 를 인코딩할 때 주어진 CU 에 이웃하는 일부 CU들을 인코딩함으로써 생성되는 정보에 액세스가능할 수도 있다. 그러나, 비디오 인코더 (20) 는 주어진 CU 를 인코딩할 때 주어진 CU 에 이웃하는 다른 CU들을 인코딩함으로써 생성되는 정보에 액세스불가능할 수도 있다.

[0054] 비디오 인코더 (20) 가 비-파티셔닝된 CU 를 인코딩할 때, 비디오 인코더 (20) 는 그 CU 에 대한 하나 이상의 예측 유닛들 (PU들) 을 생성할 수도 있다. 그 CU 의 PU들의 각각은 CU 의 비디오 블록 내에서 상이한 비디오 블록과 연관될 수도 있다. 비디오 인코더 (20) 는 CU 의 각각의 PU 에 대한 예측된 비디오 블록을 생성 할 수도 있다. PU 의 예측된 비디오 블록은 샘플들의 블록일 수도 있다. 비디오 인코더 (20) 는 인트라 예측 또는 인터 예측을 이용하여, PU 에 대한 예측된 비디오 블록을 생성할 수도 있다.

[0055] 비디오 인코더 (20) 가 PU 의 예측된 비디오 블록을 생성하기 위해 인트라 예측을 이용할 때, 비디오 인코더 (20) 는 PU 와 연관되는 픽처의 디코딩된 샘플들에 기초하여 PU 의 예측된 비디오 블록을 생성할 수도 있다. 비디오 인코더 (20) 가 CU 의 PU들의 예측된 비디오 블록들을 생성하기 위해 인트라 예측을 이용하면, 그 CU 는 인트라-예측된 CU 이다. 비디오 인코더 (20) 가 PU 의 예측된 비디오 블록을 생성하기 위해 인터 예측을 이용할 때, 비디오 인코더 (20) 는 PU 와 연관되는 픽처 이외의 하나 이상의 픽처들의 디코딩된 샘플들에 기초하여, PU 의 예측된 비디오 블록을 생성할 수도 있다. 비디오 인코더 (20) 가 CU 의 PU들의 예측된 비디오 블록들을 생성하기 위해 인터 예측을 이용하면, 그 CU 는 인터-예측된 CU 이다.

[0056] 더욱이, 비디오 인코더 (20) 가 PU 에 대한 예측된 비디오 블록을 생성하기 위해 인터 예측을 이용할 때, 비디오 인코더 (20) 는 PU 에 대한 모션 정보를 생성할 수도 있다. PU 에 대한 모션 정보는 PU 의 하나 이상의 참조 블록들을 나타낼 수도 있다. PU 의 각각의 참조 블록은 참조 픽처 내 비디오 블록일 수도 있다.

참조 픽처는 PU 와 연관되는 픽처와는 다른 픽처일 수도 있다. 일부의 경우, PU 의 참조 블록은 또한 PU 의 "참조 샘플" 로서 지정될 수도 있다. 비디오 인코더 (20) 는 PU 의 참조 블록들에 기초하여, PU 에 대한 예측된 비디오 블록을 생성할 수도 있다.

[0057] 비디오 인코더 (20) 가 CU 의 하나 이상의 PU들에 대한 예측된 비디오 블록들을 생성한 후, 비디오 인코더 (20) 는 CU 의 PU들에 대한 예측된 비디오 블록들에 기초하여, CU 에 대한 잔차 데이터를 생성할 수도 있다. CU 에 대한 잔차 데이터는 CU 의 PU들에 대한 예측된 비디오 블록들 및 CU 의 원래 비디오 블록에서의 샘플들 사이의 차이들을 나타낼 수도 있다.

[0058] 더욱이, 비-파티셔닝된 CU 에 관해 인코딩 동작을 수행하는 것의 일부로서, 비디오 인코더 (20) 는 CU 의 잔차 데이터에 관해 회귀적인 퀘드트리 파티셔닝을 수행하여, CU 의 잔차 데이터를 CU 의 변환 유닛들 (TU들) 과 연관된 잔차 데이터의 하나 이상의 블록들 (예컨대, 잔차 비디오 블록들) 로 파티셔닝할 수도 있다. CU 의 각각의 TU 는 상이한 잔차 비디오 블록과 연관될 수도 있다.

[0059] 비디오 인코더 (20) 는 하나 이상의 변환들을 TU들과 연관되는 잔차 비디오 블록들에 적용하여, TU들과 연관되는 변환 계수 블록들 (예컨대, 변환 계수들의 블록들) 을 생성할 수도 있다. 개념적으로, 변환 계수 블록은 변환 계수들의 2차원 (2D) 매트릭스일 수도 있다.

[0060] 변환 계수 블록을 생성한 후, 비디오 인코더 (20) 는 변환 계수 블록에 관해 양자화 프로세스를 수행할 수도 있다. 양자화는 일반적으로 변환 계수들이 변환 계수들을 나타내는데 사용되는 데이터의 양을 가능한 한 감소시키도록 양자화되어 추가적인 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스는 그 변환 계수들의 일부 또는 모두와 연관되는 비트 심도를 감소시킬 수도 있다. 예를 들어, n -비트 변환 계수는 양자화 동안 m -비트 변환 계수로 절사될 수도 있으며, 여기서, n 은 m 보다 더 크다.

[0061] 비디오 인코더 (20) 는 각각의 CU 를 양자화 파라미터 (QP) 값과 연관시킬 수도 있다. CU 와 연관되는 QP 값은 비디오 인코더 (20) 가 CU 와 연관되는 변환 계수 블록들을 양자화하는 방법을 결정할 수도 있다. 비디오 인코더 (20) 는 CU 와 연관되는 QP 값을 조정함으로써, CU 와 연관되는 변환 계수 블록들에 적용되는 양자화의 정도를 조정할 수도 있다.

[0062] 비디오 인코더 (20) 가 변환 계수 블록을 양자화한 후, 비디오 인코더 (20) 는 양자화된 변환 계수 블록에서 변환 계수들을 나타내는 신택스 엘리먼트들의 세트들을 생성할 수도 있다. 비디오 인코더 (20) 는 컨텍스트 적응 2진 산술 코딩 (CABAC) 동작들과 같은, 엔트로피 인코딩 동작들을 이들 신택스 엘리먼트들 중 일부에 적용할 수도 있다. 콘텐츠 적응 가변 길이 코딩 (CAVLC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 2진 산술 코딩과 같은 다른 엔트로피 코딩 기법들이 또한 사용될 수 있다.

[0063] 비디오 인코더 (20) 에 의해 생성된 비트스트림은 일련의 네트워크 추상화 계층 (NAL) 유닛들을 포함할 수도 있다. NAL 유닛들의 각각은 그 NAL 유닛에서의 데이터의 형태의 표시 및 그 데이터를 포함하는 바이트들을 포함하는 신택스 구조일 수도 있다. 예를 들어, NAL 유닛은 비디오 파라미터 세트, 시퀀스 파라미터 세트, 픽처 파라미터 세트, 코딩된 슬라이스, 보충 강화 정보 (SEI), 액세스 유닛 구분문자, 필러 데이터, 또는 또 다른 유형의 데이터를 나타내는 데이터를 포함할 수도 있다. NAL 유닛에서의 데이터는 여러 신택스 구조들을 포함할 수도 있다.

[0064] 비디오 디코더 (30) 는 비디오 인코더 (20) 에 의해 생성된 비트스트림을 수신할 수도 있다. 비트스트림은 비디오 인코더 (20) 에 의해 인코딩된 비디오 데이터의 코딩된 표현을 포함할 수도 있다. 비디오 디코더 (30) 가 비트스트림을 수신할 때, 비디오 디코더 (30) 는 비트스트림에 관해 파싱 동작을 수행할 수도 있다. 비디오 디코더 (30) 가 파싱 동작을 수행할 때, 비디오 디코더 (30) 는 비트스트림으로부터 신택스 엘리먼트들을 추출할 수도 있다. 비디오 디코더 (30) 는 비트스트림으로부터 추출된 신택스 엘리먼트들에 기초하여, 비디오 데이터의 픽처들을 복원할 수도 있다. 신택스 엘리먼트들에 기초하여 비디오 데이터를 복원하는 프로세스는 일반적으로 신택스 엘리먼트들을 생성하기 위해 비디오 인코더 (20) 에 의해 수행되는 프로세스와 반대일 수도 있다.

[0065] 비디오 디코더 (30) 가 CU 와 연관되는 신택스 엘리먼트들을 추출한 후, 비디오 디코더 (30) 는 신택스 엘리먼트들에 기초하여 CU 의 PU들에 대한 예측된 비디오 블록들을 생성할 수도 있다. 게다가, 비디오 디코더 (30) 는 CU 의 TU들과 연관되는 변환 계수 블록들을 역양자화할 수도 있다. 비디오 디코더 (30) 는 변환 계수 블록들에 관해 역변환들을 수행하여, CU 의 TU들과 연관되는 잔차 비디오 블록들을 복원할 수도 있다. 예측된 비디오 블록들을 생성하고 잔차 비디오 블록들을 복원한 후, 비디오 디코더 (30) 는 예측된 비디오 블록

들 및 잔차 비디오 블록들에 기초하여, CU 의 비디오 블록을 복원할 수도 있다. 이러한 방법으로, 비디오 디코더 (30) 는 비트스트림에서의 신택스 엘리먼트들에 기초하여, CU들의 비디오 블록들을 복원할 수도 있다.

[0066] 비디오 인코더

도 2a 는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 비디오 인코더의 예를 예시하는 블록도이다. 비디오 인코더 (20) 는 예컨대, HEVC 에 대한 비디오 프레임의 단일 계층을 프로세싱하도록 구성될 수도 있다. 게다가, 비디오 인코더 (20) 는 본 개시물의 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다. 일 예로서, 예측 프로세싱 유닛 (100) 은 본 개시물에서 설명되는 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다. 다른 실시형태에서, 비디오 인코더 (20) 는 본 개시물에서 설명되는 기법들 중 임의의 기법 또는 모두를 수행하도록 구성되는 옵션적인 인터-계층 예측 유닛 (128) 을 포함한다. 다른 실시형태들에서, 인터-계층 예측은 예측 프로세싱 유닛 (100) (예컨대, 인터 예측 유닛 (121) 및/또는 인트라 예측 유닛 (126)) 에 의해 수행될 수 있으며, 이 경우, 인터-계층 예측 유닛 (128) 은 생략될 수도 있다. 그러나, 본 개시물의 양태들은 이에 한정되지 않는다. 일부 예들에서, 본 개시물에서 설명하는 기법들은 비디오 인코더 (20) 의 여러 구성요소들 사이에 공유될 수도 있다. 일부 예들에서, 추가적으로 또는 대안적으로, 프로세서 (미도시) 는 본 개시물에서 설명되는 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다.

[0068] 설명의 목적을 위해, 본 개시물은 HEVC 코딩의 맥락에서 비디오 인코더 (20) 를 기술한다. 그러나, 본 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다. 도 2a 에 도시된 예는 단일 계층 코덱에 대한 것이다. 그러나, 도 2b 에 대해 추가로 설명되는 바와 같이, 비디오 인코더 (20) 중 일부 또는 모두는 멀티-계층 코덱의 프로세싱을 위해 중복될 수도 있다.

[0069] 비디오 인코더 (20) 는 비디오 슬라이스들 내 비디오 블록들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 또는 픽처 내 비디오에서 공간 리던던시를 감소시키거나 또는 제거하기 위해, 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 인접 프레임들 또는 픽처들 내 비디오에서 시간 리던던시를 감소시키거나 또는 제거하기 위해, 시간 예측에 의존한다. 인트라-모드 (I 모드) 는 여러 공간 기반의 코딩 모드들 중 임의의 코딩 모드를 참조할 수도 있다. 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 과 같은 인터-모드들은 여러 시간-기반의 코딩 모드들 중 임의의 모드를 참조할 수도 있다.

[0070] 도 2a 의 예에서, 비디오 인코더 (20) 는 복수의 기능적 구성요소들을 포함한다. 비디오 인코더 (20) 의 기능적 구성요소들은 예측 프로세싱 유닛 (100), 잔차 생성 유닛 (102), 변환 프로세싱 유닛 (104), 양자화 유닛 (106), 역양자화 유닛 (108), 역변환 유닛 (110), 복원 유닛 (112), 필터 유닛 (113), 디코딩된 픽처 버퍼 (114), 및 엔트로피 인코딩 유닛 (116) 을 포함한다. 예측 프로세싱 유닛 (100) 은 인터 예측 유닛 (121), 모션 추정 유닛 (122), 모션 보상 유닛 (124), 인트라 예측 유닛 (126), 및 인터-계층 예측 유닛 (128) 을 포함한다. 다른 예들에서, 비디오 인코더 (20) 는 더 많거나, 더 적거나, 또는 상이한 기능적 구성요소들을 포함할 수도 있다. 더욱이, 모션 추정 유닛 (122) 및 모션 보상 유닛 (124) 은 고도로 통합될 수도 있지만, 도 2a 의 예에서는 설명의 목적들을 위해 별개로 나타낸다.

[0071] 비디오 인코더 (20) 는 비디오 데이터를 수신할 수도 있다. 비디오 인코더 (20) 는 여러 소스들로부터 비디오 데이터를 수신할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 (예컨대, 도 1a 또는 도 1b 에 도시된) 비디오 소스 (18) 또는 또 다른 소스로부터 비디오 데이터를 수신할 수도 있다. 비디오 데이터는 일련의 픽처들을 나타낼 수도 있다. 비디오 데이터를 인코딩하기 위해, 비디오 인코더 (20) 는 픽처들의 각각에 관해 인코딩 동작을 수행할 수도 있다. 픽처에 관해 인코딩 동작을 수행하는 것의 일부로서, 비디오 인코더 (20) 는 그 픽처의 각각의 슬라이스에 관해 인코딩 동작들을 수행할 수도 있다. 슬라이스에 대해 인코딩 동작을 수행하는 것의 일부로서, 비디오 인코더 (20) 는 그 슬라이스에서의 트리블록들에 관해 인코딩 동작들을 수행할 수도 있다.

[0072] 트리블록에 대해 인코딩 동작을 수행하는 것의 일부로서, 예측 프로세싱 유닛 (100) 은 비디오 블록을 계속해서 더 작은 비디오 블록들로 분할하기 위해 트리블록의 비디오 블록에 관해 퀘드트리 파티셔닝을 수행할 수도 있다. 더 작은 비디오 블록들의 각각은 상이한 CU 와 연관될 수도 있다. 예를 들어, 예측 프로세싱 유닛 (100) 은 트리블록의 비디오 블록을 4개의 동일-사이즈로된 서브-블록들로 파티셔닝하고, 서브-블록들 중 하나 이상을 4개의 동일-사이즈로된 서브-서브-블록들로 파티셔닝하며, 기타등등으로 파티셔닝할 수도 있다.

- [0073] CU들과 연관되는 비디오 블록들의 사이즈는 8x8 샘플들로부터 64x64 샘플들 이상의 최대치를 갖는 트리블록의 사이즈까지의 범위에 이를 수도 있다. 본 개시물에서, "NxN" 및 "N 곱하기 N"은 수직 및 수평 치수들의 관점에서 비디오 블록의 샘플 치수들, 예컨대, 16x16 샘플들 또는 16 곱하기 16 샘플들을 지칭하기 위해 상호교환 가능하게 사용될 수도 있다. 일반적으로, 16x16 비디오 블록은 수직 방향으로 16개의 샘플들 ($y = 16$) 및 수평 방향으로 16개의 샘플들 ($x = 16$)을 갖는다. 이와 유사하게, NxN 블록은 일반적으로 수직 방향으로 N 개의 샘플들 및 수평 방향으로 N 개의 샘플들을 가지며, 여기서 N은 음이 아닌 정수 값을 나타낸다.
- [0074] 더욱이, 트리블록에 대해 인코딩 동작을 수행하는 것의 일부로서, 예측 프로세싱 유닛 (100)은 그 트리블록에 대한 계층적 퀘드트리 데이터 구조를 생성할 수도 있다. 예를 들어, 트리블록은 퀘드트리 데이터 구조의 루트 노드에 대응할 수도 있다. 예측 프로세싱 유닛 (100)이 트리블록의 비디오 블록을 4개의 서브-블록들로 파티셔닝하면, 그 루트 노드는 퀘드트리 데이터 구조에서 4개의 자식 노드들을 갖는다. 자식 노드들 각각은 서브-블록들 중 하나와 연관되는 CU에 대응한다. 예측 프로세싱 유닛 (100)이 서브-블록들 중 하나를 4개의 서브-서브-블록들로 파티셔닝하면, 서브-블록들과 연관되는 CU에 대응하는 노드는 4개의 자식 노드들을 가질 수도 있으며, 그 자식 노드들 각각은 서브-서브-블록들 중 하나와 연관되는 CU에 대응한다.
- [0075] 퀘드트리 데이터 구조의 각각의 노드는 대응하는 트리블록 또는 CU에 대한 선택스 데이터 (예컨대, 선택스 엘리먼트들)를 포함할 수도 있다. 예를 들어, 퀘드트리에서의 노드는 노드에 대응하는 CU의 비디오 블록이 4개의 서브-블록들로 파티셔닝되는지 (예컨대, 분할되는지) 여부를 나타내는 분할 플래그를 포함할 수도 있다. CU에 대한 선택스 엘리먼트들은 회귀적으로 정의될 수도 있으며, CU의 비디오 블록이 서브-블록들로 분할되는지에 의존할 수도 있다. 비디오 블록이 파티셔닝되지 않는 CU는 퀘드트리 데이터 구조에서 리프 노드에 대응할 수도 있다. 코딩된 트리블록은 대응하는 트리블록에 대한 퀘드트리 데이터 구조에 기초한 데이터를 포함할 수도 있다.
- [0076] 비디오 인코더 (20)는 트리블록의 각각의 비-파티셔닝된 CU에 대해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20)가 비-파티셔닝된 CU에 대해 인코딩 동작을 수행할 때, 비디오 인코더 (20)는 비-파티셔닝된 CU의 인코딩된 표현을 나타내는 데이터를 생성한다.
- [0077] CU에 대해 인코딩 동작을 수행하는 것의 일부로서, 예측 프로세싱 유닛 (100)은 CU의 하나 이상의 PU들 사이에 CU의 비디오 블록을 파티셔닝할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30)는 여러 PU 사이즈들을 지원할 수도 있다. 특정의 CU의 사이즈가 $2Nx2N$ 이라고 가정하면, 비디오 인코더 (20) 및 비디오 디코더 (30)는 $2Nx2N$ 또는 NxN 의 PU 사이즈들, 및 $2Nx2N$, $2NxN$, $Nx2N$, NxN , $2NxN$, $nLx2N$, $nRx2N$, 또는 유사한 것의 대칭적인 PU 사이즈들에서의 인터-예측을 지원할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30)는 또한 $2NxN$, $2NxN$, $nLx2N$, 및 $nRx2N$ 의 PU 사이즈들에 대한 비대칭적인 파티셔닝을 지원할 수도 있다. 일부 예들에서, 예측 프로세싱 유닛 (100)은 기하학적 파티셔닝을 수행하여, CU의 비디오 블록을, CU의 PU들 사이에, CU의 비디오 블록의 측면들과 직각으로 만나지 않는 경계를 따라서 파티셔닝할 수도 있다.
- [0078] 인터 예측 유닛 (121)은 CU의 각각의 PU에 대해 인터 예측을 수행할 수도 있다. 인터 예측은 시간 압축을 제공할 수도 있다. PU에 대해 인터 예측을 수행하기 위해, 모션 추정 유닛 (122)은 PU에 대한 모션 정보를 생성할 수도 있다. 모션 보상 유닛 (124)은 CU와 연관되는 픽처 (예컨대, 참조 픽처들) 이외의 픽처들의 모션 정보 및 디코딩된 샘플들에 기초하여, PU에 대한 예측된 비디오 블록을 생성할 수도 있다. 본 개시물에서, 모션 보상 유닛 (124)에 의해 생성되는 예측된 비디오 블록은 인터-예측된 비디오 블록으로서 지칭될 수도 있다.
- [0079] 슬라이스들은 I 슬라이스들, P 슬라이스들, 또는 B 슬라이스들일 수도 있다. 모션 추정 유닛 (122) 및 모션 보상 유닛 (124)은 PU가 I 슬라이스, P 슬라이스, 또는 B 슬라이스 내에 있는지에 따라서, CU의 PU에 대해 상이한 동작들을 수행할 수도 있다. I 슬라이스에서, 모든 PU들은 인트라 예측된다. 그러므로, PU가 I 슬라이스에 있으면, 모션 추정 유닛 (122) 및 모션 보상 유닛 (124)은 PU에 대해 인터 예측을 수행하지 않는다.
- [0080] PU가 P 슬라이스에 있으면, PU를 포함하는 픽처는 "리스트 0"으로서 지칭되는 참조 픽처들의 리스트와 연관된다. 리스트 0에서의 참조 픽처들의 각각은 다른 픽처들의 인터 예측에 사용될 수도 있는 샘플들을 포함한다. 모션 추정 유닛 (122)이 P 슬라이스에서의 PU에 대해 모션 추정 동작을 수행할 때, 모션 추정 유닛 (122)은 PU에 대한 참조 블록에 대해 리스트 0에서의 참조 픽처들을 탐색할 수도 있다. PU의 참조 블록은 PU의 비디오 블록에서의 샘플들에 가장 가깝게 대응하는 샘플들의 세트, 예컨대, 샘플들의 블록일 수도 있다. 모션 추정 유닛 (122)은 다양한 메트릭들을 이용하여, 참조 픽처에서의 샘플들의 세트가 PU의 비디오

블록에서의 샘플들에 얼마나 가깝게 대응하는지를 결정할 수도 있다. 예를 들어, 모션 추정 유닛 (122)은 SAD (sum of absolute difference), SSD (sum of square difference), 또는 다른 차이 메트릭들에 의해 참조 픽처에서의 샘플들의 세트가 PU의 비디오 블록에서의 샘플들에 얼마나 가깝게 대응하는지를 결정할 수도 있다.

[0081] P 슬라이스에서 PU의 참조 블록을 식별한 후, 모션 추정 유닛 (122)은 그 참조 블록을 포함하는 리스트 0에서의 참조 픽처를 나타내는 참조 인덱스 및 PU와 참조 블록 사이의 공간 변위를 나타내는 모션 벡터를 생성할 수도 있다. 여러 예들에서, 모션 추정 유닛 (122)은 다양한 정도의 정밀도로 모션 벡터들을 생성할 수도 있다. 예를 들어, 모션 추정 유닛 (122)은 모션 벡터들을 1/4 샘플 정밀도, 1/8 샘플 정밀도, 또는 다른 분수 샘플 정밀도로 생성할 수도 있다. 분수 샘플 정밀도의 경우, 참조 블록 값들은 참조 픽처에서의 정수-위치 샘플 값들로부터 내삽될 수도 있다. 모션 추정 유닛 (122)은 참조 인덱스 및 모션 벡터를 PU의 모션 정보로서 출력할 수도 있다. 모션 보상 유닛 (124)은 PU의 모션 정보에 의해 식별되는 참조 블록에 기초하여, PU의 예측된 비디오 블록을 생성할 수도 있다.

[0082] PU가 B 슬라이스에 있으면, PU를 포함하는 픽처는 "리스트 0" 및 "리스트 1"로 지칭되는, 참조 픽처들의 2개의 리스트들과 연관될 수도 있다. 일부 예들에서, B 슬라이스를 포함하는 픽처는 리스트 0과 리스트 1의 조합인 리스트 조합과 연관될 수도 있다.

[0083] 더욱이, PU가 B 슬라이스에 있으면, 모션 추정 유닛 (122)은 PU에 대한 단방향 예측 또는 양방향 예측을 수행할 수도 있다. 모션 추정 유닛 (122)이 PU에 대해 단방향 예측을 수행할 때, 모션 추정 유닛 (122)은 PU에 대한 참조 블록에 대해 리스트 0 또는 리스트 1의 참조 픽처들을 탐색할 수도 있다. 모션 추정 유닛 (122)은 그후 참조 블록을 포함하는 리스트 0 또는 리스트 1에서의 참조 픽처를 나타내는 참조 인덱스 및 PU와 참조 블록 사이의 공간 변위를 나타내는 모션 벡터를 생성할 수도 있다. 모션 추정 유닛 (122)은 참조 인덱스, 예측 방향 표시자, 및 모션 벡터를 PU의 모션 정보로서 출력할 수도 있다. 예측 방향 표시자는 참조 인덱스가 리스트 0 또는 리스트 1에서 참조 픽처를 나타내는지를 표시할 수도 있다. 모션 보상 유닛 (124)은 PU의 모션 정보에 의해 표시되는 참조 블록에 기초하여, PU의 예측된 비디오 블록을 생성할 수도 있다.

