



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2022년08월19일

(11) 등록번호 10-2434872

(24) 등록일자 2022년08월17일

(51) 국제특허분류(Int. Cl.)
H04N 19/597 (2014.01) **H04N 19/172** (2014.01)
H04N 19/187 (2014.01) **H04N 19/30** (2014.01)
H04N 19/42 (2014.01) **H04N 19/46** (2014.01)
H04N 19/70 (2014.01)
(52) CPC특허분류
H04N 19/597 (2015.01)
H04N 19/172 (2015.01)
(21) 출원번호 10-2016-7028747
(22) 출원일자(국제) 2015년03월17일
심사청구일자 2020년02월26일
(85) 번역문제출일자 2016년10월14일
(65) 공개번호 10-2016-0135761
(43) 공개일자 2016년11월28일
(86) 국제출원번호 PCT/US2015/021065
(87) 국제공개번호 WO 2015/142921
국제공개일자 2015년09월24일
(30) 우선권주장
61/954,551 2014년03월17일 미국(US)
(뒷면에 계속)
(56) 선행기술조사문헌

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
헨드리 프누
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
라마수브라모니안 아다르쉬 크리쉬난
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
왕 예-쿠이
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리어나

(뒷면에 계속)

전체 청구항 수 : 총 13 항

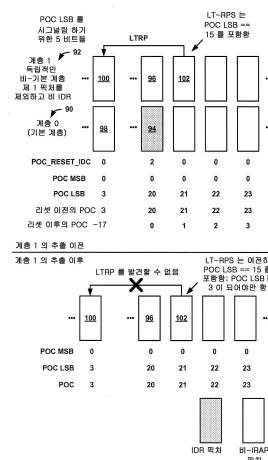
심사관 : 황수진

(54) 발명의 명칭 **멀티-계층 비디오 코딩에 대한 POC 값 설계**

(57) 요약

일 예로서, 비디오 데이터를 코딩하는 방법은, 비디오 코더에 의해, 멀티-계층 비트스트림의 독립적으로 디코딩 가능한 비-기본 계층의 픽처의 적어도 일부를 코딩하는 것을 포함한다. 그 방법은 또한, 오직 상기 픽처가 0과 동일한 상기 픽처에 대한 픽처 순서 카운트 (POC) 최하위 비트들 (LSB) 값을 가질 경우에만 상기 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩하는 단계를 포함한다.

대표도 - 도4



(52) CPC특허분류

H04N 19/187 (2015.01)
H04N 19/30 (2015.01)
H04N 19/42 (2015.01)
H04N 19/46 (2015.01)
H04N 19/70 (2015.01)

(56) 선행기술조사문헌

Adarsh K. Ramasubramanian et al, MV-HEVC/SHVC HLS: On picture order count, JCT-VC/JCT-3V of ITU-T and ISO/IEC, JCTVC-00213v4 (JCT3V-F0073v4) (2013.11.28.) 1부.*

Jianle Chen et al, High efficiency video coding (HEVC) scalable extensions Draft 5, JCT-VC/JCT-3V of ITU-T and ISO/IEC, JCTVC-P1008_v4(2014.01.22.) 1부.*

Miska M. Hannuksela et al, MV-HEVC/SHVC HLS: on POC value derivation, JCT-VC/JCT-3V of ITU-T and ISO/IEC, JCTVC-00275v6 (JCT3V-F0092v7) 1부.*

Tomohiro Ikai, On independent layer, JCT-VC/JCT-3V of ITU-T and ISO/IEC, JCTVC-00062_r2 (JCT3V-F0039_r2) (2013.10.30.) 1부.*

*는 심사관에 의하여 인용된 문헌

(30) 우선권주장

61/973,104	2014년03월31일	미국(US)
14/659,213	2015년03월16일	미국(US)

명세서

청구범위

청구항 1

비디오 데이터를 코딩하는 방법으로서,

멀티-계층 비트스트림의 독립적으로 디코딩가능한 비-기본 계층의 픽처의 적어도 일부를 코딩하는 단계;

상기 픽처의 슬라이스 헤더에서, 상기 픽처의 픽처 순서 카운트 (POC) 의 최하위 비트들의 값을 나타내는 픽처 순서 카운트 (POC) 최하위 비트 (LSB) 값을 코딩하는 단계;

슬라이스 헤더에서, 오직 상기 픽처가 0 과 동일한 상기 픽처에 대한 POC LSB 값을 가질 경우에만 상기 픽처의 상기 POC LSB 값에 대한 POC 값 리셋을 표시하는, POC 값 리셋 인덱스를 표시하는 신택스 엘리먼트를 코딩하는 단계; 및

상기 픽처가 0 과 동일하지 않은 POC LSB 값을 가질 경우, 2 이외의 값으로 상기 POC 값 리셋 인덱스 신택스 엘리먼트를 코딩하는 단계로서, 상기 값 2 는 현재의 픽처에 대한 POC 값의 최상위 비트들 (MSB 들) 및 LSB 들 양자 모두가 리셋될 수 있다는 것을 특징하는, 상기 POC 값 리셋 인덱스 신택스 엘리먼트를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 픽처의 픽처 타입이 순시적 디코더 리프레시 (IDR) 픽처, 절단된 링크 액세스 (BLA) 픽처, 또는 클린 랜덤 액세스 (CRA) 픽처 중 하나인 경우 POC 값 리셋 인덱스를 표시하는 상기 신택스 엘리먼트를 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 3

삭제

청구항 4

삭제

청구항 5

제 1 항에 있어서,

상기 하나 이상의 신택스 엘리먼트들을 코딩하는 단계는 POC 값 리셋 인덱스를 코딩하는 단계를 포함하며,

상기 픽처가 0 과 동일한 POC LSB 값을 가질 경우, 상기 POC 값 리셋 인덱스를 코딩하는 단계는 3 과 동일한 poc_reset_idc 신택스 엘리먼트를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 6

제 5 항에 있어서,

상기 POC LSB 값이 0 과 동일한 것을 표시하는 상기 픽처의 슬라이스 헤더의 poc_lsb_val 신택스 엘리먼트를 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 7

제 1 항에 있어서,

상기 독립적으로 디코딩가능한 비-기본 계층이 독립적으로 디코딩가능한 계층인 것을 표시하는 하나 이상의 신택스 엘리먼트들을 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 8

제 1 항에 있어서,

상기 멀티-계층 비트스트림은 HEVC 에 대한 멀티뷰 비디오 코딩 확장 (MV-HEVC) 비디오 코딩 표준 또는 HEVC 에 대한 스케일러블 비디오 코딩 (SVC) 확장 (SHVC) 비디오 코딩 표준 중 하나에 부합하고,

상기 독립적으로 디코딩가능한 비-기본 계층은 HEVC 비디오 코딩 표준에 부합하는, 비디오 데이터를 코딩하는 방법.

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

제 1 항에 있어서,

코딩하는 단계는 인코딩하는 단계를 포함하고,

상기 픽처의 적어도 일부를 인코딩하는 단계는,

예측 데이터와 상기 일부 간의 차이를 표시하는 상기 적어도 일부에 대한 잔차 데이터를 생성하는 단계;

상기 잔차 데이터에 변환을 적용하여 변환 계수들을 생성하는 단계; 및

상기 변환 계수들의 표시를 포함하는 비트스트림을 생성하는 단계

를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 13

제 1