[0084] 모션 추정 유닛 (122)이 PU에 대해 양방향 예측을 수행할 때, 모션 추정 유닛 (122)은 PU에 대한 참조 블록에 대해서 리스트 0에서의 참조 픽처들을 탐색할 수도 있으며, 또한 PU에 대한 또 다른 참조 블록에 대해서 리스트 1에서의 참조 픽처들을 탐색할 수도 있다. 모션 추정 유닛 (122)은 그후 참조 블록들을 포함하는 리스트 0 및 리스트 1에서 참조 픽처들을 나타내는 참조 인덱스들 및 참조 블록들과 PU 사이의 공간 변위들을 나타내는 모션 벡터들을 생성할 수도 있다. 모션 추정 유닛 (122)은 PU의 참조 인덱스들 및 모션 벡터들을 PU의 모션 정보로서 출력할 수도 있다. 모션 보상 유닛 (124)은 PU의 모션 정보에 의해 표시되는 참조 블록들에 기초하여, PU의 예측된 비디오 블록을 생성할 수도 있다.

[0085] 일부의 경우, 모션 추정 유닛 (122)은 PU에 대한 모션 정보의 전체 세트를 엔트로피 인코딩 유닛 (116)으로 출력하지 않는다. 대신, 모션 추정 유닛 (122)은 또 다른 PU의 모션 정보를 참조하여 PU의 모션 정보를 시그널링할 수도 있다. 예를 들어, 모션 추정 유닛 (122)은 PU의 모션 정보가 이웃하는 PU의 모션 정보와 충분히 유사하다고 결정할 수도 있다. 이 예에서, 모션 추정 유닛 (122)은 PU가 이웃하는 PU와 동일한 모션 정보를 갖는, 비디오 디코더 (30)에 표시되는, 값을 PU와 연관되는 선택스 구조에서, 표시할 수도 있다. 또 다른 예에서, 모션 추정 유닛 (122)은 PU와 연관되는 선택스 구조에서, 이웃하는 PU 및 모션 벡터 차이 (MVD)를 식별할 수도 있다. 모션 벡터 차이는 PU의 모션 벡터와 표시된 이웃하는 PU의 모션 벡터 사이의 차이를 표시한다. 비디오 디코더 (30)는 표시된 이웃하는 PU의 모션 벡터 및 모션 벡터 차이를 이용하여, PU의 모션 벡터를 결정할 수도 있다. 제 2 PU의 모션 정보를 시그널링할 때 제 1 PU의 모션 정보를 참조함으로써, 비디오 인코더 (20)는 더 적은 비트들을 이용하여 제 2 PU의 모션 정보를 시그널링 가능할 수도 있다.

[0086] 도 5 및 도 6을 참조하여 아래서 추가로 논의되는 바와 같이, 예측 프로세싱 유닛 (100)은 도 5 및 도 6에 예시된 방법들을 수행함으로써 PU(또는, 임의의 다른 참조 계층 및/또는 향상 계층 블록들 또는 비디오 유닛들)를 코딩(예컨대, 인코딩 또는 디코딩)하도록 구성될 수도 있다. 예를 들어, (예컨대, 모션 추정 유닛 (122) 및/또는 모션 보상 유닛 (124)을 경유한) 인터 예측 유닛 (121), 인트라 예측 유닛 (126), 또는 인터-계층 예측 유닛 (128)은 도 5 및 도 6에 예시된 방법들을, 함께 또는 별개로 수행하도록 구성될 수도 있다.

[0087] CU에 대해 인코딩 동작을 수행하는 것의 일부로서, 인트라 예측 유닛 (126)은 CU의 PU들에 관해 인트라 예측

을 수행할 수도 있다. 인트라 예측은 공간 압축을 제공할 수도 있다. 인트라 예측 유닛 (126) 이 PU에 관해 인트라 예측을 수행할 때, 인트라 예측 유닛 (126) 은 동일한 픽처에서의 다른 PU들의 디코딩된 샘플들에 기초하여 PU에 대한 예측 데이터를 생성할 수도 있다. PU에 대한 예측 데이터는 예측된 비디오 블록 및 여러 신택스 엘리먼트들을 포함할 수도 있다. 인트라 예측 유닛 (126) 은 I 슬라이스들, P 슬라이스들, 및 B 슬라이스들에서의 PU들에 대해 인트라 예측을 수행할 수도 있다.

[0088]

PU에 대해 인트라 예측을 수행하기 위해, 인트라 예측 유닛 (126) 은 다수의 인트라 예측 모드들을 이용하여 PU에 대한 예측 데이터의 다수의 세트들을 생성할 수도 있다. 인트라 예측 유닛 (126) 이 인트라 예측 모드를 이용하여 PU에 대한 예측 데이터의 세트를 생성할 때, 인트라 예측 유닛 (126) 은 샘플들을 이웃하는 PU들의 비디오 블록들로부터, PU의 비디오 블록을 가로질러, 인트라 예측 모드와 연관되는 방향 및/또는 기울기로 확장할 수도 있다. PU들, CU들, 및 트리블록들에 대해 좌-우, 상-하 인코딩 순서를 가정하면, 이웃하는 PU들은 PU의 상부에, 우상부에, 좌상부에, 또는 좌측에 있을 수도 있다. 인트라 예측 유닛 (126) 은 PU의 사이즈에 따라서, 다수의 인트라 예측 모드들, 예컨대, 33개의 방향 인트라 예측 모드들을 이용할 수도 있다.

[0089]

예측 프로세싱 유닛 (100) 은 PU에 대해 모션 보상 유닛 (124)에 의해 생성된 예측 데이터 또는 PU에 대해 인트라 예측 유닛 (126)에 의해 생성된 예측 데이터 중에서, PU에 대한 예측 데이터를 선택할 수도 있다.

일부 예들에서, 예측 프로세싱 유닛 (100) 은 예측 데이터의 세트들의 레이트/왜곡 메트릭들에 기초하여 PU에 대한 예측 데이터를 선택한다.

[0090]

예측 프로세싱 유닛 (100) 이 인트라 예측 유닛 (126)에 의해 생성된 예측 데이터를 선택하면, 예측 프로세싱 유닛 (100) 은 PU들에 대한 예측 데이터를 생성하는데 사용되었던 인트라 예측 모드, 예컨대, 선택된 인트라 예측 모드를 시그널링할 수도 있다. 예측 프로세싱 유닛 (100) 은 그 선택된 인트라 예측 모드를 여러 방법들로 시그널링할 수도 있다. 예를 들어, 선택된 인트라 예측 모드가 이웃하는 PU의 인트라 예측 모드와 동일할 가능성이 있을 수도 있다. 다시 말해서, 이웃하는 PU의 인트라 예측 모드는 현재의 PU에 대해 가장 가능성 있는 모드일 수도 있다. 따라서, 예측 프로세싱 유닛 (100) 은 선택된 인트라 예측 모드가 이웃하는 PU의 인트라 예측 모드와 동일하다는 것을 표시하기 위해 신택스 엘리먼트를 생성할 수도 있다.

[0091]

위에서 논의한 바와 같이, 비디오 인코더 (20) 는 인터-계층 예측 유닛 (128) 을 포함할 수도 있다. 인터-계층 예측 유닛 (128) 은 SVC에 이용가능한 하나 이상의 상이한 계층들 (예컨대, 기초 또는 참조 계층) 을 이용하여 현재의 블록 (예컨대, EL에서의 현재의 블록) 을 예측하도록 구성된다. 이러한 예측은 인터-계층 예측으로서 지칭될 수도 있다. 인터-계층 예측 유닛 (128) 은 예측 방법들을 이용하여 인터-계층 리던시를 감소시켜, 코딩 효율을 향상시키고 컴퓨터 리소스 요구사항들을 감소시킨다. 인터-계층 예측의 일부 예들은 인터-계층 인트라 예측, 인터-계층 모션 예측, 및 인터-계층 잔차 예측을 포함한다. 인터-계층 인트라 예측은 기초 계층에서 동일 위치에 배치된 블록들의 복원을 이용하여 항상 계층에서의 현재의 블록을 예측한다. 인터-계층 모션 예측은 기초 계층의 모션 정보를 이용하여 항상 계층에서의 모션을 예측한다. 인터-계층 잔차 예측은 기초 계층의 잔차를 이용하여 항상 계층의 잔차를 예측한다. 인터-계층 예측 방식들의 각각이 더욱 자세하게 아래에서 논의된다.

[0092]

예측 프로세싱 유닛 (100) 이 CU의 PU들에 대한 예측 데이터를 선택한 후, 잔차 생성 유닛 (102) 은 CU의 비디오 블록으로부터 CU의 PU들의 예측된 비디오 블록들을 감산함으로써 (예컨대, 마이너스 부호로 표시됨) CU에 대한 잔차 데이터를 생성할 수도 있다. CU의 잔차 데이터는 CU의 비디오 블록에서의 샘플들의 상이한 샘플 구성요소들에 대응하는 2D 잔차 비디오 블록들을 포함할 수도 있다. 예를 들어, 잔차 데이터는 CU의 PU들의 예측된 비디오 블록들에서의 샘플들의 희도 성분들과, CU의 원래 비디오 블록에서의 샘플들의 희도 성분들 사이의 차이들에 대응하는 잔차 비디오 블록을 포함할 수도 있다. 게다가, CU의 잔차 데이터는 CU의 PU들의 예측된 비디오 블록들에서의 샘플들의 색차 성분들과, CU의 원래 비디오 블록에서의 샘플들의 색차 성분들 사이의 차이들에 대응하는 잔차 비디오 블록들을 포함할 수도 있다.

[0093]

예측 프로세싱 유닛 (100) 은 쿼드트리 파티셔닝을 수행하여, CU의 잔차 비디오 블록들을 서브-블록들로 파티셔닝할 수도 있다. 각각의 미분할된 잔차 비디오 블록은 CU의 상이한 TU와 연관될 수도 있다. CU의 TU들과 연관되는 잔차 비디오 블록들의 사이즈들 및 위치들은 CU의 PU들과 연관되는 비디오 블록들의 사이즈들 및 위치들에 기초하거나 또는 기초하지 않을 수도 있다. "잔차 쿼드 트리" (RQT)로서 알려진 쿼드트리 구조는 잔차 비디오 블록들의 각각과 연관되는 노드들을 포함할 수도 있다. CU의 TU들은 RQT의 리프 노드들에 대응할 수도 있다.

[0094]

변환 프로세싱 유닛 (104) 은 하나 이상의 변환들을 TU와 연관되는 잔차 비디오 블록에 적용함으로써 CU의 각

각의 TU에 대한 하나 이상의 변환 계수 블록들을 생성할 수도 있다. 변환 계수 블록들의 각각은 변환 계수들의 2D 매트릭스일 수도 있다. 변환 프로세싱 유닛 (104)은 여러 변환들을 TU와 연관되는 잔차 비디오 블록에 적용할 수도 있다. 예를 들어, 변환 프로세싱 유닛 (104)은 이산 코사인 변환 (DCT), 방향 변환, 또는 개념적으로 유사한 변환을 TU와 연관되는 잔차 비디오 블록에 적용할 수도 있다.

[0095] 변환 프로세싱 유닛 (104)이 TU와 연관되는 변환 계수 블록을 생성한 후, 양자화 유닛 (106)은 변환 계수 블록에서의 변환 계수들을 양자화할 수도 있다. 양자화 유닛 (106)은 CU와 연관되는 QP값에 기초하여, CU의 TU와 연관되는 변환 계수 블록을 양자화할 수도 있다.

[0096] 비디오 인코더 (20)는 QP값을 CU와 여러 방법들로 연관시킬 수도 있다. 예를 들어, 비디오 인코더 (20)는 CU와 연관되는 트리블록에 관해 레이트-왜곡 분석을 수행할 수도 있다. 레이트-왜곡 분석에서, 비디오 인코더 (20)는 트리블록에 관해 인코딩 동작을 여러번 수행함으로써, 트리블록의 다수의 코딩된 표현들을 생성할 수도 있다. 비디오 인코더 (20)는 비디오 인코더 (20)가 트리블록의 상이한 인코딩된 표현들을 생성할 때, 상이한 QP값들을 CU와 연관시킬 수도 있다. 비디오 인코더 (20)는 주어진 QP값이 최저 비트레이트 및 왜곡 메트릭을 갖는 트리블록의 코딩된 표현에서 CU와 연관될 때 주어진 QP값이 CU와 연관된다고 시그널링할 수도 있다.

[0097] 역양자화 유닛 (108) 및 역변환 유닛 (110)은 역양자화 및 역변환들을 변환 계수 블록에 각각 적용하여, 변환 계수 블록으로부터 잔차 비디오 블록을 복원할 수도 있다. 복원 유닛 (112)은 복원된 잔차 비디오 블록을 예측 프로세싱 유닛 (100)에 의해 생성되는 하나 이상의 예측된 비디오 블록들로부터의 대응하는 샘플들에 가산하여, TU와 연관되는 복원된 비디오 블록을 생성할 수도 있다. CU의 각각의 TU에 대한 비디오 블록들을 이 방법으로 복원함으로써, 비디오 인코더 (20)는 CU의 비디오 블록을 복원할 수도 있다.

[0098] 복원 유닛 (112)이 CU의 비디오 블록을 복원한 후, 필터 유닛 (113)은 CU와 연관되는 비디오 블록에서 블록킹 아티팩트들을 감소시키기 위해 디블록킹 동작을 수행할 수도 있다. 하나 이상의 디블록킹 동작들을 수행한 후, 필터 유닛 (113)은 CU의 복원된 비디오 블록을 디코딩된 픽처 베퍼 (114)에 저장할 수도 있다. 모션 추정 유닛 (122) 및 모션 보상 유닛 (124)은 복원된 비디오 블록을 포함하는 참조 픽처를 이용하여, 후속 픽처들의 PU들에 관해 인터 예측을 수행할 수도 있다. 게다가, 인트라 예측 유닛 (126)은 디코딩된 픽처 베퍼 (114)에서의 복원된 비디오 블록들을 이용하여, CU와 동일한 픽처에서의 다른 PU들에 관해 인트라 예측을 수행할 수도 있다.

[0099] 엔트로피 인코딩 유닛 (116)은 비디오 인코더 (20)의 다른 기능적 구성요소들로부터 데이터를 수신할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (116)은 양자화 유닛 (106)으로부터 변환 계수 블록들을 수신할 수도 있으며 예측 프로세싱 유닛 (100)으로부터 신팩스 엘리먼트들을 수신할 수도 있다. 엔트로피 인코딩 유닛 (116)이 데이터를 수신할 때, 엔트로피 인코딩 유닛 (116)은 하나 이상의 엔트로피 인코딩 동작들을 수행하여, 엔트로피 인코딩된 데이터를 생성할 수도 있다. 예를 들어, 비디오 인코더 (20)는 그 데이터에 관해, 컨텍스트 적용 가변 길이 코딩 (CAVLC) 동작, CABAC 동작, 변수-대-변수 (V2V) 길이 코딩 동작, 신팩스-기반 컨텍스트-적용 2진 산술 코딩 (SBAC) 동작, 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩 동작, 또는 또 다른 유형의 엔트로피 인코딩 동작을 수행할 수도 있다. 엔트로피 인코딩 유닛 (116)은 엔트로피 인코딩된 데이터를 포함하는 비트스트림을 출력할 수도 있다.

[0100] 데이터에 대해 엔트로피 인코딩 동작을 수행하는 것의 일부로서, 엔트로피 인코딩 유닛 (116)은 컨텍스트 모델을 선택할 수도 있다. 엔트로피 인코딩 유닛 (116)이 CABAC 동작을 수행하고 있으면, 컨텍스트 모델은 특정의 값을 가질 특정의 빈들의 확률들의 추정치들을 나타낼 수도 있다. CABAC의 맥락에서, 용어 "빈"은 신팩스 엘리먼트의 2진화된 버전의 비트를 지칭하기 위해 사용된다.

멀티-계층 비디오 인코더

[0102] 도 2b는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 멀티-계층 비디오 인코더 (23)의 예를 예시하는 블록도이다. 비디오 인코더 (23)는 예컨대, SHVC 및 멀티뷰 코딩에 있어 멀티-계층 비디오 프레임들을 프로세싱하도록 구성될 수도 있다. 또, 비디오 인코더 (23)는 본 개시물의 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다.

[0103] 비디오 인코더 (23)는 비디오 인코더 (20A) 및 비디오 인코더 (20B)를 포함하며, 그 각각은 비디오 인코더 (20)로서 구성될 수도 있으며 비디오 인코더 (20)에 관하여 위에서 설명된 기능들을 수행할 수도 있다.

또, 참조 번호들의 재사용에 의해 표시되는 바와 같이, 비디오 인코더들 (20A 및 20B) 은 시스템들 및 서브시스템들 중 적어도 일부를 비디오 인코더 (20) 로서 포함할 수도 있다. 비디오 인코더 (23) 가 2개의 비디오 인코더들 (20A 및 20B) 을 포함하는 것으로 예시되지만, 비디오 인코더 (23) 는 그에 한정되지 않으며 임의 개수의 비디오 인코더 (20) 계층들을 포함할 수도 있다. 일부 실시형태들에서, 비디오 인코더 (23) 는 액세스 유닛에서의 각각의 픽처 또는 프레임에 대한 비디오 인코더 (20) 를 포함할 수도 있다. 예를 들어, 5개의 픽처들을 포함하는 액세스 유닛이 5개의 인코더 계층들을 포함하는 비디오 인코더에 의해 프로세싱되거나 또는 인코딩될 수도 있다. 일부 실시형태들에서, 비디오 인코더 (23) 는 액세스 유닛에서의 프레임들보다 더 많은 인코더 계층들을 포함할 수도 있다. 이러한 일부 경우들에서, 비디오 인코더 계층들 중 일부는 어떤 액세스 유닛들을 프로세싱할 때 비활성일 수도 있다.

[0104] 비디오 인코더들 (20A 및 20B) 에 추가하여, 비디오 인코더 (23) 는 리샘플링 유닛 (90) 을 포함할 수도 있다. 리샘플링 유닛 (90) 은 일부의 경우, 수신된 비디오 프레임의 기초 계층을 업샘플링하여, 예를 들어, 향상 계층을 생성할 수도 있다. 리샘플링 유닛 (90) 은 프레임의 수신된 기초 계층과 연관된 특정의 정보를 업샘플링하지만, 다른 정보를 업샘플링하지 않을 수도 있다. 예를 들어, 리샘플링 유닛 (90) 은 기초 계층의 공간 사이즈 또는 픽셀들의 개수를 업샘플링할 수도 있지만, 슬라이스들의 개수 또는 픽처 순서 카운트는 일정하게 유지할 수도 있다. 일부의 경우, 리샘플링 유닛 (90) 은 수신된 비디오를 프로세싱하지 않을 수도 있으며 및/또는 옵션적일 수도 있다. 예를 들어, 일부의 경우, 예측 프로세싱 유닛 (100) 은 업샘플링을 수행할 수도 있다. 일부 실시형태들에서, 리샘플링 유닛 (90) 은 계층을 업샘플링하고 슬라이스 경계 규칙들 및/또는 래스터 스캔 규칙들의 세트에 따르도록 하나 이상의 슬라이스들을 재편성하거나, 재정의하거나, 수정하거나 조정하도록 구성된다. 액세스 유닛에서 기초 계층, 또는 하위 계층을 업샘플링하는 것으로 주로 설명되지만, 일부의 경우, 리샘플링 유닛 (90) 은 계층을 다운샘플링할 수도 있다. 예를 들어, 비디오의 스트리밍 동안 대역폭이 감소되면, 프레임은 업샘플링 대신 다운샘플링될 수도 있다.

[0105] 리샘플링 유닛 (90) 은 하위 계층 인코더 (예컨대, 비디오 인코더 (20A)) 의 디코딩된 픽처 버퍼 (114) 로부터 픽처 또는 프레임 (또는, 픽처와 연관된 픽처 정보) 을 수신하고 픽처 (또는, 수신된 픽처 정보) 를 업샘플링하도록 구성될 수도 있다. 이 업샘플링된 픽처는 그후 하위 계층 인코더와 동일한 액세스 유닛에서의 픽처를 인코딩하도록 구성된 상위 계층 인코더 (예컨대, 비디오 인코더 (20B)) 의 예측 프로세싱 유닛 (100) 에 제공될 수도 있다. 일부의 경우, 상위 계층 인코더는 하위 계층 인코더로부터 한 계층 제거된다. 다른 경우, 도 2b 의 계층 0 비디오 인코더와 계층 1 인코더 사이에 하나 이상의 상위 계층 인코더들이 존재할 수도 있다.

[0106] 일부의 경우, 리샘플링 유닛 (90) 은 생략되거나 또는 바이패스될 수도 있다. 이러한 경우, 비디오 인코더 (20A) 의 디코딩된 픽처 버퍼 (114) 로부터의 픽처는 직접, 또는 적어도 리샘플링 유닛 (90) 에 제공됨이 없이, 비디오 인코더 (20B) 의 예측 프로세싱 유닛 (100) 에 제공될 수도 있다. 예를 들어, 비디오 인코더 (20B) 에 제공된 비디오 데이터 및 비디오 인코더 (20A) 의 디코딩된 픽처 버퍼 (114) 로부터의 참조 픽처가 동일한 사이즈 또는 해상도이면, 참조 픽처는 비디오 인코더 (20B) 에 임의의 리샘플링 없이 제공될 수도 있다.

[0107] 일부 실시형태들에서, 비디오 인코더 (23) 는 비디오 데이터를 비디오 인코더 (20A) 에 제공되기 전에 다운샘플링 유닛 (94) 을 이용하여 하위 계층 인코더에 제공될 비디오 데이터를 다운샘플링한다. 대안적으로, 다운샘플링 유닛 (94) 은 비디오 데이터를 업샘플링하거나 또는 다운샘플링하는 것이 가능한 리샘플링 유닛 (90) 일 수도 있다. 또한 다른 실시형태들에서, 다운샘플링 유닛 (94) 은 생략될 수도 있다.

[0108] 도 2b 에 예시된 바와 같이, 비디오 인코더 (23) 는 멀티플렉서 (98), 또는 mux 를 더 포함할 수도 있다. mux (98) 는 비디오 인코더 (23) 로부터의 결합된 비트스트림을 출력할 수도 있다. 결합된 비트스트림은 비디오 인코더들 (20A 및 20B) 의 각각으로부터 비트스트림을 취하여 어느 비트스트림이 주어진 시간에 출력되는지를 교번시킴으로써 생성될 수도 있다. 일부 경우들에서 2개의 (또는, 2개보다 많은 비디오 인코더 계층들) 경우, 그 이상의 비트스트림들로부터의 비트들은 한번에 1 비트씩 교번될 수도 있지만, 많은 경우, 비트스트림들은 상이하게 결합된다. 예를 들어, 출력 비트스트림은 선택된 비트스트림을 한번에 한 블록씩 교번시킴으로써 생성될 수도 있다. 다른 예에서, 출력 비트스트림은 비디오 인코더들 (20A 및 20B) 의 각각으로부터 블록들의 비-1:1 비율을 출력함으로써 생성될 수도 있다. 예를 들어, 2개의 블록들이 비디오 인코더 (20A) 로부터 출력된 각각의 블록에 대해 비디오 인코더 (20B) 로부터 출력될 수도 있다. 일부 실시형태들에서, mux (98) 로부터의 출력 스트림은 사전 프로그래밍될 수도 있다. 다른 실시형태들에서, mux (98) 는 비디오 인코더 (23) 외부에 있는 시스템으로부터, 예컨대, 소스 모듈 (12) 을 포함한 소스 디바이스 상의 프로세서로부터 수신된 제어 신호에 기초하여, 비디오 인코더들 (20A, 20B) 로부터의 비트스트림들을 결합할 수도 있다. 제어 신호는 비디오 소스 (18) 로부터의 비디오의 해상도 또는 비트레이트에 기초하여, 링크 (16) 의

대역폭에 기초하여, 사용자와 연관된 가입 (예컨대, 유료 가입 대 무료 가입)에 기초하여, 또는 비디오 인코더 (23)로부터 요망된 해상도 출력을 결정하기 위한 임의의 다른 인자에 기초하여, 생성될 수도 있다.

[0109] 비디오 디코더

도 3a는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록도이다. 비디오 디코더 (30)는 예컨대, HEVC에 있어 비디오 프레임의 단일 계층을 프로세싱하도록 구성될 수도 있다. 게다가, 비디오 디코더 (30)는 본 개시물의 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다. 일 예로서, 모션 보상 유닛 (162) 및/또는 인트라 예측 유닛 (164)은 본 개시물에서 설명되는 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다. 일 실시형태에서, 비디오 디코더 (30)는 옵션적으로, 본 개시물에서 설명되는 기법들 중 임의의 기법 또는 모두를 수행하도록 구성되는 인터-계층 예측 유닛 (166)을 포함할 수도 있다. 다른 실시형태들에서, 인터-계층 예측은 예측 프로세싱 유닛 (152) (예컨대, 모션 보상 유닛 (162) 및/또는 인트라 예측 유닛 (164))에 의해 수행될 수 있으며, 이 경우, 인터-계층 예측 유닛 (166)은 생략될 수도 있다. 그러나, 본 개시물의 양태들은 이에 한정되지 않는다.

일부 예들에서, 본 개시물에서 설명하는 기법들은 비디오 디코더 (30)의 여러 구성요소들 사이에 공유될 수도 있다. 일부 예들에서, 추가적으로 또는 대안적으로, 프로세서 (미도시)는 본 개시물에서 설명되는 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다.

[0111] 설명의 목적을 위해, 본 개시물은 HEVC 코딩의 맥락에서의 비디오 디코더 (30)를 기술한다. 그러나, 본 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다. 도 3a에 도시된 예는 단일 계층 코덱에 대한 것이다. 그러나, 도 3b에 대해 추가로 설명되는 바와 같이, 비디오 디코더 (30) 중 일부 또는 모두는 멀티-계층 코덱의 프로세싱을 위해 중복될 수도 있다.

[0112] 도 3a의 예에서, 비디오 디코더 (30)는 복수의 기능적 구성요소들을 포함한다. 비디오 디코더 (30)의 기능적 구성요소들은 엔트로피 디코딩 유닛 (150), 예측 프로세싱 유닛 (152), 역양자화 유닛 (154), 역변환 유닛 (156), 복원 유닛 (158), 필터 유닛 (159), 및 디코딩된 픽처 버퍼 (160)를 포함한다. 예측 프로세싱 유닛 (152)은 모션 보상 유닛 (162), 인트라 예측 유닛 (164), 및 인터-계층 예측 유닛 (166)을 포함한다. 일부 예들에서, 비디오 디코더 (30)는 도 2a의 비디오 인코더 (20)에 대해 설명된 인코딩 과정과는 일반적으로 반대인 디코딩 과정을 수행할 수도 있다. 다른 예들에서, 비디오 디코더 (30)는 더 많거나, 더 적거나, 또는 상이한 기능적 구성요소들을 포함할 수도 있다.

[0113] 비디오 디코더 (30)는 인코딩된 비디오 데이터를 포함하는 비트스트림을 수신할 수도 있다. 비트스트림은 복수의 신팩스 엘리먼트들을 포함할 수도 있다. 비디오 디코더 (30)가 비트스트림을 수신할 때, 엔트로피 디코딩 유닛 (150)은 비트스트림에 관해 파싱 동작을 수행할 수도 있다. 비트스트림에 관해 파싱 동작을 수행하는 것의 결과로서, 엔트로피 디코딩 유닛 (150)은 비트스트림으로부터 신팩스 엘리먼트들을 추출할 수도 있다. 파싱 동작을 수행하는 것의 일부로서, 엔트로피 디코딩 유닛 (150)은 비트스트림에서의 엔트로피 인코딩된 신팩스 엘리먼트들을 엔트로피 디코딩할 수도 있다. 예측 프로세싱 유닛 (152), 역양자화 유닛 (154), 역변환 유닛 (156), 복원 유닛 (158), 및 필터 유닛 (159)은 비트스트림으로부터 추출된 신팩스 엘리먼트들에 기초하여, 디코딩된 비디오 데이터를 생성하는 복원 동작을 수행할 수도 있다.

[0114] 위에서 논의한 바와 같이, 비트스트림은 일련의 NAL 유닛들을 포함할 수도 있다. 비트스트림의 NAL 유닛들은 비디오 파라미터 세트 NAL 유닛들, 시퀀스 파라미터 세트 NAL 유닛들, 픽처 파라미터 세트 NAL 유닛들, SEI NAL 유닛들 등을 포함할 수도 있다. 비트스트림에 대해 파싱 동작을 수행하는 것의 일부로서, 엔트로피 디코딩 유닛 (150)은 시퀀스 파라미터 세트 NAL 유닛들로부터 시퀀스 파라미터 세트들을, 픽처 파라미터 세트 NAL 유닛들로부터 픽처 파라미터 세트들을, SEI NAL 유닛들로부터 SEI 데이터를, 기타 등을 추출하여 엔트로피 디코딩하는 파싱 동작들을 수행할 수도 있다.

[0115] 게다가, 비트스트림의 NAL 유닛들은 코딩된 슬라이스 NAL 유닛들을 포함할 수도 있다. 비트스트림에 관해 파싱 동작을 수행하는 것의 일부로서, 엔트로피 디코딩 유닛 (150)은 코딩된 슬라이스 NAL 유닛들로부터 코딩된 슬라이스들을 추출하여 엔트로피 디코딩하는 파싱 동작들을 수행할 수도 있다. 코딩된 슬라이스들 각각은 슬라이스 헤더 및 슬라이스 데이터를 포함할 수도 있다. 슬라이스 헤더는 슬라이스에 관련된 신팩스 엘리먼트들을 포함할 수도 있다. 슬라이스 헤더에서의 신팩스 엘리먼트들은 그 슬라이스를 포함하는 픽처와 연관된 픽처 파라미터 세트를 식별하는 신팩스 엘리먼트를 포함할 수도 있다. 엔트로피 디코딩 유닛 (150)

은 슬라이스 헤더를 복원하기 위해, 코딩된 슬라이스 헤더에서의 신팩스 엘리먼트들에 관해 CABAC 디코딩 동작들과 같은, 엔트로피 디코딩 동작들을 수행할 수도 있다.

[0116] 코딩된 슬라이스 NAL 유닛들로부터 슬라이스 데이터를 추출하는 것의 일부로서, 엔트로피 디코딩 유닛 (150)은 슬라이스 데이터에서의 코딩된 CU들로부터 신팩스 엘리먼트들을 추출하는 과정 동작들을 수행할 수도 있다.

추출된 신팩스 엘리먼트들은 변환 계수 블록들과 연관되는 신팩스 엘리먼트들을 포함할 수도 있다. 엔트로피 디코딩 유닛 (150)은 그후 신팩스 엘리먼트들 중 일부에 관해 CABAC 디코딩 동작들을 수행할 수도 있다.

[0117] 엔트로피 디코딩 유닛 (150)이 비-파티셔닝된 CU에 관해 과정 동작을 수행한 후, 비디오 디코더 (30)는 비-파티셔닝된 CU에 관해 복원 동작을 수행할 수도 있다. 비-파티셔닝된 CU에 대해 복원 동작을 수행하기 위해, 비디오 디코더 (30)는 CU의 각각의 TU에 대해 복원 동작을 수행함으로써, 비디오 디코더 (30)는 CU와 연관되는 잔차 비디오 블록을 복원할 수도 있다.

[0118] TU에 관해 복원 동작을 수행하는 것의 일부로서, 역양자화 유닛 (154)은 TU와 연관되는 변환 계수 블록을 역양자화할 수도 있다. 예컨대, 양자화 해제 (de-quantize) 할 수도 있다. 역양자화 유닛 (154)은 변환 계수 블록을 HEVC에 대해 제안되거나 또는 H.264 디코딩 표준에 의해 정의되는 역양자화 프로세스들과 유사한 방법으로 역양자화할 수도 있다. 역양자화 유닛 (154)은 변환 계수 블록의 CU에 대해 비디오 인코더 (20)에 의해 계산된 양자화 파라미터 QP를 이용하여, 양자화의 정도 및, 이와 유사하게, 적용할 역양자화 유닛 (154)에 대한 역양자화의 정도를 결정할 수도 있다.

[0119] 역양자화 유닛 (154)이 변환 계수 블록을 역양자화한 후, 역변환 유닛 (156)은 변환 계수 블록과 연관되는 TU에 대한 잔차 비디오 블록을 생성할 수도 있다. 역변환 유닛 (156)은 TU에 대한 잔차 비디오 블록을 생성하기 위해 변환 계수 블록에 역변환을 적용할 수도 있다. 예를 들어, 역변환 유닛 (156)은 역 DCT, 역 정수 변환, 역 Karhunen-Loeve 변환 (KLT), 역 회전 변환, 역 방향 변환, 또는 또 다른 역변환을 변환 계수 블록에 적용할 수도 있다. 일부 예들에서, 역변환 유닛 (156)은 비디오 인코더 (20)로부터의 시그널링에 기초하여, 변환 계수 블록에 적용할 역변환을 결정할 수도 있다. 이러한 예들에서, 역변환 유닛 (156)은 변환 계수 블록과 연관되는 트리블록에 대한 쿼드트리의 루트 노드에서의 시그널링된 변환에 기초하여, 역변환을 결정할 수도 있다. 다른 예들에서, 역변환 유닛 (156)은 하나 이상의 코딩 특성들, 예컨대 블록 사이즈, 코딩 모드 등으로부터 역변환을 추론할 수도 있다. 일부 예들에서, 역변환 유닛 (156)은 캐스케이드된 역변환을 적용할 수도 있다.

[0120] 일부 예들에서, 모션 보상 유닛 (162)은 내삽 필터들에 기초하여 내삽을 수행함으로써, PU의 예측된 비디오 블록을 개선할 수도 있다. 서브-샘플 정밀도로 모션 보상에 사용되는 내삽 필터들에 대한 식별자들이 신팩스 엘리먼트들에 포함될 수도 있다. 모션 보상 유닛 (162)은 PU의 예측된 비디오 블록의 생성 동안 비디오 인코더 (20)에 의해 사용되는 동일한 내삽 필터들을 이용하여 참조 블록의 서브-정수 샘플들에 대해 내삽된 값들을 계산할 수도 있다. 모션 보상 유닛 (162)은 수신된 신팩스 정보에 따라 비디오 인코더 (20)에 의해 사용되는 내삽 필터들을 결정하고 그 내삽 필터들을 이용하여 예측된 비디오 블록을 생성할 수도 있다.

[0121] 도 5 및 도 6을 참조하여 아래서 추가로 논의되는 바와 같이, 예측 프로세싱 유닛 (152)은 도 5 및 도 6에 예시된 방법들을 수행함으로써 PU(또는, 임의의 다른 참조 계층 및/또는 향상 계층 블록들 또는 비디오 유닛들)를 코딩(예컨대, 인코딩 또는 디코딩)할 수도 있다. 예를 들어, 모션 보상 유닛 (162), 인트라 예측 유닛 (164), 또는 인터-계층 예측 유닛 (166)은 도 5 및 도 6에 예시된 방법들을, 함께 또는 별개로 수행하도록 구성될 수도 있다.

[0122] PU가 인트라 예측을 이용하여 인코딩되면, 인트라 예측 유닛 (164)은 인트라 예측을 수행하여, PU에 대한 예측된 비디오 블록을 생성할 수도 있다. 예를 들어, 인트라 예측 유닛 (164)은 비트스트림에서의 신팩스 엘리먼트들에 기초하여, 그 PU에 대한 인트라 예측 모드를 결정할 수도 있다. 비트스트림은 인트라 예측 유닛 (164)이 PU의 인트라 예측 모드를 결정하는데 사용할 수도 있는 신팩스 엘리먼트들을 포함할 수도 있다.

[0123] 일부의 경우, 신팩스 엘리먼트들은 인트라 예측 유닛 (164)이 현재의 PU의 인트라 예측 모드를 결정하기 위해 또 다른 PU의 인트라 예측 모드를 이용해야 한다고 표시할 수도 있다. 예를 들어, 현재의 PU의 인트라 예측 모드가 이웃하는 PU의 인트라 예측 모드와 동일할 가능성성이 있을 수도 있다. 다시 말해서, 이웃하는 PU의 인트라 예측 모드는 현재의 PU에 대해 가장 가능성 있는 모드일 수도 있다. 그러므로, 이 예에서, 비트스트림은 PU의 인트라 예측 모드가 이웃하는 PU의 인트라 예측 모드와 동일하다는 것을 나타내는 작은 신팩스 엘리먼트를 포함할 수도 있다. 인트라 예측 유닛 (164)은 그후 공간적으로 이웃하는 PU들의 비디오 블록들

에 기초하여, PU에 대한 예측 데이터 (예컨대, 예측된 샘플들)을 생성하기 위해, 인트라 예측 모드를 이용할 수도 있다.

[0124] 위에서 논의한 바와 같이, 비디오 디코더 (30)는 또한 인터-계층 예측 유닛 (166)을 포함할 수도 있다. 인터-계층 예측 유닛 (166)은 SVC에 이용가능한 하나 이상의 상이한 계층들 (예컨대, 기초 또는 참조 계층)을 이용하여 현재의 블록 (예컨대, EL에서의 현재의 블록)을 예측하도록 구성된다. 이러한 예측은 인터-계층 예측으로서 지정될 수도 있다. 인터-계층 예측 유닛 (166)은 예측 방법들을 이용하여 인터-계층 리던시를 감소시켜, 코딩 효율을 향상시키고 컴퓨터 리소스 요구사항들을 감소시킨다. 인터-계층 예측의 일부 예들은 인터-계층 인트라 예측, 인터-계층 모션 예측, 및 인터-계층 잔차 예측을 포함한다. 인터-계층 인트라 예측은 기초 계층에서 동일 위치에 배치된 블록들의 복원을 이용하여 항상 계층에서의 현재의 블록을 예측한다. 인터-계층 모션 예측은 기초 계층의 모션 정보를 이용하여 항상 계층에서의 모션을 예측한다. 인터-계층 잔차 예측은 기초 계층의 잔차를 이용하여 항상 계층의 잔차를 예측한다. 인터-계층 예측 방식들의 각각이 더욱더 자세하게 아래에서 논의된다.

[0125] 복원 유닛 (158)은 CU의 TU들과 연관되는 잔차 비디오 블록들 및 CU의 PU들의 예측된 비디오 블록들, 예컨대, 적용가능한 경우, 인트라-예측 데이터 또는 인터-예측 데이터 중 어느 하나를 이용하여, CU의 비디오 블록을 복원할 수도 있다. 따라서, 비디오 디코더 (30)는 비트스트림에서의 신택스 엘리먼트들에 기초하여, 예측된 비디오 블록 및 잔차 비디오 블록을 생성할 수도 있으며 그 예측된 비디오 블록 및 잔차 비디오 블록에 기초하여 비디오 블록을 생성할 수도 있다.

[0126] 복원 유닛 (158)이 CU의 비디오 블록을 복원한 후, 필터 유닛 (159)은 CU와 연관되는 블록킹 아티팩트들을 감소시키기 위해 디블록킹 동작을 수행할 수도 있다. 필터 유닛 (159)이 CU와 연관되는 블록킹 아티팩트들을 감소시키기 위해 디블록킹 동작을 수행한 후, 비디오 디코더 (30)는 CU의 비디오 블록을 디코딩된 픽처 버퍼 (160)에 저장할 수도 있다. 디코딩된 픽처 버퍼 (160)는 후속 모션 보상, 인트라 예측, 및 도 1a 또는 도 1b의 디스플레이 디바이스 (32)와 같은 디스플레이 디바이스 상에의 프리젠테이션을 위해, 참조 픽처들을 제공할 수도 있다. 예를 들어, 비디오 디코더 (30)는 디코딩된 픽처 버퍼 (160)에서의 비디오 블록들에 기초하여, 다른 CU들의 PU들에 관해 인트라 예측 또는 인터 예측 동작들을 수행할 수도 있다.

멀티-계층 디코더

[0128] 도 3b는 본 개시물에서 설명하는 양태들에 따른 기법들을 구현할 수도 있는 멀티-계층 비디오 디코더 (33)의 예를 예시하는 블록도이다. 비디오 디코더 (33)는 멀티-계층 비디오 프레임들을, 예컨대 SHVC 및 멀티뷰 코딩을 위해 프로세싱하도록 구성될 수도 있다. 게다가, 비디오 디코더 (33)는 본 개시물의 기법들 중 임의의 기법 또는 모두를 수행하도록 구성될 수도 있다.

[0129] 비디오 디코더 (33)는 비디오 디코더 (30A) 및 비디오 디코더 (30B)를 포함하며, 그 각각은 비디오 디코더 (30)로서 구성될 수도 있으며 비디오 디코더 (30)에 대하여 위에서 설명된 기능들을 수행할 수도 있다. 또, 참조 번호들의 재사용에 의해 표시되는 바와 같이, 비디오 디코더들 (30A 및 30B)은 시스템들 및 서브시스템들 중 적어도 일부를 비디오 디코더 (30)로서 포함할 수도 있다. 비디오 디코더 (33)가 2개의 비디오 디코더들 (30A 및 30B)을 포함하는 것으로 예시되지만, 비디오 디코더 (33)는 그에 한정되지 않으며 임의의 개수의 비디오 디코더 (30) 계층들을 포함할 수도 있다. 일부 실시형태들에서, 비디오 디코더 (33)는 액세스 유닛에서의 각각의 픽처 또는 프레임에 대한 비디오 디코더 (30)를 포함할 수도 있다. 예를 들어, 5개의 픽처들을 포함하는 액세스 유닛은 5개의 디코더 계층들을 포함하는 비디오 디코더에 의해 프로세싱되거나 또는 디코딩될 수도 있다. 일부 실시형태들에서, 비디오 디코더 (33)는 액세스 유닛에서의 프레임들보다 더 많은 디코더 계층들을 포함할 수도 있다. 어떤 이러한 경우들에서, 비디오 디코더 계층들 중 일부는 어떤 액세스 유닛들을 프로세싱할 때 비활성일 수도 있다.

[0130] 비디오 디코더들 (30A 및 30B)에 추가하여, 비디오 디코더 (33)는 업샘플링 유닛 (92)을 포함할 수도 있다. 일부 실시형태들에서, 업샘플링 유닛 (92)은 프레임 또는 액세스 유닛에 대한 참조 픽처 리스트에 추가될 향상된 계층을 생성하기 위해 수신된 비디오 프레임의 기초 계층을 업샘플링할 수도 있다. 이 향상된 계층은 디코딩된 픽처 버퍼 (160)에 저장될 수 있다. 일부 실시형태들에서, 업샘플링 유닛 (92)은 도 2a의 리샘플링 유닛 (90)에 대하여 설명된 실시형태들 중 일부 또는 모두를 포함할 수 있다. 일부 실시형태들에서, 업샘플링 유닛 (92)은 계층을 업샘플링하고 슬라이스 경계 규칙들 및/또는 래스터 스캔 규칙들의 세트에

따르도록 하나 이상의 슬라이스들을 재편성하거나, 재정의하거나, 수정하거나 조정하도록 구성된다. 일부의 경우, 업샘플링 유닛 (92)은 수신된 비디오 프레임의 계층을 업샘플링하고 및/또는 다운샘플링하도록 구성된 리샘플링 유닛일 수도 있다.

[0131] 업샘플링 유닛 (92)은 하위 계층 디코더 (예컨대, 비디오 디코더 (30A))의 디코딩된 픽처 버퍼 (160)로부터 픽처 또는 프레임 (또는, 픽처와 연관된 픽처 정보)을 수신하고 픽처 (또는, 수신된 픽처 정보)를 업샘플링하도록 구성될 수도 있다. 이 업샘플링된 픽처는 그후 하위 계층 디코더와 동일한 액세스 유닛에서의 픽처를 디코딩하도록 구성된 상위 계층 디코더 (예컨대, 비디오 디코더 (30B))의 예측 프로세싱 유닛 (152)에 제공될 수도 있다. 일부의 경우, 상위 계층 디코더는 하위 계층 디코더로부터 한 계층 제거된다. 다른 경우, 도 3b의 계층 0 디코더와 계층 1 디코더 사이에 하나 이상의 상위 계층 디코더들이 존재할 수도 있다.

[0132] 일부의 경우, 업샘플링 유닛 (92)은 생략되거나 또는 바이패스될 수도 있다. 이러한 경우, 비디오 디코더 (30A)의 디코딩된 픽처 버퍼 (160)로부터의 픽처는 직접, 또는 적어도 업샘플링 유닛 (92)에 제공됨이 없이, 비디오 디코더 (30B)의 예측 프로세싱 유닛 (152)에 제공될 수도 있다. 예를 들어, 비디오 디코더 (30B)에 제공된 비디오 데이터 및 비디오 디코더 (30A)의 디코딩된 픽처 버퍼 (160)로부터의 참조 픽처가 동일한 사이즈 또는 해상도이면, 참조 픽처는 업샘플링함이 없이 비디오 디코더 (30B)에 제공될 수도 있다. 또, 일부 실시형태들에서, 업샘플링 유닛 (92)은 비디오 디코더 (30A)의 디코딩된 픽처 버퍼 (160)로부터 수신된 참조 픽처를 업샘플링되거나 또는 다운샘플링되도록 구성된 리샘플링 유닛 (90) 일 수도 있다.

[0133] 도 3b에 예시된 바와 같이, 비디오 디코더 (33)는 디멀티플렉서 (99), 또는 demux를 더 포함할 수도 있다. demux (99)는 인코딩된 비디오 비트스트림을 다수의 비트스트림들로 분할할 수 있고, demux (99)에 의해 출력된 각각의 비트스트림은 상이한 비디오 디코더 (30A 및 30B)에 제공된다. 다수의 비트스트림들이 비트스트림을 수신함으로써 생성될 수도 있으며, 비디오 디코더들 (30A 및 30B)의 각각이 비트스트림의 일부를 주어진 시간에 수신한다. 일부 경우들에서 demux (99)에서 수신된 비트스트림으로부터의 비트들은 비디오 디코더들 (예컨대, 도 3b의 예에서 비디오 디코더들 (30A 및 30B))의 각각 사이에 한번에 1비트씩 교변될 수도 있지만, 많은 경우들에서, 비트스트림은 다르게 분할된다. 예를 들어, 비트스트림은 어느 비디오 디코더가 비트스트림을 한번에 하나의 블록씩 수신하는지를 교변시킴으로써 분할될 수도 있다. 다른 예에서, 비트스트림은 블록들 대 비디오 디코더들 (30A 및 30B)의 각각의 비-1:1 비율로 분할될 수도 있다. 예를 들어, 2개의 블록들이 비디오 디코더 (30A)에 제공되는 각각의 블록에 대해 비디오 디코더 (30B)에 제공될 수도 있다. 일부 실시형태들에서, demux (99)에 의한 비트스트림의 분할은 사전 프로그래밍될 수도 있다. 다른 실시형태들에서, demux (99)는 비디오 디코더 (33) 외부에 있는 시스템으로부터, 예컨대 목적지 모듈 (14)을 포함한 목적지 디바이스 상의 프로세서로부터 수신된 제어 신호에 기초하여 비트스트림을 분할할 수도 있다. 제어 신호는 입력 인터페이스 (28)로부터의 비디오의 해상도 또는 비트레이트에 기초하여, 링크 (16)의 대역폭에 기초하여, 사용자와 연관된 가입 (예컨대, 유료 가입 대 무료 가입)에 기초하여, 또는 비디오 디코더 (33)에 의해 획득가능한 해상도를 결정하기 위한 임의의 다른 인자에 기초하여 생성될 수도 있다.

인트라 랜덤 액세스 지점 (IRAP) 픽처들

[0134] 어떤 비디오 코딩 방식들은 비트스트림이 비트스트림에서 그들 랜덤 액세스 지점들에 선행하는 임의의 픽처들을 디코딩할 필요 없이, 그들 랜덤 액세스 지점들 중 임의의 지점에서 시작하여 디코딩될 수도 있도록, 비트스트림 전반에 걸쳐서 여러 랜덤 액세스 지점들을 제공할 수도 있다. 이러한 비디오 코딩 방식들에서, (예컨대, 랜덤 액세스 지점을 제공하는 픽처와 동일한 액세스 유닛에 있는 그들 픽처들을 포함한) 출력 순서에서 랜덤 액세스 지점에 뒤따르는 모든 픽처들이 랜덤 액세스 지점에 선행하는 임의의 픽처들을 이용함이 없이 정확하게 디코딩될 수 있다. 예를 들어, 비록 비트스트림의 일부분이 송신 동안 또는 디코딩 동안 손실되더라도, 디코더는 다음 랜덤 액세스 지점에서 시작하여 비트스트림을 디코딩하는 것을 재개할 수 있다. 랜덤 액세스에 대한 지원은 예를 들어, 동적 스트리밍 서비스들, 탐색 (seek) 동작들, 채널 스위칭, 등을 용이하게 할 수도 있다.

[0135] 일부 코딩 방식들에서, 이러한 랜덤 액세스 지점들은 인트라 랜덤 액세스 지점 (IRAP) 픽처들로서 지정되는 픽처들에 의해 제공될 수도 있다. 예를 들어, 액세스 유닛 ("auA")에 포함된 향상 계층 ("layerA")에서의 (예컨대, 향상 계층 IRAP 픽처에 의해 제공된) 랜덤 액세스 지점은, layerB에 있으며 디코딩 순서에서 auA에 선행하는 액세스 유닛 ("auB")에 포함된 랜덤 액세스 지점 (또는, auA에 포함된 랜덤 액세스 지점)을 가지는 layerA의 각각의 참조 계층 ("layerB") (예컨대, layerA를 예측하는데 사용되는 계층인 참조 계층)에 대해,

(auB 에 로케이트된 그들 픽처들을 포함한) 출력 순서에서 auB 에 뒤따르는 layerA 에서의 픽처들이 auB 에 선 행하는 layerA 에서의 임의의 픽처들을 디코딩할 필요 없이, 정확하게 디코딩 가능하도록, 계층-특정 랜덤 액세스를 제공할 수도 있다.

[0137]

IRAP 픽처들은 인트라 예측을 이용하여 코딩 (예컨대, 다른 픽처들을 참조함이 없이 코딩) 되고/되거나 인터-계 층 예측을 이용하여 코딩될 수도 있으며, 예를 들어, IDR (순시 디코더 리프레시) 픽처들, 클린 랜덤 액세스 (CRA) 픽처들, 및 깨진 링크 액세스 (BLA) 픽처들을 포함할 수도 있다. 비트스트림에 IDR 픽처가 존재할 때, 디코딩 순서에서 IDR 픽처에 선행하는 모든 픽처들은 IDR 픽처에 뒤따르는 픽처들에 의한 예측에 사용되지 않는다. 비트스트림에 CRA 픽처가 존재할 때, CRA 픽처에 뒤따르는 픽처들은 예측을 위해 디코딩 순서에서 CRA 픽처에 선행하는 픽처들을 사용할 수도 또는 하지 않을 수도 있다. 디코딩 순서에서 CRA 픽처에 뒤따르지만 디코딩 순서에서 CRA 픽처에 선행하는 픽처들을 이용하는 그들 픽처들은 랜덤 액세스 스kip된 선두 (RASL) 픽처들로서 지정될 수도 있다. 디코딩 순서에서 IRAP 픽처에 뒤따르고 출력 순서에서 IRAP 픽처에 선행할 수 있는 다른 유형의 픽처는, 디코딩 순서에서 IRAP 픽처에 선행하는 임의의 픽처들에 대한 참조들을 포함하지 않을 수도 있는, 랜덤 액세스 디코딩 가능한 선두 (RADL) 픽처이다. RASL 픽처들은 CRA 픽처에 선행하는 픽처들이 이용불가능한 경우 디코더에 의해 폐기될 수도 있다. BLA 픽처는 (예컨대, 2개의 비트스트림들이 함께 스플라이스되고 BLA 픽처가 디코딩 순서에서 제 2 비트스트림의 제 1 픽처이기 때문에) BLA 픽처에 선행하는 픽처들이 디코더에 이용불가능할 수도 있다고, 디코더에게 표시한다. IRAP 픽처인 (예컨대, 0 의 계층 ID 값을 가지는) 기초 계층 픽처를 포함하는 액세스 유닛 (예컨대, 다수의 계층들에 걸쳐서 동일한 출력 시간과 연관된 모든 코딩된 픽처들로 이루어진 픽처들의 그룹) 은 IRAP 액세스 유닛으로서 지정될 수도 있다.

[0138]

IRAP 픽처들의 교차-계층 정렬

[0139]

SVC 에서, IRAP 픽처들은 상이한 계층들을 가로질러 정렬되도록 (예컨대, 동일한 액세스 유닛에 포함되도록) 요구되지 않을 수도 있다. 예를 들어, IRAP 픽처들이 정렬되도록 요구되었으면, 적어도 하나의 IRAP 픽처를 포함하는 임의의 액세스 유닛은 단지 IRAP 픽처들만을 포함할 것이다. 한편, IRAP 픽처들이 정렬되도록 요구되지 않았으면, 단일 액세스 유닛에서, (예컨대, 제 1 계층에서) 하나의 픽처는 IRAP 픽처일 수도 있으며, (예컨대, 제 2 계층에서) 다른 픽처는 비-IRAP 픽처일 수도 있다. 비트스트림에서 이러한 비-정렬된 IRAP 픽처들을 가지는 것은 어떤 이점들을 제공할 수도 있다. 예를 들어, 2-계층 비트스트림에서, 향상 계층보다 기초 계층에서 더 많은 IRAP 픽처들이 있으면, 브로드캐스트 및 멀티캐스트 애플리케이션들에서, 낮은 투-인 지연 (tune-in delay) 과 동시에 높은 코딩 효율이 달성될 수 있다.

[0140]

일부 비디오 코딩 방식들에서, POC (picture order count) 는 디코딩된 픽처들이 디스플레이되는 상대적인 순서를 추적하는데 사용될 수도 있다. 이러한 코딩 방식들 중 일부는 어떤 유형들의 픽처들이 비트스트림에 나타날 때마다 POC 값들이 (예컨대, 제로로 또는 비트스트림으로 시그널링되는 어떤 값으로) 리셋되도록 할 수도 있다. 예를 들어, 어떤 IRAP 픽처들의 POC 값들이 리셋될 수도 있으며, 이에 의해 디코딩 순서에서 그들 IRAP 픽처들에 선행하는 다른 픽처들의 POC 값들이 또한 리셋될 수도 있다. 이것은 IRAP 픽처들이 상이한 계층들을 가로질러 정렬되도록 요구되지 않는 경우 문제가 될 수도 있다. 예를 들어, 하나의 픽처 ("picA") 가 IRAP 픽처이고 동일한 액세스 유닛에서의 다른 픽처 ("picB") 가 IRAP 픽처가 아닐 때, picA 를 포함하는 계층에서, IRAP 픽처인 picA 로 인해 리셋되는 픽처 ("picC") 의 POC 값은 picB 를 포함하는 계층에서, 리셋되지 않은, 픽처 ("picD") 의 POC 값과 상이할 수도 있으며, 여기서, picC 및 picD 는 동일한 액세스 유닛 내에 있다. 이는 picC 및 picD 가, 설령 그들이 동일한 액세스 유닛 (예컨대, 동일한 출력 시간) 에 속하더라도, 상이한 POC 값들을 갖도록 한다. 따라서, 이 예에서, picC 및 picD 의 POC 값들을 유도하는 유도 프로세스는 POC 값들 및 액세스 유닛들의 정의에 따르는 POC 값들을 생성하도록 수정될 수 있다.

[0141]

계층 초기화 픽처 (LIP)

[0142]

일부 코딩 방식들에서, 계층 초기화 픽처 ("LIP 픽처") 는 1 로 설정된 NoRas1OutputFlag 플래그 (예컨대, RASL 픽처들이 1 로 설정되면 출력되지 않아야 한다는 것을 표시하고 RASL 픽처들이 0 으로 설정되면 출력되어야 한다는 것을 표시하는 플래그) 를 가지는 IRAP 픽처인 픽처 또는 기초 계층 픽처 (예컨대, 0 의 계층 ID 또는 비트스트림에서 정의된 가장 작은 계층 ID 를 가지는 픽처) 가 1 로 설정된 NoRas1OutputFlag 를 가지는 IRAP 액세스 유닛인 초기 IRAP 액세스 유닛에 포함되는 픽처로서 정의될 수도 있다.

[0143]

일부 실시형태들에서, SPS 는 각각의 LIP 팩처에서 활성화될 수 있다. 예를 들어, 1로 설정된 NoRas1OutputFlag 플래그를 가지는 각각의 IRAP 팩처 또는 초기 IRAP 액세스 유닛에 포함되는 각각의 팩처, 이전에 활성화된 SPS 와는 상이할 수도 있는 (예컨대, 상이한 팩처 해상도들, 등을 규정하는) 새로운 SPS. 그러나, LIP 팩처가 IRAP 팩처 (예컨대, 초기 IRAP 액세스 유닛에 포함된 임의의 팩처) 가 아니고, 그리고 초기 IRAP 액세스 유닛에서의 기초 계층 팩처가 0으로 설정된 플래그 NoClrasOutputFlag 플래그 (예컨대, 교차-계층 랜덤 액세스 스킵 팩처들이 1로 설정되면 출력되지 않아야 한다는 것을 표시하고 교차-계층 랜덤 액세스 스킵 팩처들이 0으로 설정되면 출력되어야 한다는 것을 표시하는 플래그) 를 가지는 IDR 팩처인 경우, LIP 팩처는 새로운 SPS 를 활성화하도록 허용되지 않아야 한다. 이러한 경우, 특히 새로운 SPS 의 SPS RBSP 의 콘텐츠가 초기 IRAP 액세스 유닛 이전에 활성화되었던 SPS 의 콘텐츠와 상이할 때 새로운 SPS 가 이러한 LIP 팩처에서 활성화되면, 상이한 팩처 해상도들 및 여러 복원력에서 문제들이 있을 수 있다. 예를 들어, 새로운 SPS 는 해상도를 업데이트하고, 시간 예측을 이용하여 상이한 사이즈들의 팩처들을 참조할 수도 있다.

[0144]

팩처들의 범평 및 플러싱

[0145]

(예컨대, 팩처들이 다른 팩처들을 예측하기 위해 디스플레이되거나 또는 사용될 수도 있도록) 디코딩되는 팩처들은 디코딩된 팩처 버퍼 (DPB) 에 저장된다. 출력되어야 하는 팩처들은 "출력용으로 요구됨" 으로서 마킹될 수도 있으며, 다른 팩처들을 예측하는데 사용되어야 하는 팩처들은 "참조용으로 사용됨" 으로서 마킹될 수도 있다. "출력용으로 요구됨" 으로도 "참조용으로 사용됨" 으로도 마킹되지 않는 디코딩된 팩처들 (예컨대, 처음에 "참조용으로 사용됨" 또는 "출력용으로 요구됨" 으로서 마킹되었지만 그 후에 "참조용으로 미사용됨" 또는 "출력용으로 미요구됨" 으로서 마킹된 팩처들) 이 그들이 디코딩 프로세스에 의해 제거될 때까지 DPB 에 존재할 수도 있다. 출력 순서 순응 디코더들에서, DPB 로부터 팩처들을 제거하는 프로세스는 종종 "출력용으로 요구됨" 으로서 마킹되는 팩처들의 출력에 바로 뒤따른다. 이 출력 및 후속 제거의 프로세스는 "범평" 으로서 지칭될 수도 있다.

[0146]

또한, 설령 이들 팩처들이 "출력용으로 요구됨" 으로서 마킹될 수도 있더라도, 디코더가 출력 없이 DPB 에서의 팩처들을 제거할 수도 있는 상황들이 존재한다. 본원에서 설명의 용이성을 위해, IRAP 팩처를 디코딩할 때에 DPB 에 존재하는 디코딩된 팩처들은 (디코딩된 팩처들이 "출력용으로 요구됨" 또는 "참조용으로 사용됨" 으로서 마킹되는지 여부에 관계 없이) IRAP 팩처와 연관된 "래깅 DPB 팩처들" 또는 IRAP 팩처의 "연관된 래깅 DPB 팩처들" 로서 지칭된다. HEVC 맥락에서, 이러한 상황들의 일부 예들이 아래에 설명된다.

[0147]

일 예에서, "1"의 값과 동일한 NoRas1OutputFlag 를 가지는 CRA 팩처가 (예컨대, 비트스트림에서 제 1 팩처가 아닌) 비트스트림의 중간에 존재할 때, CRA 팩처와 연관된 래깅 DPB 팩처들은 출력되지 않을 것이며 DPB 로부터 제거될 것이다. 이러한 상황들은 스플라이스 지점들에서 발생할 가능성이 있으며, 여기서, 2개의 비트스트림들이 함께 결합되며 후자의 비트스트림의 제 1 팩처는 "1"의 값과 동일한 NoRas1OutputFlag 를 가지는 CRA 팩처이다. 다른 예에서, "1"의 값과 동일한 NoRas1OutputFlag 를 가지고 CRA 팩처가 아닌 (예컨대, IDR 팩처인) IRAP 팩처 picA 가 비트스트림의 중간에 존재하고 팩처의 해상도가 picA 에서 (예컨대, 새로운 SPS 의 활성화에 의해) 변할 때, 연관된 래깅 DPB 팩처들이 DPB 를 계속 점유하면, picA 에서 시작하는 팩처들의 디코딩이 예를 들어, 버퍼 오버플로우로 인해, 문제가 될 수도 있기 때문에, picA 의 연관된 래깅 DPB 팩처들은, 그들이 출력될 수 있기 전에, DPB 로부터 제거될 수도 있다. 이 경우, DPB 에서의 출력 없이 래깅 팩처들을 플러싱하기 위해, picA 와 연관된 no_output_of_prior_pics_flag (예컨대, 이전에 디코딩되어 DPB 에 저장된 팩처들이 1로 설정되면 출력됨이 없이 DPB 로부터 제거되어야 한다는 것을 표시하고, 그리고 이전에 디코딩되어 DPB 에 저장된 팩처들이 0으로 설정되면 출력됨이 없이 DPB 로부터 제거되지 않아야 한다는 것을 표시하는 플래그) 의 값은 인코더 또는 스플라이서에 의해 "1"의 값과 동일하게 설정되어야 하거나, 또는 NoOutputOfPriorPicsFlag (예컨대, 비트스트림에 포함된 정보에 기초하여 결정될 수도 있는 유도된 값) 는 디코더에 의해 "1"의 값과 동일하게 유도될 수도 있다. 스플라이싱 동작은 도 4에 대하여 아래에서 추가로 설명된다.

[0148]

출력 없이 DPB 로부터 연관된 래깅 DPB 팩처들을 제거하는 이 프로세스는 "플러싱" 으로서 지칭될 수도 있다.

심지어 위에서 설명되지 않은 상황들에서도, IRAP 팩처는 디코더가 IRAP 팩처의 연관된 DPB 래깅 팩처들을 플러싱하도록, "1"의 값과 동일한 no_output_of_prior_pics_flag 의 값을 규정할 수도 있다.

[0149] 스플라이스 지점을 포함하는 비트스트림

도 4 를 참조하여, 스플라이스 지점을 가지는 예시적인 비트스트림이 설명될 것이다. 도 4 는 비트스트림들 (410 및 420) 을 스플라이싱함으로써 생성된 멀티-계층 비트스트림 (400) 을 나타낸다. 비트스트림 (410) 은 향상 계층 (EL) (410A) 및 기초 계층 (BL) (410B) 을 포함하며, 비트스트림 (420) 은 EL (420A) 및 BL (420B) 을 포함한다. EL (410A) 은 EL 팩처 (412A) 를 포함하며, BL (410B) 은 BL 팩처 (412B) 를 포함한다. EL (420A) 은 EL 팩처들 (422A, 424A, 및 426A) 을 포함하며, BL (420B) 은 BL 팩처들 (422B, 424B, 및 426B) 을 포함한다. 멀티-계층 비트스트림 (400) 은 액세스 유닛들 (AU들) (430 내지 460) 을 더 포함한다. AU (430) 는 EL 팩처 (412A) 및 BL 팩처 (412B) 를 포함하며, AU (440) 는 EL 팩처 (422A) 및 BL 팩처 (422B) 를 포함하며, AU (450) 는 EL 팩처 (424A) 및 BL 팩처 (424B) 를 포함하며, AU (460) 는 EL 팩처 (426A) 및 BL 팩처 (426B) 를 포함한다. 도 4 의 예에서, BL 팩처 (422B) 는 IRAP 팩처이고, AU (440) 에서의 대응하는 EL 팩처 (422A) 는 트레일링 팩처 (예컨대, 비-IRAP 팩처) 이며, 결과적으로, AU (440) 는 비-정렬된 IRAP AU 이다. 또한, AU (440) 가 스플라이스 지점 (470) 에 바로 뒤따르는 액세스 유닛인 점에 유의해야 한다.

도 4 의 예가 2개의 상이한 비트스트림들이 함께 결합되는 경우를 예시하지만, 일부 실시형태들에서, 스플라이스 지점은 비트스트림의 일부가 제거될 때 존재할 수도 있다. 예를 들어, 비트스트림은 부분들 A, B, 및 C 를 가질 수도 있으며, 부분 B 는 부분 A 와 부분 C 사이에 있다. 부분 B 가 비트스트림으로부터 제거되면, 나머지 부분들 A 및 C 는 함께 결합될 수도 있으며, 그들이 함께 결합되는 지점은 스플라이스 지점으로서 지정될 수도 있다. 더 일반적으로는, 본 출원에서 논의되는 바와 같은 스플라이스 지점은 하나 이상의 시그널링 된 또는 유도된 파라미터들 또는 플래그들이 미리 결정된 값들을 가질 때 존재하는 것으로 간주될 수도 있다. 예를 들어, 스플라이스 지점이 특정의 로케이션에 존재한다는 특정의 표시를 수신하지 않고, 디코더는 플래그 (예컨대, NoClrasOutputFlag) 의 값을 결정하고, 그 플래그의 값에 기초하여 본 출원에서 설명되는 하나 이상의 기법들을 수행할 수도 있다.

[0152] 멀티-계층 맵에서의 팩처들의 플러싱

팩처들을 플러싱하는 프로세스는 또한 멀티-계층 비트스트림들에 관련된다. 좀더 구체적으로, 그 프로세스는 초기 IRAP 액세스 유닛에 속하는 모든 팩처들에, 또한 초기 IRAP 액세스 유닛에 있지 않는 IRAP 팩처들에 관련된다. 위에서 설명된 바와 같이, SHVC 및 MV-HEVC 와 같은 일부 기존 구현예들에서, IRAP 액세스 유닛은 (액세스 유닛에서의 다른 팩처들이 IRAP 팩처들인지 여부에 관계 없이) "0" 의 값과 동일한 nuh_layer_id 를 가지는 IRAP 팩처를 포함하는 액세스 유닛으로서 정의될 수도 있으며, 초기 IRAP 액세스 유닛은 (또한, 액세스 유닛에서의 다른 팩처들이 IRAP 팩처들인지 여부에 관계 없이) "0" 의 값과 동일한 nuh_layer_id 를 가지고 그리고 "1" 의 값과 동일한 NoRas1OutputFlag 를 가지는 IRAP 팩처를 포함하는 액세스 유닛으로서 정의될 수도 있다.

SHVC 및 MV-HEVC 에서 액세스 유닛들에서 비-정렬된 IRAP 팩처들을 가질 가능성에 의해 (예컨대, 액세스 유닛은 IRAP 팩처들 및 비-IRAP 팩처들 양자를 포함할 수도 있다), HEVC 의 맵에서 이전 섹션에서 설명된 상황들이 SHVC/MV-HEVC 비트스트림의 상이한 계층들에서 발생할 수 있다. 예를 들어, "1" 의 값과 동일한 NoRas1OutputFlag 를 가지는 CRA 팩처 picA 는 picA 와 동일한 계층에서의 CRA 팩처를 가지지 않는 초기 IRAP 액세스 유닛에서 시작하는 (예컨대, 비트스트림의 제 1 액세스 유닛이 아닌) 비트스트림의 중간의 향상 계층에 존재할 수도 있다. 또한, 팩처의 해상도 변화가 기초 계층의 해상도가 변하지 않는 액세스 유닛에서 향상 계층에서의 IRAP 팩처들에서 일어날 수 있거나, 또는 반대의 경우도 마찬가지이다. 유사한 상황들이 상이한 DPB 사이즈들에 대해 발생할 수도 있다.

[0155] SVC 및 MVC 에서의 팩처들의 플러싱

SVC 의 단일-루프 코딩 설계로 인해, 액세스 유닛 당 오직 하나의 복원된 팩처가, 소위 중간-그래뉼라 스케일러 빌리티 (MGS) 가 사용 중일 때의 경우들 (이 경우 DPB 에 저장되는 소위 주요-팩처 액세스 유닛들로부터의 2개의 디코딩된 팩처들이 있을 수 있다) 을 제외하고는, DPB 에 삽입된다. 그러나, 각각의 액세스 유닛에서, 오직 최고 계층의 디코딩된 팩처만이 출력될 수도 있다. 따라서, 팩처들의 플러싱을 포함한, DPB 를 관리하는 동작들은, 대개, 기초 계층의 디코딩된 팩처가 향상 계층을 예측하기 위해 DPB 에 존재하도록 요구되지 않기

때문에, 단지 최고 계층에서의 픽처들에만 관계한다.

[0157] MVC에서, 하나 보다 많은 뷰가 목표 출력 뷰일 수도 있으며, 디코딩된 뷰 성분들은, 그들이 동일한 계층에서 뷰 성분들을 예측하는데 필요로 하지 않더라도, 다른 계층에서의 뷰 성분들을 예측하기 위해 유지될 필요가 있다. 따라서, 하나 보다 많은 뷰로부터의 뷰 성분들이 DPB에 존재할 수도 있다. 플래그 no_output_of_prior_pics_flag는 각각의 IDR 뷰 성분에 대해 시그널링되며 (예컨대, 비-베이스 뷰의 IDR 뷰 성분이 "0"의 값과 동일한 non_idr_flag로 시그널링되며), 뷰 성분들의 플러싱은 계층-특정적 (또는, 뷰-특정적)이다. MVC에서는, 간결성을 위해, MVC에서 액세스 유닛에서의 IDR 뷰 성분들은 정렬된다. 예를 들어, 액세스 유닛에서의 하나의 뷰 성분이 IDR 뷰 성분이면, 그 액세스 유닛에서의 모든 뷰 성분들은 또한 IDR 뷰 성분들이다. 그러므로, 플러싱 동작은, 또한 그 동작이 뷰/계층-특정적일 수 있더라도, 비트스트림에서 모든 뷰들에 걸쳐서 수행된다.

SHVC 및 MV-HEVC에서의 픽처들의 플러싱

[0159] 플러싱이 SHVC 및 MV-HEVC에서의 현재의 설계 하에서 발생할 때, DPB에서의 모든 픽처들은 출력됨이 (예컨대, 디스플레이됨이) 없이 제거된다. (오직 기초 계층만이 비트스트림에 존재하는 사소한 경우를 제외하고는) 비트스트림에서의 오직 하나의 계층만의 픽처들이 플러싱되는 것이 불가능하며, 따라서 플러싱은 계층-특정적이다.

출력 타이밍 순응성 (Conformance)

[0161] 일부 구현예들 (예컨대, SHVC, MV-HEVC, 등)에서, 출력 타이밍 순응성을 위한 DPB로부터의 픽처들의 출력 및 제거가 아래에서 설명되는 바와 같이 수행된다. 플러싱 프로세스에 관련된 부분들은 이탈릭체로 나타낸다.

아래 예에서, 호출된 픽처들의 제거는 HEVC 사양의 섹션 F.13.3.2에 규정된 바와 같이 각각의 계층에 특정적이다.

C.3.2DPB로부터의 픽처들의 제거

[0163] 현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 제거는 (현재의 픽처를 포함한) 액세스 유닛 n의 제 1 디코딩 유닛의 CPB 제거 시간에 동시에 발생하며, 다음과 같이 속행한다:

[0164] - 하위 조항 8.3.2에 규정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.

[0165] - 현재의 픽처가 픽처 0이 아닌, 1과 동일한 NoRas1OutputFlag를 가지는 IRAP 픽처일 때, 다음 순서로된 단계들이 적용된다:

[0166] 1. 변수 NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 대해 다음과 같이 유도된다:

[0167] - 현재의 픽처가 CRA 픽처이면, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계 없이) 1과 동일하게 설정된다.

[0168] - 그렇지 않고, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과 각각 상이하면, NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag의 값에 관계 없이, 1로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

[0169] 주 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이를 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

[0170] - 그렇지 않으면, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

[0171] 2. NoOutputOfPriorPicsFlag의 값이 1과 동일할 때 DPB에서의 모든 픽처 스토리지 버퍼들이 그들이 포함하는 픽처들의 출력 없이 비워지도록, 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값이 HRD에

적용되며, DPB 충만도는 0 과 동일하게 설정된다.

[0172] - 다음 조건들 양자가 DPB 에서의 임의의 픽처들 k 에 대해 참일 때, DPB 에서의 모든 이러한 픽처들 k 는 DPB로부터 제거된다:

[0173] - 픽처 k 는 "참조용으로 미사용됨" 으로서 마킹된다

[0174] - 픽처 k 가 0 과 동일한 PicOutputFlag 를 가지거나 또는 그의 DPB 출력 시간이 현재의 픽처 n 의 제 1 디코딩 유닛 (디코딩 유닛 m 으로 표시됨) 의 CPB 제거 시간 이하이다; 즉 DpbOutputTime[k] 는 CpbRemovalTime(m) 이하이다

[0175] - DPB로부터 제거되는 각각의 픽처에 대해, DPB 충만도가 1 만큼 감소된다.

F.13.3.2DPB로부터의 픽처들의 제거

[0177] 하위 조항 C.3.2 에서의 사양들은 nuh_layer_id 의 특정의 값을 가지는 디코딩된 픽처들의 각각의 세트에 대해, 다음 변경들과 함께, 별개로 적용한다.

[0178] - "현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 제거는 (현재의 픽처를 포함한) 액세스 유닛 n 의 제 1 디코딩 유닛의 CPB 제거 시간에 동시에 발생하며, 다음과 같이 속행한다:" 를 "현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 제거는 픽처 n 의 제 1 디코딩 유닛의 CPB 제거 시간에 동시에 발생하며 다음과 같이 속행한다:" 로 대체한다.

출력 순서 순응성

[0180] 일부 구현예들 (예컨대, SHVC, MV-HEVC, 등) 에서, 출력 순서 순응성을 위한 DPB로부터의 픽처들의 출력 및 제거가 아래에서 설명되는 바와 같이 수행된다. 플러싱 프로세스에 관련된 부분들은 이탈릭체로 나타낸다.

아래 예에서, 픽처들의 제거는, 호출될 때, 모든 계층들에 대해 수행된다.

F.13.5.2.2DPB로부터의 픽처들의 출력 및 제거

[0182] 현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 출력 및 제거는 현재의 픽처의 제 1 디코딩 유닛이 CPB로부터 제거될 때 동시에 발생하며, 다음과 같이 속행한다:

[0183] 하위 조항 F.8.3.2 에 규정된 바와 같은 RPS 에 대한 디코딩 프로세스가 호출된다.

[0184] - 현재의 픽처가 픽처 0 이 아닌, 1 과 동일한 NoRaslOutputFlag 및 0 과 동일한 nuh_layer_id 를 가지는 IRAP 픽처이면, 다음 순서로된 단계들이 적용된다:

[0185] 1. 변수 NoOutputOfPriorPicsFlag 는 테스트 중인 디코더에 대해 다음과 같이 유도된다:

[0186] - 현재의 픽처가 CRA 픽처이면, NoOutputOfPriorPicsFlag 는 (no_output_of_prior_pics_flag 의 값에 관계 없이) 1 과 동일하게 설정된다.

[0187] - 그렇지 않고, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid] 의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid] 의 값과 각각 상이하면, NoOutputOfPriorPicsFlag 는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag 의 값에 관계 없이, 1 로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

[0188] 주 - NoOutputOfPriorPicsFlag 를 no_output_of_prior_pics_flag 와 동일하게 설정하는 것이 이를 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag 를 1 로 설정하도록 허용된다.

[0189] - 그렇지 않으면, NoOutputOfPriorPicsFlag 는 no_output_of_prior_pics_flag 와 동일하게 설정된다.

[0190] 2. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag 의 값이 HRD 에 대해 다음과 같이 적용된다:

[0191] - NoOutputOfPriorPicsFlag 가 1 과 동일하면, DPB 에서의 모든 픽처 스토리지 버퍼들은 그들이 포함하는 픽처

들의 출력 없이 비워지며, DPB 충만도는 0 과 동일하게 설정된다.

[0192] - 그렇지 않으면 (*NoOutputOfPriorPicsFlag* 가 0 과 동일하다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨"으로서 마킹되는 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워지며, DPB 에서의 모든 비어 있지 않은 픽처 스토리지 버퍼들은 하위 조항 F.13.5.2.4 에 규정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지며, DPB 충만도는 0 과 동일하게 설정된다.

[0193] - 그렇지 않으면 (현재의 픽처가 1 과 동일한 *NoRas1OutputFlag* 를 가지거나 또는 0 과 동일하지 않은 *nuh_layer_id* 를 가지는 IRAP 픽처가 아니다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨"으로서 마킹된 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워진다. 비어있는 각각의 픽처 스토리지 버퍼에 대해, DPB 충만도가 1 만큼 감분된다. 변수 *currLayerId* 는 현재의 디코딩된 픽처의 *nuh_layer_id* 와 동일하게 설정되며, 그리고 다음 조건들 중 하나 이상이 참일 때, 하위 조항 F.13.5.2.4 에 규정된 "범핑" 프로세스가 반복적으로 호출되면서, 비어 있는 각각의 추가적인 픽처 스토리지 버퍼에 대해, 다음 조건들 중 어느 조건도 참이 아닐 때까지 DPB 충만도를 1 만큼 추가 감분시킨다:

[0194] - "출력용으로 요구됨"으로서 마킹되는 DPB 에서의 *currLayerId* 와 동일한 *nuh_layer_id* 를 가지는 픽처들의 개수는 활성 SPS (*currLayerId* 가 0 과 동일할 때) 로부터의, 또는 *currLayerId*의 값에 대한 활성 계층 SPS (*currLayerId* 가 0 과 동일하지 않을 때) 로부터의 *sps_max_num_reordered_pics[HighestTid]* 보다 크다.

[0195] - 활성 SPS (*currLayerId* 가 0 과 동일할 때), 또는 *currLayerId*의 값에 대한 활성 계층 SPS 의 *sps_max_latency_increase_plus1[HighestTid]* 는 0 과 동일하지 않으며, 연관된 변수 *PicLatencyCount[currLayerId]* 가 활성 SPS (*currLayerId* 가 0 과 동일할 때) 로부터 또는 *currLayerId*의 값에 대한 활성 계층 SPS 로부터 유도된 *SpsMaxLatencyPictures[HighestTid]* 보다 크거나 또는 동일한, "출력용으로 요구됨"으로서 마킹되는 DPB 에서의 *currLayerId* 와 동일한 *nuh_layer_id* 를 가지는 적어도 하나의 픽처가 존재한다.

[0196] - DPB 에서의 *currLayerId* 와 동일한 *nuh_layer_id* 를 가진 픽처들의 개수는 활성 SPS (*currLayerId* 가 0 과 동일할 때) 로부터 또는 *currLayerId*의 값에 대한 활성 계층 SPS 로부터의 *sps_max_dec_pic_buffering_minus1[HighestTid] + 1* 보다 크거나 또는 동일하다.

출력 타이밍 순응성과 출력 순서 순응성의 비교

[0197] 위에서 설명된 바와 같이, 출력 타이밍 순응성 및 출력 순서 순응성은 양쪽 다 동일한 플러싱 거동을 초래하는 것은 아닐 수도 있다. 예를 들어, 출력 타이밍 순응성 디코더들에 대해, 비트스트림에서 계층의 제 1 픽처가 아니고 "1" 의 값과 동일한 *NoRas1OutputFlag* 를 가지는 계층에서의 각각의 픽처에 대해 플러싱이 호출된다.

플러싱이 호출될 때, DPB 에서의 그 계층의 모든 디코딩된 픽처들이 플러싱된다. 한편, 출력 순서 순응성 디코더들에 대해, 비트스트림에서 제 1 픽처가 아니고 "1" 의 값과 동일한 *NoRas1OutputFlag* 를 가지는 기초 계층에서의 픽처에 대해서만 플러싱이 호출된다. 플러싱이 호출될 때, DPB 에서의 모든 계층들의 모든 디코딩된 픽처들이 플러싱된다.

[0198] 2개의 계층들을 가지는 비트스트림에서, IRAP 픽처이고 IRAP AU 에 속하지 않는 EL 에서의 LIP 픽처가 상이한 해상도를 활성화하고 그리고 BL 픽처가 비-IRAP AU 이기 (예컨대, IRAP 픽처가 아니기) 때문에 BL 의 해상도가 이 AU 에서 변할 수 없을 때, 픽처들의 계층-특정의 플러싱이 요망될 수도 있다. 여기서, 오직 BL 이 아닌, EL 로부터의 픽처들만이 플러싱되어야 한다. 이 특징은 출력 순서 순응성을 위해 이용불가능하다.

[0199] 2개의 계층들을 가지는 비트스트림에서, 액세스 유닛이 IDR 픽처인 BL 픽처 및 비-IRAP 픽처인 EL 픽처를 포함하는 경우에, BL 픽처의 해상도는 액세스 유닛에서 업데이트될 수도 있으며, 반면 EL 픽처의 해상도는 업데이트되지 않는다. 이러한 경우, 플러싱은 단지 BL 로부터의 픽처들에 대해서만 수행되어야 하며, EL 픽처들은 플러싱되지 않아야 한다. 이 특징은 출력 순서 순응성을 위해 이용불가능하다.

출력 타이밍 순응성 및 출력 순서 순응성의 SPS 활성화

[0200] "0" 의 값과 동일한 *nuh_layer_id* 를 가지는 SPS 가 하나 보다 많은 계층을 가지는 비트스트림에서 "0" 의 값과 동일한 *nuh_layer_id* 를 가지는 최종 VCL NAL 유닛 이후에 액세스 유닛에 존재하고, 그리고 디코딩 순서에서 다음 액세스 유닛이 (예컨대, 0 보다 큰 *nuh_layer_id* 값들을 가지는 추가적인 VCL NAL 유닛들이 비트스트림에 존

재한다는 것을 표시하는) 액세스 유닛 구분문자를 가질 때, 오직 BL 만을 포함하는 출력 계층 세트로 추출된 비트스트림은 비-순응적 (non-conforming) 일 수도 있다. 예를 들어, 추출된 비트스트림은 그의 VCL NAL 유닛들의 모두 이후에 SPS NAL 유닛을 포함할 수도 있다. 이러한 SPS NAL 유닛은 랭글링 SPS 로서 지정될 수도 있다. 디코더들은 일반적으로 SPS NAL 유닛을 프로세싱한 후 비트스트림에서 뒤따르도록 추가적인 VCL NAL 유닛들을 프로세싱하리라는 것을 예상한다. 따라서, 일부 코딩 방식들에서, 이러한 랭글링 SPS 는 비-순응 비트스트림을 초래할 수도 있다.

[0203] 예시적인 실시형태들

[0204] 위에서 설명된 어떤 문제들을 해결하는데 사용될 수도 있는 여러 방법들이 아래에서 설명될 것이다. 이들 방법들 중 일부는 독립적으로 적용될 수도 있으며, 그들 중 일부는 조합하여 적용될 수도 있다. 게다가, 본원에서 설명되는 방법들 중 하나 이상을 구현하는데 사용될 수도 있는 예시적인 선택 및 의미들이 또한 아래에 제공된다. 본원에서 설명되는 방법들 중 하나 이상을 구현하기 위해 포함될 수도 있는 추가 부분들 및 삭제 부분들을 예시하기 위해 HEVC 사양의 어떤 부분들이 재현될 때, 이러한 변경들은 이탈릭체 및 **취소선**으로 각각 나타낸다.

[0205] 픽처들의 계층-특정의 플러싱

[0206] 일부 실시형태들에서, 픽처들의 플러싱은 양쪽 유형들의 디코더 순응성들 (예컨대, 출력 타이밍 순응성 및 출력 순서 순응성)에 대해 계층-특정의 방법으로 수행된다. 플러싱 프로세스는 (예컨대, 단지 "1"의 값과 동일한 NoRas1OutputFlag 및 "0"의 값과 동일한 nuh_layer_id 를 가지는 IRAP 픽처에서만 일어나는 대신) "1"의 값과 동일한 NoRas1OutputFlag 를 가지는 각각의 IRAP 픽처에서 그리고 각각의 LIP 픽처에서 일어날 수도 있다 (또는, 일어나도록 인에이블될 수도 있다).

[0207] 이전 픽처들의 출력을 표시하는 플래그의 시그널링

[0208] 일부 실시형태들에서, 플래그 no_output_of_prior_pics_flag 는 (예컨대, "0"의 값과 동일한 nuh_layer_id 를 가지는) BL 에서의 모든 IRAP 픽처들에 대해 시그널링되며, 플래그 no_output_of_prior_pics_flag 는 "0"의 값과 동일하지 않은 nuh_layer_id 를 가지는 모든 VCL NAL 유닛들의 슬라이스 세그먼트 헤더에서 시그널링된다. 위에서 논의한 바와 같이, no_output_of_prior_pics_flag 는 이전에 디코딩되어 DPB 에 저장된 픽처들이 출력됨이 없이 DPB로부터 제거되어야 하는지 여부를 표시할 수도 있다. 다른 실시형태들에서, 플래그 no_output_of_prior_pics_flag 는 모든 VCL NAL 유닛들의 슬라이스 세그먼트 헤더에서 시그널링된다.

[0209] 현재의 픽처가 (예컨대, "0"의 값보다 큰 nuh_layer_id 를 가지는) EL 내에 있으면, no_output_of_prior_pics_flag 가 시그널링되기 전에 (예컨대, 현재의 픽처가 IRAP 픽처인지 여부를) 일반적으로 체킹될 수도 있는 조건들은 스kip될 수 있다. 예를 들어, 플래그 no_output_of_prior_pics_flag 는 비트스트림에 존재하는 각각의 EL 에 대해 시그널링될 수도 있다. 플래그 no_output_of_prior_pics_flag 는 (예컨대, 조건들이 만족되는지 여부를 체킹하는 여분의 단계 없이) 선택스 테이블에서 원래 위치에 존재할 수도 있다. 현재의 픽처가 IRAP 픽처가 아닌 (예컨대, "0"의 값과 동일한 nuh_layer_id 를 가지는) BL 픽처이면, 플래그 no_output_of_prior_pics_flag (또는, 유사한 표시 및/또는 기능을 가지는 다른 플래그) 는 슬라이스 헤더에서의 예약된 비트들 중 하나로서 또는 슬라이스 헤더 확장판의 부분으로서 존재할 수도 있다.

현재의 픽처가 (예컨대, "0"의 값과 동일한 nuh_layer_id 를 가지는) BL 픽쳐이고 IRAP 픽쳐이면, 플래그 no_output_of_prior_pics_flag 의 시그널링은 변경되지 않고 유지할 수도 있다.

[0210] 액세스 유닛 조건들에 기초한 픽처들의 출력

[0211] 일부 실시형태들에서, 변수 NoOutputOfPriorPicsFlag (예컨대, DPB 가 플러싱되기 전에 DPB에서의 픽처들을 출력할지 여부를 결정하기 위해 디코더에 의해 유도되는 값) 는 적어도 IRAP 픽처들이 아닌 모든 LIP 픽처들에 대해, no_output_of_prior_pics_flag 및 다른 조건들에 기초하여 유도된다. 예를 들어, no_output_of_prior_pics_flag 는 비트스트림에서 시그널링되는 값일 수도 있으며, 반면

NoOutputOfPriorPicsFlag 는 비트스트림에 포함된 정보에 기초하여 인코더에 의해 유도된 값일 수도 있다. 디코더는 no_output_of_prior_pics_flag 의 값 및 다른 조건들에 기초하여 NoOutputOfPriorPicsFlag의 값을 유도할 수도 있으며, 그후 유도된 NoOutputOfPriorPicsFlag 의 값을 이용하여 픽처들을 출력할지 여부를 결정할 수도 있다. 일부 실시형태들에서, IRAP 픽처가 아닌 각각의 LIP 픽처 picA 에 대한 NoOutputOfPriorPicsFlag의 값은, picA 를 포함하는 액세스 유닛에 속하며, "0" 의 값과 동일한 nuh_layer_id 를 가지며, 그리고 "1" 의 값과 동일한 NoRas1OutputFlag 를 가지는 IRAP 픽처와 연관된 NoCIRasOutputFlag 의 값에 기초하여 추론될 수도 있다. 일부 실시형태들에서, 플래그 NoOutputOfPriorPicsFlag 는 현재의 액세스 유닛이 2개의 상이한 비트스트림들이 함께 스티칭되는 스플라이스 지점을 포함하는지 여부를 표시할 수도 있다.

[0212] 일부 실시형태들에서, NoCIRasOutputFlag 및 NoRas1OutputFlag 는 비트스트림에 포함된 정보에 기초하여 유도된 변수들일 수도 있다. 예를 들어, NoRas1OutputFlag 는 (예컨대, BL 및/또는 EL 에서) 모든 IRAP 픽처에 대해 유도될 수도 있으며, NoCIRasOutputFlag 는 단지 최저 계층 픽처들 (예컨대, BL 픽처들)에 대해서만 유도될 수도 있다. NoCIRasOutputFlag 및 NoRas1OutputFlag 의 각각의 값은 비트스트림에서의 일부 픽처들이 어떤 참조 픽처들의 이용불가능성으로 인해 정확하게 디코딩가능하지 않을 수도 있다는 것을 표시할 수도 있다. 참조 픽처들의 이러한 이용불가능성은 랜덤 액세스 지점들에서 발생할 수도 있다. 교차-계층 랜덤 액세스 스킵 (CL-RAS) 픽처들은 일부 방법들에서, RASL 픽처들의 멀티-계층 등가물이다. 디코더가 랜덤 액세스 지점 (예컨대, BL IRAP 픽처를 가지는 액세스 유닛)에서 비트스트림을 디코딩하기 시작하고, 그리고 액세스 유닛에서의 EL 픽처가 IRAP 픽처가 아니면, 그 EL 픽처는 CL-RAS 픽처이다. EL 에서의 모든 픽처들은 IRAP 픽처가 EL 에서 발생할 때까지 (예컨대, 디코딩가능하지만, 정확하게 디코딩불가능한) CL-RAS 픽처들일 수도 있다. 이러한 EL IRAP 픽처가 비트스트림에 제공될 때, EL 은 초기화되었다라고 할 수도 있다.

[0213] 예를 들어, 도 4 의 예에서, EL 픽처 (422A) 는 IRAP 픽처가 아닌 LIP 픽처일 수도 있으며, BL 픽처 (422B) 는 연관된 플래그 NoCIRasOutputFlag 를 가지는 IRAP 픽처일 수도 있다. 이 예에서, EL 픽처 (422A) 와 연관된 NoOutputOfPriorPicsFlag 의 값은 BL 픽처 (422B) 와 연관된 NoCIRasOutputFlag 의 값에 기초하여 추론될 수도 있다. 예를 들어, NoCIRasOutputFlag 가 "1" 의 값과 동일하면, EL 픽처 (422A) 에 대한 NoOutputOfPriorPicsFlag 는, 또한 DPB 에서의 픽처들이 그들이 DPB 로부터 제거되기 전에 출력되지 않도록 초래하는 "1" 의 값으로 설정될 수도 있다. 한편, NoCIRasOutputFlag 가 "0" 의 값과 동일하면, EL 픽처 (422A) 에 대한 NoOutputOfPriorPicsFlag 는, 또한 DPB 에서의 픽처들이 출력 이후 DPB 로부터 제거되도록 초래하는 "0" 의 값으로 설정될 수도 있다.

액세스 유닛 조건들에 기초하여 픽처들을 출력하는 예시적인 플로우차트

[0214] 도 5 를 참조하여, DPB 를 플러싱하는 예시적인 루틴이 설명될 것이다. 도 5 는 본 개시물의 일 실시형태에 따른, 비디오 정보를 코딩하는 방법 (500) 을 예시하는 플로우차트이다. 도 5 에 예시된 단계들은 인코더 (예컨대, 도 2a 또는 도 2b 에 도시된 바와 같은 비디오 인코더), 디코더 (예컨대, 도 3a 또는 도 3b 에 도시된 바와 같은 비디오 디코더), 또는 임의의 다른 구성요소에 의해 수행될 수도 있다. 편의를 위해, 본 방법 (500) 은 인코더, 디코더, 또는 또 다른 구성요소일 수도 있는 코더에 의해 수행되는 것으로 설명된다.

[0215] 방법 (500) 은 블록 501 에서 시작한다. 블록 505 에서, 코더는 픽처가 스플라이스 지점 비-IRAP 픽처인지 여부를 결정한다. 예를 들어, 코더는 그 픽처가 스플라이스 지점에 바로 뒤따르는 액세스 유닛에 있는 비-IRAP 픽처인지 여부를 결정할 수도 있다. 일부 실시형태들에서, 특정의 픽처가 스플라이스 지점에 바로 뒤따르는 액세스 유닛에 있는지 여부가 플래그로서 시그널링되거나 또는 프로세싱될 수도 있다. 이러한 실시형태들에서, 1 의 플래그 값은 픽처가 스플라이스 지점에 바로 뒤따르는 액세스 유닛에 있다는 것을 표시할 수도 있으며, 0 의 플래그 값은 픽처가 스플라이스 지점에 바로 뒤따르는 액세스 유닛에 있지 않다는 것을 표시할 수도 있다. 픽처가 스플라이스 지점 비-IRAP 픽처가 아니라고 코더가 결정하면, 본 방법 (500) 은 블록 510 으로 진행한다. 픽처가 스플라이스 지점 비-IRAP 픽처라고 코더가 결정하면, 본 방법 (500) 은 블록 515 로 진행한다.

[0216] 블록 510 에서, 코더는 DPB 로부터 픽처들을 제거하기 전에 DPB 에서의 픽처들을 출력한다. 블록 515 에서, 코더는 픽처들을 출력함이 없이 DPB 에서의 픽처들을 제거한다. 방법 (500) 은 515 에서 종료한다.

[0217] 위에서 논의한 바와 같이, 도 2a 의 비디오 인코더 (20), 도 2b 의 비디오 인코더 (23), 도 3a 의 비디오 디코더 (30), 또는 도 3b 의 비디오 디코더 (33) 의 하나 이상의 구성요소들 (예컨대, 인터-계층 예측 유닛 (128)

및/또는 인터-계층 예측 유닛 (166)) 은, 픽처가 스플라이스 지점 비-IRAP 픽처인지 여부를 결정하고, 그리고 픽처들을 출력하고 및/또는 DPB로부터 픽처들을 제거하는 것과 같은, 본 개시물에서 논의되는 기법들 중 임의의 기법을 구현하는데 사용될 수도 있다.

[0219] DPB 플러싱 방법들의 조합

[0220] 일부 실시형태들에서, 플러싱 프로세스는 또한 IRAP 픽처들인 EL 픽처들 중 하나에서 호출될 때에만 오직 계층-특정적이다. 플러싱 프로세스가 "1"의 값과 동일한 NoRas1OutputFlag 를 가지는 BL에 속하는 IRAP 픽처에서 호출될 때, 모든 계층들 전체에 걸친 모든 픽처들이 DPB로부터 플러싱될 수도 있다.

[0221] 네트워크 추상화 계층 (NAL) 유닛들 및 파라미터 세트들

[0222] 위에서 논의한 바와 같이, 인코더 또는 디코더에 의해 사용되는 파라미터들은, 그들이 이용될 수도 있는 코딩 레벨에 기초하여 파라미터 세트들로 그룹화될 수도 있다. 예를 들어, 비트스트림에서의 하나 이상의 코딩된 비디오 시퀀스들에 의해 이용되는 파라미터들은 비디오 파라미터 세트 (VPS)에 포함될 수도 있으며, 코딩된 비디오 시퀀스에서의 하나 이상의 픽처들에 의해 이용되는 파라미터들은 시퀀스 파라미터 세트 (SPS)에 포함될 수도 있다. 이와 유사하게, 픽처에서의 하나 이상의 슬라이스들에 의해 이용되는 파라미터들은 픽처 파라미터 세트 (PPS)에 포함될 수도 있으며, 단일 슬라이스에 고유한 다른 파라미터들은 슬라이스 헤더에 포함될 수도 있다. 이러한 파라미터 세트들은 파라미터 세트 NAL 유닛들 (예컨대, SPS NAL 유닛, PPS NAL 유닛, 등)에 의한 주어진 계층에 대해 활성화될 수도 있다 (또는, 활성화로서 표시될 수도 있다). NAL 유닛은 미가공 바이트 시퀀스 페이로드 (RBSP) 및 NAL 유닛 헤더를 포함한다. RBSP는 파라미터 세트 ID (예컨대, SPS ID)를 규정할 수도 있으며, NAL 유닛 헤더는 어느 계층들이 SPS를 이용할 수 있는지를 표시할 수도 있는 계층 ID를 규정할 수도 있다.

[0223] 일부의 경우, 주어진 계층에 의해 사용될 수도 있거나 또는 그에 대해 활성화될 수도 있는 특정의 파라미터 세트의 비트스트림에 다수의 인스턴스들을 제공하는 것이 유익할 수도 있다. 예를 들어, 특정의 파라미터 세트가 주어진 계층에 대해 이미 활성화된 이후에도, 특정의 파라미터 세트의 추가적인 인스턴스가 주어진 계층에 의한 사용을 위해 비트스트림에 제공될 수도 있다. 비트스트림이 SPS의 이러한 추가적인 인스턴스를 포함할 때, 이전에 시그널링된 특정의 파라미터 세트가 손실되더라도, 디코더는 주어진 계층에 대해, 후속 시그널링된 특정의 파라미터 세트를 이용할 수도 있다.

[0224] 그러나, 일부 코딩 방식들에서, 이러한 파라미터 세트 (예컨대, 반복된 SPS)의 추가적인 인스턴스가 비트스트림으로 제공될 때, 파라미터 세트 (예컨대, SPS NAL 유닛)의 콘텐츠는 파라미터 세트의 모든 이전 인스턴스들과 동일하도록 요구될 수도 있다. 예를 들어, 비트스트림이 기초 계층 및 향상 계층을 포함하면, SPS NAL 유닛이 비트스트림에 제공된 후, 기초 계층 및 향상 계층 양자는 SPS를 참조할 수도 있다. 기초 계층의 모든 NAL 유닛들 (예컨대, VCL NAL 유닛들)이 비트스트림에 제공된 후, 예를 들어, 에러 복원력을 향상시키기 위해 SPS를 다시 비트스트림에 제공하는 것이 바람직한 상황이 있을 수도 있다. 이러한 상황에서, 후속 SPS NAL 유닛은 비트스트림에서 이전에 제공된 SPS NAL 유닛과 동일한 콘텐츠를 가지도록 요구될 수도 있다. 어느 계층들이 SPS NAL 유닛들을 이용할 수도 있는지를 표시하기 위해 SPS NAL 유닛들이 계층 ID를 규정할 수도 있기 때문에, 위에서 설명된 예에서, 후속 SPS NAL 유닛이 향상 계층에 의해 단독으로 사용될 수도 있더라도, 기초 계층 및 향상 계층 양자가 SPS를 이용할 수도 있다는 것을 표시할 수도 있는, 이전에 제공된 SPS NAL 유닛과 동일한 계층 ID를 규정하기 위해 비트스트림 제약에 의해 후속 SPS NAL 유닛이 요구될 수도 있다. SPS NAL 유닛들 양자가 동일한 계층 ID를 규정하면서도 상이한 계층들에 의해 사용되면, 디코딩 프로세스 동안 문제점들이 발생할 수도 있다.

[0225] 예를 들어, 특정의 비트스트림이 기초 계층 및 향상 계층을 가지면, 기초 계층 및 향상 계층 양자는 동일한 SPS를 참조한다. EL이 BL보다 더 높은 프레임 레이트를 가지면, 비트스트림에서의 마지막 몇 개의 액세스 유닛들은 단지 EL 픽처들만을 포함하고 어떤 BL 픽처들은 포함하지 않을 수도 있다. 이러한 예에서, 마지막 몇 개의 액세스 유닛들 중 하나가 반복된 SPS NAL 유닛을 포함하면, 위에서 설명된 비트스트림 제약은 SPS NAL 유닛의 계층 ID가 SPS를 활성화하는 이전 SPS NAL 유닛과 동일하도록 강제할 수도 있다 (예컨대, 코더는 이러한 비트스트림 제약이 적용가능하다고 결정하고, 코딩된 비트스트림이 비트스트림 제약을 따르도록 비트스트림 제약을 따를 수도 있다). 예를 들어, 이러한 이전 SPS NAL 유닛은 기초 계층에 의해 사용될 수도 있으며,

이전 SPS NAL 유닛은 기초 계층이 SPS 를 사용할 수도 있다는 것을 표시하는 0 의 계층 ID 값을 가질 수도 있다. 이러한 경우, 반복된 SPS NAL 유닛이 기초 계층에 의해 사용되도록 의도되지 않더라도, 반복된 SPS NAL 유닛의 계층 ID 는 또한 "0" 의 값과 동일하여야 할 것이다. 디코더가 이 예에서 (예컨대, 0 의 계층 ID 를 가지는 모든 NAL 유닛들을 취함으로써) 비트스트림의 기초 계층을 추출하려고 시도하면, 최종 비트스트림은 반복된 SPS NAL 유닛을 비트스트림의 끝에 가질 것이다. 이것은, 디코더가 반복된 SPS NAL 유닛을 프로세싱 시, 반복된 SPS NAL 유닛이 다음 액세스 유닛 (또는, 코딩된 비디오 시퀀스) 의 시작 부분을 시그널링한다고 가정할 수도 있기 때문에, 문제가 있을 수도 있다. 이러한 문제를 피하기 위해, 인코더는 후속 SPS NAL 유닛을 비트스트림에 전혀 제공하지 않기로 결정함으로써, 반복된 SPS NAL 유닛들과 연관된 잠재적인 이익들을 포기할 수도 있다.

시퀀스 파라미터 세트 (SPS) 미가공 바이트 시퀀스 페이로드 (RBSP) 의 활성화

SPS RBSP 는 활성 파라미터 세트를 SEI 메시지를 포함하는 하나 이상의 SEI NAL 유닛들 또는 하나 이상의 픽처 파라미터 세트 (PPS) RBSP들에 의해 참조될 수 있는 파라미터들을 포함한다. 각각의 SPS RBSP 는 디코딩 프로세스의 시작 시에 임의의 계층에 대해 활성화되지 않는 것으로 처음에 간주될 수도 있다. 각각의 계층에 대해, 많아 봐야 하나의 SPS RBSP 는 디코딩 프로세스 동안 임의의 주어진 순간에 활성인 것으로 간주되며, 특정의 계층에 대한 임의의 특정의 SPS RBSP 의 활성화는, 있다면, 그 특정의 계층에 대해 이전-활성 SPS RBSP 의 비활성화를 초래한다.

하나의 SPS RBSP 는 하나 보다 많은 계층에 대한 활성 SPS RBSP 일 수도 있다. 예를 들어, 기초 계층 및 향상 계층 양쪽이 3 의 SPS ID 를 가지는 SPS 를 결과적으로 참조하는 PPS 를 참조하는 픽처를 포함하면, 3 의 SPS ID 를 가지는 SPS 는 참조 계층 및 향상 계층 양쪽에 대해 활성 SPS RBSP 이다.

(예컨대, 특정의 SPS ID 를 가지는) SPS RBSP 가 X 의 계층 ID (예컨대, `nuh_layer_id`) 를 가지는 (예컨대, 비-제로 계층 ID 값 또는 0 보다 큰 계층 ID 를 가지는) 특정의 비-기초 계층에 대해 아직 활성이 아니고 SPS RBSP 가 픽처 파라미터 세트 (PPS) RBSP 에서 참조되면, SPS RBSP 가 특정의 비-기초 계층에 대해 활성화된다. 이 SPS 는 특정의 비-기초 계층에 대해 다른 SPS RBSP 의 활성화에 의해 비활성화될 때까지 특정의 비-기초 계층에 대해 활성 SPS RBSP 로서 참조될 수도 있다.

파라미터 세트들의 활성화

위에서 논의한 바와 같이, 일부 코딩 방식들에서, 계층 초기화 픽처 ("LIP 픽처") 는 (1) 1 로 설정된 `NoRas1OutputFlag` 플래그 (예컨대, `RASL` 픽처들이 출력되어야 하는지 여부를 표시하는 플래그) 를 가지는 IRAP 픽처인 픽처 또는 (2) 기초 계층 픽처 (예컨대, 0 의 계층 ID 또는 비트스트림에서 정의된 가장 작은 계층 ID 를 가지는 픽처) 가 1 로 설정된 `NoRas1OutputFlag` 를 가지는 IRAP 액세스 유닛인 초기 IRAP 액세스 유닛에 포함되는 픽처로서 정의된다.

일부 실시형태들에서, SPS 는 각각의 LIP 픽처에서 활성화될 수 있다. 예를 들어, 1 로 설정된 `NoRas1OutputFlag` 플래그를 가지는 각각의 IRAP 픽처 또는 초기 IRAP 액세스 유닛에 포함되는 각각의 픽처는, (예컨대, 픽처 사이즈, 등과 같은 상이한 파라미터들을 가지는) 이전에 활성화된 SPS 와 동일하거나 또는 상이 할 수도 있는, 새로운 SPS 를 활성화할 수 있다. 또, SPS 가 활성화된 후, 동일한 SPS 의 추가적인 인스턴스가 비트스트림에 제공될 수도 있다. 이러한 SPS 의 반복된 인스턴스 (또는, 반복된 SPS) 는 이전에 제공된 SPS 가 비트스트림으로부터 손실 또는 드롭 (drop) 된 경우에 백업 SPS 로서 기능함으로써 여러 복원력을 향상시킬 수도 있다.

반복된 SPS 를 비트스트림에 제공하는 예시적인 플로우차트

도 6 은 본 개시물의 일 실시형태에 따른, 비디오 정보를 코딩하는 방법 (600) 을 예시하는 플로우차트이다. 도 6 에 예시된 단계들은 인코더 (예컨대, 도 2a 또는 도 2b 에 도시된 바와 같은 비디오 인코더), 디코더 (예컨대, 도 3a 또는 도 3b 에 도시된 바와 같은 비디오 디코더), 또는 임의의 다른 구성요소에 의해 수행될 수도 있다. 편의를 위해, 본 방법 (600) 은 인코더, 디코더, 또는 또 다른 구성요소일 수도 있는 코더에 의해

수행되는 것으로 설명된다.

[0235] 방법 (600) 은 블록 601에서 시작한다. 블록 605에서, 코더는 시퀀스 파라미터 세트 (SPS) 를 SPS 가 제 1 비디오 계층 및 제 2 비디오 계층에 대해 활성화될 수도 있다는 표시와 함께, 비트스트림에 제공한다. 예를 들어, 제 1 비디오 계층은 기초 계층일 수도 있으며, 제 2 비디오 계층은 향상 계층일 수도 있다. 제 2 비디오 계층은 제 1 비디오 계층과는 상이한 계층 ID 를 가지는 임의의 계층일 수도 있다. SPS 는 비트스트림에 계층 ID 및 SPS ID 를 가지는 SPS NAL 유닛의 유형으로 제공될 수도 있다. 예를 들어, SPS NAL 유닛은 SPS 가 제 1 및 제 2 비디오 계층들 양쪽에 대해 활성화될 수도 있다는 것을 표시하는 계층 ID 를 가질 수도 있다. 일부 실시형태들에서, SPS 의 계층 ID 가 0 의 값을 가지면, SPS 는 "0" 의 값보다 크거나 또는 동일한 계층 ID 를 가지는 임의의 계층에 대해 활성화될 수도 있다. 예를 들어, 기초 계층이 0 의 계층 ID 를 가지고 향상 계층이 1 의 계층 ID 를 가지는 경우에, SPS 의 계층 ID 가 0 의 값을 가지면, SPS 는 기초 계층 및 향상 계층 양쪽에 대해 활성화될 수도 있다.

[0236] 블록 610에서, 코더는 동일한 SPS (예컨대, 이전에 제공된 SPS NAL 유닛과 동일한 SPS ID 를 가지는 SPS NAL 유닛인, 반복된 SPS) 를, SPS 가 제 2 비디오 계층에 대해 활성화될 수도 있지만 제 1 비디오 계층에 대해 활성화되지 않을 수도 있다는 표시와 함께, 비트스트림에 제공한다. 예를 들어, 반복된 SPS NAL 유닛은 이전에 제공된 SPS NAL 유닛과는 상이한 계층 ID 를 가질 수도 있다. 기초 계층이 0 의 계층 ID 및 향상 계층이 1 의 계층 ID 를 가지는 경우, 반복된 SPS 의 계층 ID 가 1 의 값을 가지면, 반복된 SPS 는 (예컨대, 0 의 계층 ID 값을 가지는) 기초 계층이 아닌 (예컨대, 1 의 계층 ID 값을 가지는) 향상 계층에 의해 활성화될 수도 있다.

하나 이상의 하위 계층들 (예컨대, 기초 계층) 의 모든 VCL NAL 유닛들이 제공된 후 반복된 SPS 가 비트스트림에 제공되면, 이전에 제공된 SPS 와는 동일한 SPS ID 를 갖지만 상이한 계층 ID 를 가지는 반복된 SPS 를 제공하는 것이 유용할 수도 있다. 반복된 SPS 가 하나 이상의 하위 계층들에 대해 활성화되지 않아야 한다는 것을 표시하는 계층 ID 를 가지는 반복된 SPS 를 제공함으로써, 하나 이상의 하위 계층들이 비트스트림으로부터 추출될 때, 최종 추출된 비트스트림은, 모든 VCL NAL 유닛들이 반복된 SPS 이전에 제공되었기 때문에, 하위 계층들에 대해 요구되지 않는 반복된 SPS 를 포함하지 않을 것이다. 방법 (600) 은 615에서 종료한다.

[0237] 위에서 논의한 바와 같이, 도 2a 의 비디오 인코더 (20), 도 2b 의 비디오 인코더 (23), 도 3a 의 비디오 디코더 (30), 또는 도 3b 의 비디오 디코더 (33) 의 하나 이상의 구성요소들 (예컨대, 인터-계층 예측 유닛 (128) 및/또는 인터-계층 예측 유닛 (166)) 은, 시퀀스 파라미터 세트 (SPS) 를, SPS 가 제 1 비디오 계층 및 제 2 비디오 계층에 대해 활성화될 수도 있다는 표시와 함께, 비트스트림에 제공하고, 동일한 SPS 를, SPS 가 제 2 비디오 계층에 대해 활성화될 수도 있지만 제 1 비디오 계층에 대해 활성화되지 않을 수도 있다는 표시와 함께, 비트스트림에 제공하는 것과 같은, 본 개시물에서 논의되는 기법들 중 임의의 기법을 구현하는데 사용될 수도 있다.

[0238] 방법 (600) 에서, 도 6 에 도시된 블록들 중 하나 이상이 제거될 (예컨대, 수행되지 않을) 수도 있으며 및/또는 본 방법이 수행되는 순서는 스위칭될 수도 있다. 일부 실시형태들에서, 추가적인 블록들이 방법 (600) 에 추가될 수도 있다. 방법 (600) 은 SPS 를 참조하여 설명되지만, 방법 (600) 과 관련하여 설명되는 기법들은 VPS, PPS, 및 슬라이스 헤더와 같은, 다른 파라미터 세트들로 확장되어 적용될 수 있는 것으로 인식되어야 한다. 따라서, 본 개시물의 실시형태들은 도 6 에 나타낸 예에 또는 그에 의해 한정되지 않으며, 다른 변형 예들이 본 개시물의 정신으로부터 일탈함이 없이 구현될 수도 있다.

반복된 SPS 에 관한 비트스트림 제약 (Constraint)

[0240] 일부 실시형태들에서, 비트스트림 제약은, SPS NAL 유닛 (예컨대, 반복된 SPS) 이 이전에 시그널링된 SPS 와 동일한 SPS ID 값 (예컨대, sps_seq_parameter_set_id) 을 포함할 때, 반복된 SPS 가 활성 SPS 가 활성 상태로 유지하도록 요구되는 최종 코딩된 픽처에 뒤따르지 않는 한, 그리고 동일한 SPS ID 값 (예컨대, sps_seq_parameter_set_id) 을 가지는 SPS 를 활성화하는 제 1 NAL 유닛에 선행하지 않는 한, 반복된 SPS NAL 유닛의 SPS RBSP 는 이전에 시그널링된 SPS NAL 유닛의 콘텐츠와 동일한 콘텐츠를 가져야 한다는 것을 규정할 수도 있다.

예시적인 구현예들 #1

[0242] 아래에서 설명하는 실시형태들에서, no_output_of_prior_pics_flag 의 값은 모든 VCL NAL 유닛들의 슬라이스

세그먼트 헤더들에서 시그널링된다. 본원에서 설명되는 방법들 중 하나 이상을 구현하기 위해 포함될 수도 있는 추가 부분들 및 삭제 부분들을 예시하기 위해 HEVC 사양의 어떤 부분들이 재현될 때, 이러한 변경들은 이 텔릭체 및 ~~주소선~~ 으로 각각 나타낸다.

[0243] 슬라이스 세그먼트 헤더 신팩스

다음 예시적인 신팩스가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC 사양에서의 기준 표현에 대한 추가부분들은 이텔릭체로 표시된다.

slice_segment_header() {	디스크립터
first_slice_segment_in_pic_flag	u(1)
if(<i>nuh_layer_id</i> > 0 (<i>nal_unit_type</i> >= BLA_W_LP && <i>nal_unit_type</i> <= RSV_IRAP_VCL23))	
no_output_of_prior_pics_flag	u(1)
slice_pic_parameter_set_id	ue(v)
...	

[0245]

테이블 1. 예시적인 슬라이스 세그먼트 헤더 신팩스

[0247] 슬라이스 세그먼트 헤더 의미들

다음 예시적인 의미들이 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC 사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 이텔릭체 및 ~~주소선~~ 으로 각각 표시된다.

[0249] **no_output_of_prior_pics_flag** 는 부속서 C 에 규정된 바와 같이 비트스트림에서 제 1 액세스 유닛 ~~또는~~ 포함되지 않은 IDR ~~또는~~, BLA 픽처, 또는 LIP 의 디코딩 후, 디코딩된 픽처 버퍼에서의 이전에-디코딩된 픽처들의 출력에 영향을 미친다.

[0250] 일 실시형태에서, **base_no_output_of_prior_pics_flag** 는 IRAP 가 아닌 BL 픽처에 대해 그것이 LIP 픽처가 될 수 있도록 시그널링될 수도 있다.

[0251] **base_no_output_of_prior_pics_flag** 는 부속서 C 에 규정된 바와 같이 비트스트림에서 제 1 액세스 유닛에 포함되지 않은 계층 초기화 픽처 이후 디코딩된 픽처 버퍼에서의 이전에-디코딩된 픽처들의 출력에 영향을 미친다. 존재할 때 **no_output_of_prior_pics_flag** 는 **base_no_output_of_prior_pics_flag** 와 동일하게 설정된다.

[0252] 다른 실시형태에서, **base_no_output_of_prior_pics_flag** 는 시그널링되지 않는다.

[0253] 활성화 프로세스에 대한 변경들

[0254] 현재의 HEVC 사양 (예컨대, 섹션 F.7.4.2.4.2) 에서의 활성화 프로세스는 아래에 나타낸 바와 같이 수정되며, 프로세스의 나머지는 동일하다. HEVC 사양에서의 기준 표현에 대한 추가부분들은 이텔릭체로 표시된다.

[0255] 활성 계층 SPS RBSP 에 대한 **sps_seq_parameter_set_id** 의 값을 포함하는 임의의 SPS NAL 유닛은, 그것이 활성 계층 SPS 가 활성인 것으로 요구되는 최종 코딩된 픽처에 뒤따르지 않고 동일한 **seq_parameter_set_id** 의 값의 SPS 를 활성화하는 제 1 NAL 유닛에 선행하지 않는 한, 활성 계층 SPS RBSP 의 콘텐츠와 동일한 SPS RBSP 의 콘텐츠를 가져야 한다.

[0256] 일부 실시형태들에서, 유사한 제약들이 비디오 파라미터 세트들 (VPS) 및 픽처 파라미터들 세트들 (PPS) 과 같은 다른 파라미터 세트들에 추가될 수도 있다.

[0257] DPB로부터의 픽처들의 제거에 대한 변경들

[0258] 다음 예시적인 텍스트가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC 사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 이탤릭체 및 취소선으로 각각 표시된다.

[0259] C.3.2DPB로부터의 픽처들의 제거

[0260] 현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 제거는 (현재의 픽처를 포함한) 액세스 유닛 n의 제 1 디코딩 유닛의 CPB 제거 시간에 동시에 발생하며, 다음과 같이 속행한다:

[0261] - 하위 조항 8.3.2에 규정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.

[0262] - 현재의 픽처가 ~~1과 동일한 NoRaslOutputFlag를 가진 IRAP 픽처~~ (픽처 0이 아닌 LIP (또는, 1과 동일한 NoRaslOutputFlag 및 0과 동일한 nuh_layer_id를 가지는 IRAP 픽처) 이면, 다음 순서로된 단계들이 적용된다:

[0263] 1. 변수 NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 대해 다음과 같이 유도된다:

[0264] - 현재의 픽처가 CRA 픽처이면, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계 없이) 1과 동일하게 설정된다.

[0265] - 그렇지 않고, 현재의 픽처의 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 현재의 픽처와 동일한 계층 ID를 가지는 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과 각각 상이하면, NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag의 값에 관계 없이, 1로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

[0266] 주 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이들 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

[0267] - 그렇지 않으면, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

[0268] 2. NoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에 있고 그리고 현재의 픽처의 그것과 동일한 nuh_layer_id의 값을 가지는 픽처들을 포함하는 모든 픽처 스토리지 버퍼들이 그들이 포함하는 픽처들의 출력 없이 비워지도록, 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값이 HRD에 적용되며, DPB 충만도는 ~~0과 동일하게 설정된~~ 제거된 픽처들의 개수 만큼 감분된다.

[0269] - 다음 조건들 양자가 DPB에서의 임의의 픽처들 k에 대해 참일 때, DPB에서의 모든 이러한 픽처들 k는 DPB로부터 제거된다:

[0270] - 픽처 k는 "참조용으로 미사용됨"으로서 마킹된다

[0271] - 픽처 k가 0과 동일한 PicOutputFlag를 가지거나 또는 그의 DPB 출력 시간이 현재의 픽처 n의 제 1 디코딩 유닛 (디코딩 유닛 m으로 표시됨)의 CPB 제거 시간 이하이다; 즉 DpbOutputTime[k]는 CpbRemovalTime(m) 이하이다

[0272] - DPB로부터 제거되는 각각의 픽처에 대해, DPB 충만도가 1 만큼 감분된다.

[0273] DPB로부터의 픽처들의 출력 및 제거에 대한 변경들

[0274] 다음 예시적인 텍스트가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC 사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 이탤릭체 및 **취소선**으로 각각 표시된다.

[0275] F.13.5.2.2DPB로부터의 픽처들의 출력 및 제거

[0276] 현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 포함한 후) DPB로부터의 픽처들의 출력 및 제거는 현재의 픽처의 제 1 디코딩 유닛이 CPB로부터 제거될 때 동시에 발생하며, 다음과 같이 속행한다:

[0277] 하위 조항 F.8.3.2에 규정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.

[0278] - 현재의 ~~1과 동일한 NoRas1OutputFlag를 갖고 0과 동일한~~ ~~nuh_layer_id를 가진 IRAP 픽처~~ 픽처 0이 아닌 LIP (또는, 1과 동일한 NoRas1OutputFlag 및 0과 동일한 ~~nuh_layer_id를 가진 IRAP 픽처~~ 이면, 다음 순서로된 단계들이 적용된다:

[0279] 1. 변수 NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 대해 다음과 같이 유도된다:

[0280] - 현재의 픽처가 CRA 픽처이면, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계 없이) 1과 동일하게 설정된다.

[0281] - 그렇지 않고, 현재의 픽처의 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 현재의 픽처와 동일한 계층 ID를 가지는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과 각각 상이하면, NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag의 값에 관계 없이, 1로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

[0282] 주 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이를 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

[0283] - 그렇지 않으면, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

[0284] 2. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값이 HRD에 대해 다음과 같이 적용된다:

[0285] - NoOutputOfPriorPicsFlag가 1과 동일하면, DPB에 있고 그리고 현재의 픽처의 그것과 동일한 ~~nuh_layer_id~~의 값을 가지는 픽처들을 포함하는 모든 픽처 스토리지 버퍼들은 그들이 포함하는 픽처들의 출력 없이 비워지며, DPB 충만도는 ~~0과 동일하게 설정된~~ 제거된 픽처들의 개수 만큼 감분된다.

[0286] - 그렇지 않으면 (NoOutputOfPriorPicsFlag가 0과 동일하다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨"으로서 마킹되는 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워지며, DPB에서의 모든 비어 있지 않은 픽처 스토리지 버퍼들은 하위 조항 F.13.5.2.4에 규정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지며, DPB 충만도는 0과 동일하게 설정된다.

[0287] - 그렇지 않으면 (현재의 픽처가 1과 동일한 NoRas1OutputFlag를 가지거나 또는 0과 동일하지 않은 ~~nuh_layer_id~~를 가지는 IRAP 픽처가 아니다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨"으로서 마킹된 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워진다. 비어있는 각각의 픽처 스토리지 버퍼에 대해, DPB 충만도가 1 만큼 감분된다. 변수 currLayerId는 현재의 디코딩된 픽처의 ~~nuh_layer_id~~와 동일하게 설정되며, 그리고 다음 조건들 중 하나 이상이 참일 때, 하위 조항 F.13.5.2.4에 규정된 "범핑" 프로세스가 반복적으로 호출되면서, 비어 있는 각각의 추가적인 픽처 스토리지 버퍼에 대해, 다음 조건들 중 어느 조건도 참이 아닐 때까지 DPB 충만도를 1 만큼 추가 감분시킨다:

[0288] - "출력용으로 요구됨"으로서 마킹되는 DPB에서의 currLayerId와 동일한 ~~nuh_layer_id~~를 가지는 픽처들의 개수는 활성 SPS (currLayerId가 0과 동일할 때)로부터의, 또는 currLayerId의 값에 대한 활성 계층 SPS (currLayerId가 0과 동일하지 않을 때)로부터의 sps_max_num_reordered_pics[HighestTid]보다 크다.

[0289] - 활성 SPS (currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS의

sps_max_latency_increase_plus1[HighestTid] 는 0 과 동일하지 않으며, 연관된 변수 PicLatencyCount[currLayerId] 가 활성 SPS (currLayerId 가 0 과 동일할 때) 로부터 또는 currLayerId의 값에 대한 활성 계층 SPS 로부터 유도된 SpsMaxLatencyPictures[HighestTid] 보다 크거나 또는 동일한, "출력용으로 요구됨" 으로서 마킹되는 DPB 에서의 currLayerId 와 동일한 nuh_layer_id 를 가지는 적어도 하나의 픽처가 존재한다.

- [0290] - DPB 에서의 currLayerId 와 동일한 nuh_layer_id 를 가진 픽처들의 개수는 활성 SPS (currLayerId 가 0 과 동일할 때) 로부터 또는 currLayerId의 값에 대한 활성 계층 SPS 로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1 보다 크거나 또는 동일하다.

[0291] 예시적인 구현예들 #2

- [0292] 아래에서 설명하는 실시형태들에서, no_output_of_prior_pics_flag 는 단지 IRAP 픽처들에 대해서만 시그널링되며, NoOutputOfPriorPicsFlag 는 "1" 의 값과 동일한 NoRas1OutputFlag 를 가지는 모든 IRAP 픽처들에 대해 유도되며, NoC1RasOutputFlag의 값에 기초하여, LIP 픽처들인 모든 비-IRAP 픽처들에 대해 추론된다.

[0293] DPB 로부터의 픽처들의 제거에 대한 변경들

- [0294] 다음 예시적인 텍스트가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC 사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 이탤릭체 및 **취소선** 으로 각각 표시된다.

[0295] C.3.2DPB 로부터의 픽처들의 제거

- [0296] 현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB 로부터의 픽처들의 제거는 (현재의 픽처를 포함한) 액세스 유닛 n 의 제 1 디코딩 유닛의 CPB 제거 시간에 동시에 발생하며, 다음과 같이 속행한다:

- [0297] - 하위 조항 8.3.2 에 규정된 바와 같은 RPS 에 대한 디코딩 프로세스가 호출된다.
- [0298] - 현재의 픽처가 ~~1과 동일한 NoRas1OutputFlag 를 가지는 IRAP 픽처~~ 픽처 0 이 아닌 LIP (또는, 1 과 동일한 NoRas1OutputFlag 및 0 과 동일한 nuh_layer_id 를 가지는 IRAP 픽처) 일 때, 다음 순서로된 단계들이 적용된다:

- [0299] 1. 변수 NoOutputOfPriorPicsFlag 는 테스트 중인 디코더에 대해 다음과 같이 유도된다:
- [0300] - 현재의 픽처가 1 과 동일한 NoRas1OutputFlag 를 가지는 CRA 픽처이면, NoOutputOfPriorPicsFlag 는 (no_output_of_prior_pics_flag 의 값에 관계 없이) 1 과 동일하게 설정된다.
- [0301] - 그렇지 않고, 현재의 픽처의 활성 SPS 로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid] 의 값이 현재의 픽처와 동일한 계층 ID 를 가지는 선행하는 픽처에 대해 활성인 SPS 로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid] 의 값과 각각 상이하면, NoOutputOfPriorPicsFlag 는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag 의 값에 관계 없이, 1 로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

- [0302] 주 - NoOutputOfPriorPicsFlag 를 no_output_of_prior_pics_flag 와 동일하게 설정하는 것이 이를 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag 를 1 로 설정하도록 허용된다.

- [0303] - 그렇지 않고, 현재의 픽처가 IRAP 픽처이면, NoOutputOfPriorPicsFlag 는 no_output_of_prior_pics_flag 와 동일하게 설정된다.

- [0304] - 그렇지 않고, (LIP 인) 현재의 픽처가 IRAP 픽처가 아니면 그리고 현재의 액세스 유닛에서의 nuh_layer_id 가 0 과 동일한 IRAP 픽처와 연관된 NoC1rasOutputFlag 가 1 과 동일하면, NoOutputOfPriorPicsFlag 는 1 과 동일하게 설정된다.

- [0305] - 그렇지 않으면, *NoOutputOfPriorPicsFlag* 는 0 과 동일하게 설정된다.
- [0306] 2. *NoOutputOfPriorPicsFlag* 의 값이 1 과 동일할 때, DPB 에 있고 그리고 현재의 픽처의 그것과 동일한 *nuh_layer_id* 의 값을 가지는 픽처들을 포함하는 모든 픽처 스토리지 버퍼들이 그들이 포함하는 픽처들의 출력 없이 비워지도록, 테스트 중인 디코더에 대해 유도된 *NoOutputOfPriorPicsFlag* 의 값이 HRD 에 적용되며, DPB 충만도는 ~~0과 동일하게 설정된다~~ 제거된 픽처들의 개수 만큼 감소된다.
- [0307] - 다음 조건들 양자가 DPB 에서의 임의의 픽처들 k 에 대해 참일 때, DPB 에서의 모든 이러한 픽처들 k 는 DPB 로부터 제거된다:
- [0308] - 픽처 k 는 "참조용으로 미사용됨" 으로서 마킹된다
- [0309] - 픽처 k 가 0 과 동일한 *PicOutputFlag* 를 가지거나 또는 그의 DPB 출력 시간이 현재의 픽처 n 의 제 1 디코딩 유닛 (디코딩 유닛 m 으로 표시됨) 의 CPB 제거 시간 이하이다; 즉 *DpbOutputTime[k]* 는 *CpbRemovalTime(m)* 이하이다
- [0310] - DPB 로부터 제거되는 각각의 픽처에 대해, DPB 충만도가 1 만큼 감소된다.
- [0311] DPB 로부터의 픽처들의 출력 및 제거에 대한 변경들
- [0312] 다음 예시적인 텍스트가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC 사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 **이탤릭체** 및 **취소선** 으로 각각 표시된다.
- [0313] **F.13.5.2.2DPB 로부터의 픽처들의 출력 및 제거**
- [0314] 현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB 로부터의 픽처들의 출력 및 제거는 현재의 픽처의 제 1 디코딩 유닛이 CPB 로부터 제거될 때 동시에 발생하며, 다음과 같이 속행한다:
- [0315] 하위 조항 F.8.3.2 에 규정된 바와 같은 RPS 에 대한 디코딩 프로세스가 호출된다.
- [0316] - 현재의 픽처가 ~~1과 동일한 NoRas1OutputFlag 를 가지는 IRAP 픽처~~ 픽처 0 이 아닌 *LIP* (또는, 1 과 동일한 *NoRas1OutputFlag* 및 0 과 동일한 *nuh_layer_id* 를 가지는 *IRAP* 픽처) 이면, 다음 순서로된 단계들이 적용된다:
1. 변수 *NoOutputOfPriorPicsFlag* 는 테스트 중인 디코더에 대해 다음과 같이 유도된다:
- [0318] - 현재의 픽처가 *CRA* 픽처이면, *NoOutputOfPriorPicsFlag* 는 (*no_output_of_prior_pics_flag* 의 값에 관계 없이) 1 과 동일하게 설정된다.
- [0319] - 그렇지 않고, 현재의 픽처의 활성 SPS 로부터 유도된 *pic_width_in_luma_samples*, *pic_height_in_luma_samples*, 또는 *sps_max_dec_pic_buffering_minus1[HighestTid]* 의 값이 현재의 픽처와 동일한 계층 *ID* 를 가지는 선행하는 픽처에 대해 활성인 SPS 로부터 유도된 *pic_width_in_luma_samples*, *pic_height_in_luma_samples*, 또는 *sps_max_dec_pic_buffering_minus1[HighestTid]* 의 값과 각각 상이하면, *NoOutputOfPriorPicsFlag* 는 테스트 중인 디코더에 의해, *no_output_of_prior_pics_flag* 의 값에 관계 없이, 1 로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).
- [0320] 주 - *NoOutputOfPriorPicsFlag* 를 *no_output_of_prior_pics_flag* 와 동일하게 설정하는 것이 이를 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 *NoOutputOfPriorPicsFlag* 를 1 로 설정하도록 허용된다.
- [0321] - 그렇지 않고, 현재의 픽처가 *IRAP* 픽처이면, *NoOutputOfPriorPicsFlag* 는 *no_output_of_prior_pics_flag* 와 동일하게 설정된다.
- [0322] - 그렇지 않고, (*LIP* 인) 현재의 픽처가 *IRAP* 픽처가 아니면 그리고 현재의 액세스 유닛에서의 *nuh_layer_id* 가 0 과 동일한 *IRAP* 픽처와 연관된 *NoC1rasOutputFlag* 가 1 과 동일하면, *NoOutputOfPriorPicsFlag* 는 1 과 동일하게 설정된다.

- [0323] - 그렇지 않으면, *NoOutputOfPriorPicsFlag* 는 0 과 동일하게 설정된다.
- [0324] 2. 테스트 중인 디코더에 대해 유도된 *NoOutputOfPriorPicsFlag* 의 값이 HRD 에 대해 다음과 같이 적용된다:
- [0325] - *NoOutputOfPriorPicsFlag* 가 1 과 동일하면, DPB 에 있으며 그리고 현재의 픽처의 그것과 동일한 *nuh_layer_id* 를 가지는 픽처들을 포함하는 모든 픽처 스토리지 버퍼들은 그들이 포함하는 픽처들의 출력 없이 비워지며, DPB 충만도는 ~~0과 동일하게 설정된~~ 제거된 픽처들의 개수 만큼 감분된다.
- [0326] - 그렇지 않으면 (*NoOutputOfPriorPicsFlag* 가 0 과 동일하다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨" 으로서 마킹되는 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워지며, DPB 에서의 모든 비어 있지 않은 픽처 스토리지 버퍼들은 하위 조항 F.13.5.2.4 에 규정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지며, DPB 충만도는 0 과 동일하게 설정된다.
- [0327] - 그렇지 않으면 (현재의 픽처가 1 과 동일한 *NoRaslOutputFlag* 를 가지거나 또는 0 과 동일하지 않은 *nuh_layer_id* 를 가지는 IRAP 픽처가 아니다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨" 으로서 마킹된 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워진다. 비어있는 각각의 픽처 스토리지 버퍼에 대해, DPB 충만도가 1 만큼 감분된다. 변수 *currLayerId* 는 현재의 디코딩된 픽처의 *nuh_layer_id* 와 동일하게 설정되며, 그리고 다음 조건들 중 하나 이상이 참일 때, 하위 조항 F.13.5.2.4 에 규정된 "범핑" 프로세스가 반복적으로 호출되면서, 비어 있는 각각의 추가적인 픽처 스토리지 버퍼에 대해, 다음 조건들 중 어느 조건도 참이 아닐 때까지 DPB 충만도를 1 만큼 추가 감분시킨다:
- [0328] - "출력용으로 요구됨"으로서 마킹되는 DPB 에서의 *currLayerId* 와 동일한 *nuh_layer_id* 를 가지는 픽처들의 개수는 활성 SPS (*currLayerId* 가 0 과 동일할 때) 로부터의, 또는 *currLayerId*의 값에 대한 활성 계층 SPS (*currLayerId* 가 0 과 동일하지 않을 때) 로부터의 *sps_max_num_reordered_pics[HighestTid]* 보다 크다.
- [0329] - 활성 SPS (*currLayerId* 가 0 과 동일할 때), 또는 *currLayerId*의 값에 대한 활성 계층 SPS 의 *sps_max_latency_increase_plus1[HighestTid]* 는 0 과 동일하지 않으며, 연관된 변수 *PicLatencyCount[currLayerId]* 가 활성 SPS (*currLayerId* 가 0 과 동일할 때) 로부터 또는 *currLayerId*의 값에 대한 활성 계층 SPS 로부터 유도된 *SpsMaxLatencyPictures[HighestTid]* 보다 크거나 또는 동일한, "출력용으로 요구됨" 으로서 마킹되는 DPB 에서의 *currLayerId* 와 동일한 *nuh_layer_id* 를 가지는 적어도 하나의 픽처가 존재한다.
- [0330] - DPB 에서의 *currLayerId* 와 동일한 *nuh_layer_id* 의 픽처들의 개수는 활성 SPS (*currLayerId* 가 0 과 동일할 때) 로부터 또는 *currLayerId*의 값에 대한 활성 계층 SPS 로부터의 *sps_max_dec_pic_buffering_minus1[HighestTid] + 1* 보다 크거나 또는 동일하다.
- [0331] 일부 실시형태들에서, 출력 타이밍 순응성 및 출력 순서 순응성 양쪽에서, 현재의 픽처가 IRAP 픽처가 아닌 LIP 픽처일 때, 그리고 현재의 액세스 유닛에서, "0" 의 값과 동일한 *nuh_layer_id* 를 가지는, IRAP 픽처인 다른 픽처와 연관된 *NoClrasOutputFlag* 가 "1" 의 값과 동일할 때, *NoOutputOfPriorPicsFlag* 는 "0" 의 값과 동일한 *nuh_layer_id* 를 가지는 현재의 AU 에서의 IRAP 픽처의 *NoOutputOfPriorPicsFlag* 와 동일하게 설정된다.
- [0332] 예시적인 구현예들 #3
- [0333] 아래에서 설명하는 실시형태들에서, "1" 의 값과 동일한 *NoRaslOutputFlag* 를 가지는 비-BL IRAP 픽처들을 디코딩할 때에 픽처들의 플러싱은 계층-특정의 방법으로 수행되도록 규정된다. 플러싱이 "1" 의 값과 동일한 *NoRaslOutputFlag* 를 가지는 BL IRAP 픽처들을 디코딩할 때에 이루어질 때, 플러싱 동작은 모든 계층들에 걸쳐서 수행되도록 규정된다. 예를 들어, 모든 플러싱은 비-기초 계층들 (예컨대, 향상 계층들) 에 대해 계층-특정적이며, 그러나 기초 계층과 관련하여 수행되는 플러싱 동작은 비-기초 계층들에서의 픽처들을 플러싱할 수도 있다.
- [0334] DPB 로부터의 픽처들의 제거에 대한 변경들
- [0335] 다음 예시적인 텍스트가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC

사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 **이탤릭체** 및 **취소선**으로 각각 표시된다.

[0336] C.3.2DPB로부터의 픽처들의 제거

현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 제거는 (현재의 픽처를 포함한) 액세스 유닛 n의 제 1 디코딩 유닛의 CPB 제거 시간에 동시에 발생하며, 다음과 같이 속행한다:

- 하위 조항 8.3.2에 규정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.

- 현재의 픽처가 1과 동일한 NoRas1OutputFlag를 가지는, 그리고 0과 동일한 nuh_layer_id를 가지는, 픽처 0이 아닌, IRAP 픽처, 또는 1과 동일한 NoRas1OutputFlag를 가지며 0과 동일하지 않은 nuh_layer_id를 가지는, 초기 IRAP AU에 속하지 않는, IRAP 픽처이면, 다음 순서로된 단계들이 적용된다:

1. 변수 NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 대해 다음과 같이 유도된다:

- 현재의 픽처가 0과 동일한 nuh_layer_id를 가지는 CRA 픽처이면, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계 없이) 1과 동일하게 설정된다.

- 그렇지 않고, 현재의 픽처의 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 현재의 픽처와 동일한 계층 ID를 가지는 선형하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과 각각 상이하면, NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag의 값에 관계 없이, 1로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

주 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이들 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

- 그렇지 않으면, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

1. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값이 HRD에 대해 다음과 같이, **다음을 하도록** 적용된다:

- NoOutputOfPriorPicsFlag의 값이 1과 동일하고 그리고 현재의 픽처의 nuh_layer_id가 0과 동일 **할 때** 하면, DPB에서의 모든 픽처 스토리지 버퍼들은 그들이 포함하는 픽처들의 출력 없이 비워지며, DPB 충만도는 0과 동일하게 설정된다.

- 그렇지 않고, NoOutputOfPriorPicsFlag가 1과 동일하고 현재의 픽처의 nuh_layer_id가 0과 동일하지 않으면, DPB에 있으며 그리고 현재의 픽처의 그것과 동일한 nuh_layer_id를 가지는 픽처들을 포함하는 모든 픽처 스토리지 버퍼들은 비워지며, DPB 충만도는 제거된 픽처들의 개수 만큼 감소된다.

- 다음 조건들 양자가 DPB에서의 임의의 픽처들 k에 대해 참일 때, DPB에서의 모든 이러한 픽처들 k는 DPB로부터 제거된다:

- 픽처 k는 "참조용으로 미사용됨"으로서 마킹된다

- 픽처 k가 0과 동일한 PicOutputFlag를 가지거나 또는 그의 DPB 출력 시간이 현재의 픽처 n의 제 1 디코딩 유닛 (디코딩 유닛 m으로 표시됨)의 CPB 제거 시간 이하이다; 즉 DpbOutputTime[k]는 CpbRemovalTime(m) 이하이다

- DPB로부터 제거되는 각각의 픽처에 대해, DPB 충만도가 1 만큼 감소된다.

[0352] DPB로부터의 픽처들의 출력 및 제거에 대한 변경들

다음 예시적인 텍스트가 본원에서 설명되는 실시형태들 중 하나 이상을 구현하는데 사용될 수도 있다. HEVC

사양에서의 기준 표현에 대한 추가부분들 및 삭제 부분들은 **이탤릭체** 및 **취소선**으로 각각 표시된다.

[0354] F.13.5.2.2DPB로부터의 픽처들의 출력 및 제거

현재의 픽처의 디코딩 이전 (그러나 현재의 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후) DPB로부터의 픽처들의 출력 및 제거는 현재의 픽처의 제 1 디코딩 유닛이 CPB로부터 제거될 때 동시에 발생하며, 다음과 같이 속행한다:

하위 조항 F.8.3.2에 규정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.

- 현재의 픽처가 1과 동일한 NoRas1OutputFlag를 가지는, 그리고 0과 동일한 nuh_layer_id를 가지는, 픽처 0이 아닌, IRAP 픽처, 또는 1과 동일한 NoRas1OutputFlag를 가지며 0과 동일하지 않은 nuh_layer_id를 가지는, 초기 IRAP AU에 속하지 않는, IRAP 픽처이면, 다음 순서로된 단계들이 적용된다:

1. 변수 NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 대해 다음과 같이 유도된다:

- 현재의 픽처가 0과 동일한 nuh_layer_id를 가지는 CRA 픽처이면, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계 없이) 1과 동일하게 설정된다.

- 그렇지 않고, 현재의 픽처의 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 현재의 픽처와 동일한 계층 ID를 가지는 실행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과 각각 상이하면, NoOutputOfPriorPicsFlag는 테스트 중인 디코더에 의해, no_output_of_prior_pics_flag의 값에 관계 없이, 1로 설정될 수도 있다 (그러나, 설정되지 않아야 한다).

주 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이들 조건들 하에서 선호되지만, 이 경우 테스트 중인 디코더는 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

- 그렇지 않으면, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

2. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값이 HRD에 대해 다음과 같이 적용된다:

- NoOutputOfPriorPicsFlag가 1과 동일하고 그리고 현재의 픽처의 nuh_layer_id가 0과 동일하면, DPB에서의 모든 픽처 스토리지 버퍼들은 그들이 포함하는 픽처들의 출력 없이 비워지며, DPB 충만도는 0과 동일하게 설정된다.

- 그렇지 않고, NoOutputOfPriorPicsFlag가 1과 동일하고 현재의 픽처의 nuh_layer_id가 0과 동일하지 않으면, DPB에 있으며 그리고 현재의 픽처의 그것과 동일한 nuh_layer_id를 가지는 픽처들을 포함하는 모든 픽처 스토리지 버퍼들은 비워지며, DPB 충만도는 제거된 픽처들의 개수 만큼 감소된다.

- 그렇지 않으면 (NoOutputOfPriorPicsFlag가 0과 동일하다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨"으로서 마킹되는 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워지며, DPB에서의 모든 비어 있지 않은 픽처 스토리지 버퍼들은 하위 조항 F.13.5.2.4에 규정된 "범평" 프로세스를 반복적으로 호출함으로써 비워지며, DPB 충만도는 0과 동일하게 설정된다.

- 그렇지 않으면 (현재의 픽처가 1과 동일한 NoRas1OutputFlag를 가지거나 또는 0과 동일하지 않은 nuh_layer_id를 가지는 IRAP 픽처가 아니다), "출력용으로 미요구됨" 및 "참조용으로 미사용됨"으로서 마킹되는 픽처를 포함하는 모든 픽처 스토리지 버퍼들은 (출력 없이) 비워진다. 비어있는 각각의 픽처 스토리지 버퍼에 대해, DPB 충만도가 1 만큼 감소된다. 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정되며, 그리고 다음 조건들 중 하나 이상이 참일 때, 하위 조항 F.13.5.2.4에 규정된 "범평" 프로세스가 반복적으로 호출되면서, 비어 있는 각각의 추가적인 픽처 스토리지 버퍼에 대해, 다음 조건들 중 어느 조건도 참이 아닐 때까지 DPB 충만도를 1 만큼 추가 감소시킨다:

- "출력용으로 요구됨"으로서 마킹되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 가지는 픽처들의 개수는 활성 SPS (currLayerId가 0과 동일할 때)로부터의, 또는 currLayerId의 값에 대한 활성 계층 SPS (currLayerId가 0과 동일하지 않을 때)로부터의 sps_max_num_reordered_pics[HighestTid]보다 크다.

[0369]

- 활성 SPS (currLayerId 가 0 과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS 의 sps_max_latency_increase_plus1[HighestTid] 는 0 과 동일하지 않으며, 연관된 변수 PicLatencyCount[currLayerId] 가 활성 SPS (currLayerId 가 0 과 동일할 때)로부터 또는 currLayerId의 값에 대한 활성 계층 SPS로부터 유도된 SpsMaxLatencyPictures[HighestTid] 보다 크거나 또는 동일한, "출력용으로 요구됨" 으로서 마킹되는 DPB 에서의 currLayerId 와 동일한 nuh_layer_id 를 가지는 적어도 하나의 픽처가 존재한다.

[0370]

- DPB 에서의 currLayerId 와 동일한 nuh_layer_id 의 픽처들의 개수는 활성 SPS (currLayerId 가 0 과 동일할 때)로부터 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1 보다 크거나 또는 동일하다.

[0371]

다른 고려사항들

[0372]

본원에서 개시된 정보 및 신호들은 다양한 상이한 기술들 및 기법들 중 어느 것을 이용하여서도 표현될 수도 있다. 예를 들어, 상기 설명 전반에 걸쳐서 인용될 수도 있는 데이터, 명령들, 지령들, 정보, 신호들, 비트들, 심볼들 및 칩들은, 전압들, 전류들, 전자기파들, 자기장들 또는 자기 입자들, 광학장들 또는 광학 입자들, 또는 이들의 임의의 조합으로 표현될 수도 있다.

[0373]

본원에서 개시된 실시형태들과 관련하여 설명된 여러가지 예시적인 로직 블록들, 모듈들, 회로들, 및 알고리즘 단계들은 전자적 하드웨어, 컴퓨터 소프트웨어, 또는 양쪽의 조합들로서 구현될 수도 있다. 이러한 하드웨어와 소프트웨어의 상호 교환가능성을 명확히 예시하기 위하여, 이상에서는, 여러 예시적인 구성요소들, 블록들, 모듈들, 회로들 및 단계들이 그들의 기능의 관점에서 일반적으로 설명되었다. 이런 기능이 하드웨어 또는 소프트웨어로 구현되는지 여부는 특정의 애플리케이션 및 전체 시스템에 부과되는 설계 제한 사항들에 의존한다. 숙련자들은 각각의 특정의 애플리케이션마다 설명한 기능을 여러가지 방법으로 구현할 수도 있으며, 그러나 이런 구현 결정들이 본 발명의 범위로부터 일탈을 초래하는 것으로 해석되어서는 안된다.

[0374]

본원에서 설명된 기법들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 이러한 기법들은 범용 컴퓨터들, 무선 통신 디바이스 핸드셋들, 또는 무선 통신 디바이스 핸드셋들 및 다른 디바이스들에서의 애플리케이션을 포함한 다수의 용도들을 가진 집적 회로 디바이스들과 같은 다양한 디바이스들 중 임의의 디바이스에서 구현될 수도 있다. 모듈들 또는 구성요소들로서 설명하는 임의의 특징들은 통합 로직 디바이스 내에 함께, 또는 별개의 공용가능한 로직 디바이스들로서 별개로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 이 기법들은 실행될 때, 위에서 설명된 방법들 중 하나 이상을 수행하는 명령들을 포함하는 프로그램 코드를 포함하는 컴퓨터 판독가능 데이터 저장 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터 판독가능 데이터 저장 매체는 패키징 재료들을 포함할 수도 있는 컴퓨터 프로그램 제품의 일부를 형성할 수도 있다. 컴퓨터 판독가능 매체는 메모리 또는 데이터 저장 매체들, 예컨대, 동기식 동적 랜덤 액세스 메모리 (SDRAM) 과 같은 랜덤 액세스 메모리 (RAM), 판독 전용 메모리 (ROM), 비-휘발성 랜덤 액세스 메모리 (NVRAM), 전기적으로 소거가능한 프로그래밍가능 판독 전용 메모리 (EEPROM), 플래시 메모리, 자기 또는 광학 데이터 저장 매체들 등을 포함할 수도 있다. 기법들은 추가적으로, 또는 대안적으로, 프로그램 코드를 명령들 또는 데이터 구조들의 유형으로 반송하거나 또는 통신하고 그리고 컴퓨터에 의해 액세스되거나, 판독되거나, 및/또는 실행될 수 있는, 전파 신호들 또는 파들과 같은, 컴퓨터 판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.

[0375]

프로그램 코드는 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 프로세서들을 포함할 수도 있는 프로세서에 의해 실행될 수도 있다. 이러한 프로세서는 본 개시물에서 설명하는 기법들 중 임의의 기법들을 수행하도록 구성될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있으며; 그러나 대안적으로는, 프로세서는 임의의 종래의 프로세서, 제어기, 마이크로제어기 또는 상태 머신일 수도 있다. 프로세서는 또한 컴퓨팅 디바이스들의 조합, 예컨대, DSP 와 마이크로프로세서의 조합, 복수의 마이크로프로세서들, DSP 코어와 결합된 하나 이상의 마이크로프로세서들, 또는 임의의 다른 이러한 구성으로서 구현될 수도 있다. 따라서, 용어 "프로세서" 는, 본원에서 사용될 때 전술한 구조 중 임의의 구조, 전술한 구조의 임의의 조합, 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조 또는 장치를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능은 인코딩 및 디코딩하도록 구성된 전용 소프트웨어 모듈들 또는 하드웨어 모듈들 내에 제공되거나, 또는 결합된 비디오 인코더-디코더 (코덱) 에 포함될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들로 전적으로 구현될

수 있다.

[0376]

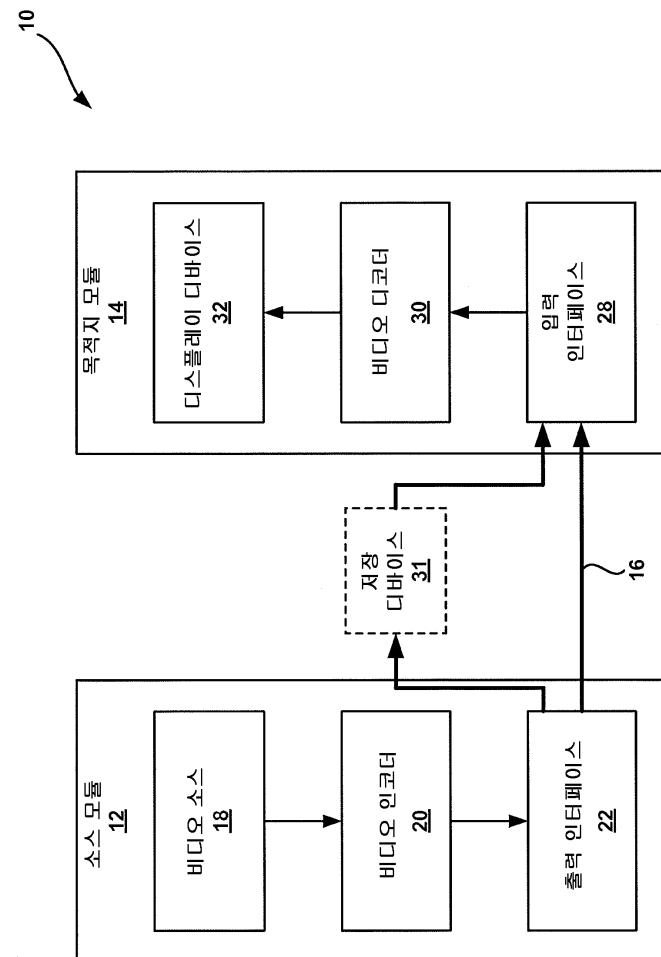
본 개시물의 기법들은 무선 핸드셋, 접적 회로 (IC) 또는 IC들의 세트 (예컨대, 칩 세트)를 포함한, 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 개시한 기법들을 수행하도록 구성되는 디바이스들의 기능적 양태들을 강조하기 위해서 여러 구성요소들, 모듈들, 또는 유닛들이 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다. 대신, 위에서 설명한 바와 같이, 여러 유닛들이 코텍 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명한 바와 같은 하나 이상의 프로세서들을 포함한, 상호작용하는 하드웨어 유닛들의 컬렉션으로 제공될 수도 있다.

[0377]

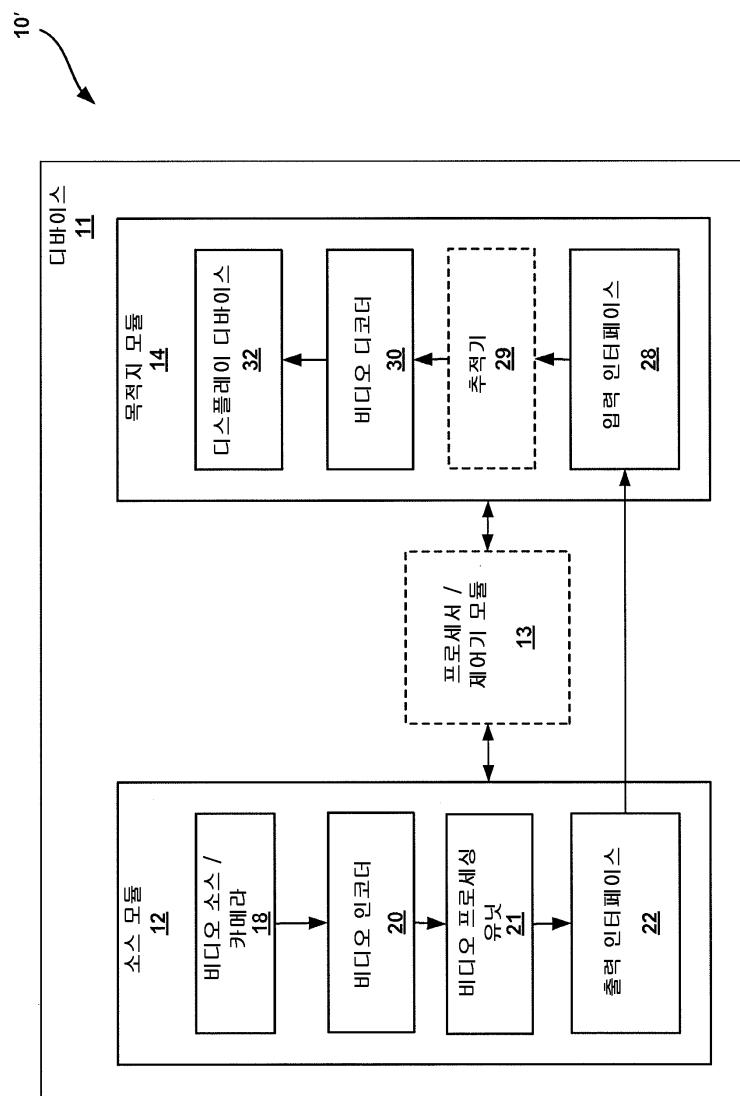
여러 본 발명의 실시형태들이 설명되었다. 이들 및 다른 실시형태들은 다음 청구항들의 범위 이내이다.

도면

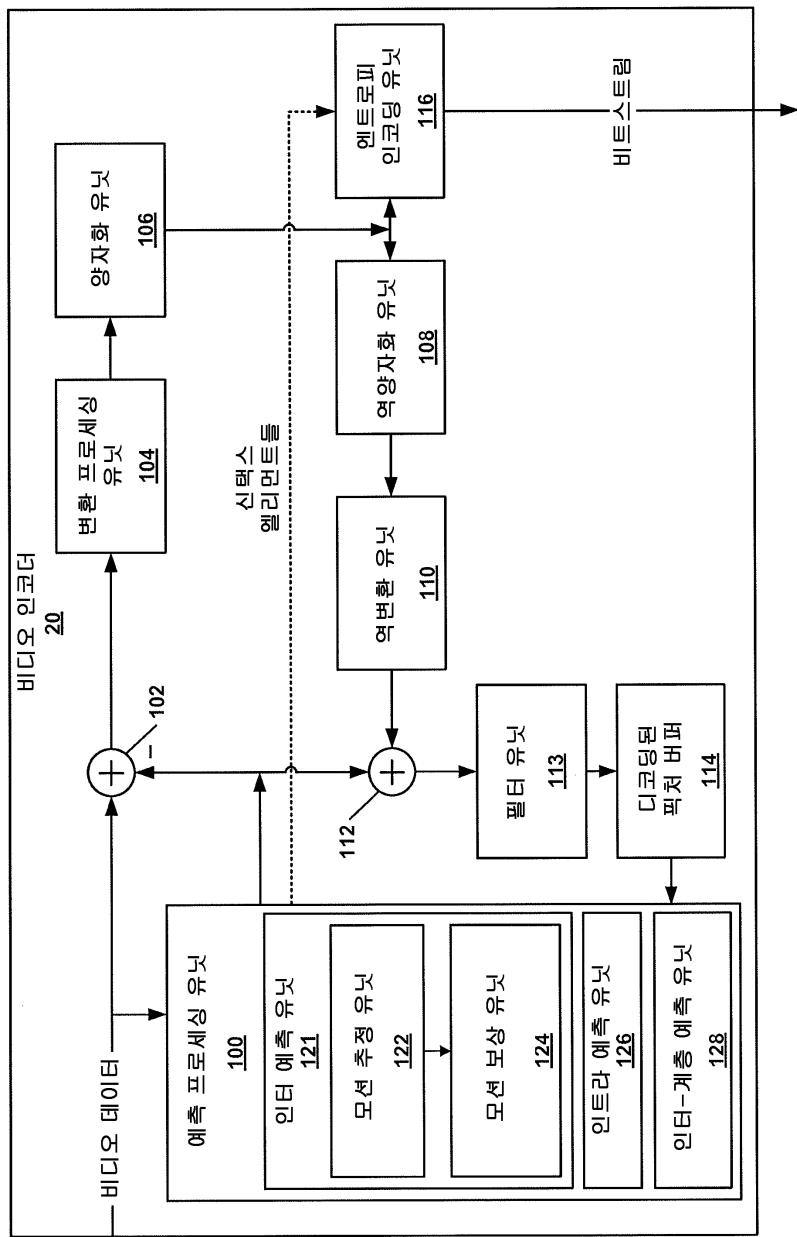
도면 1a



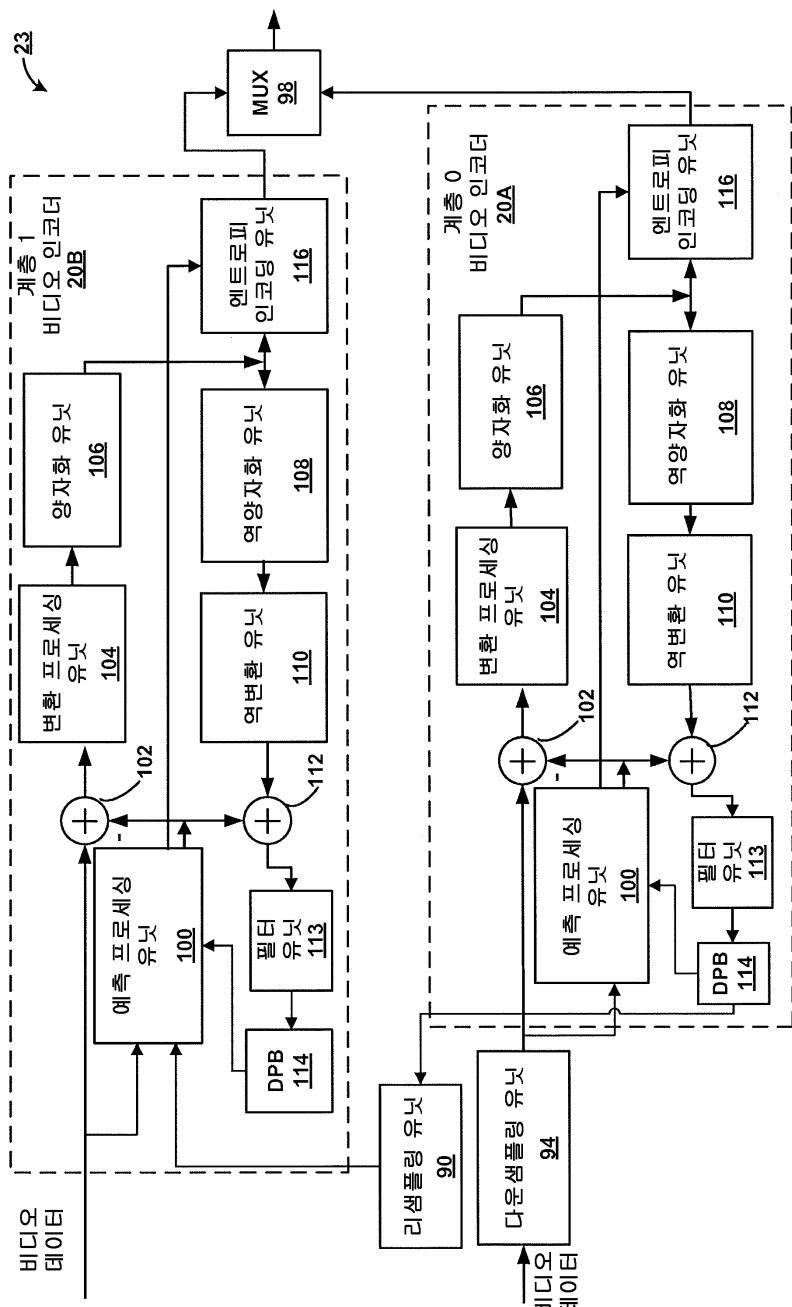
도면 1b



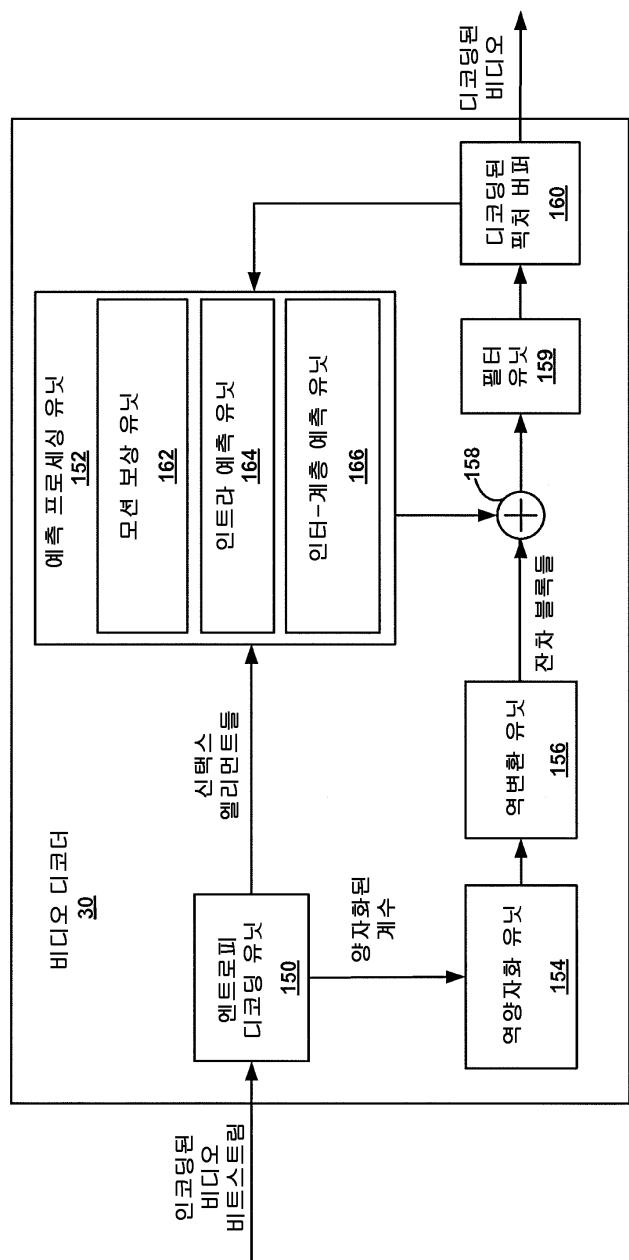
도면2a



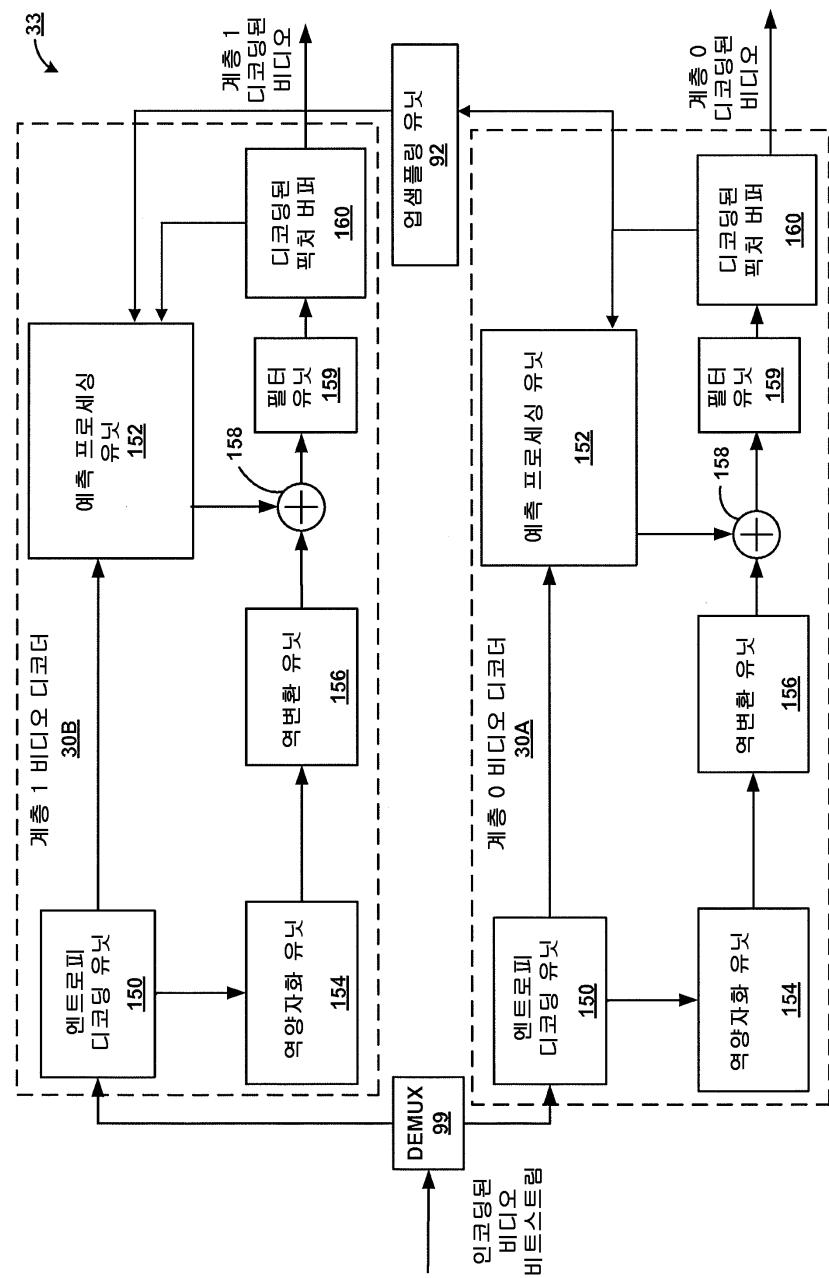
도면2b



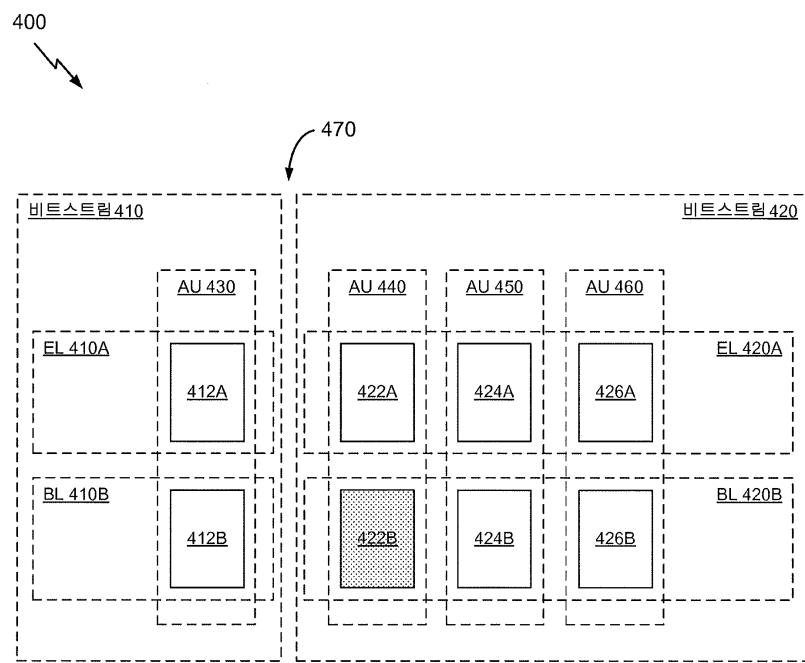
도면3a



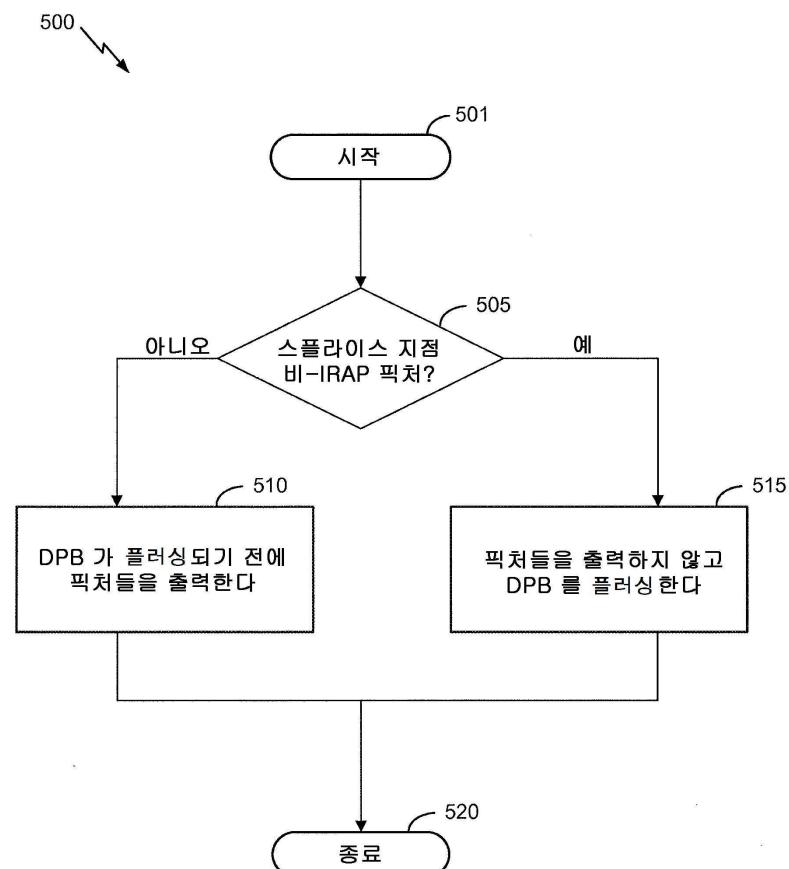
도면3b



도면4



도면5



도면6

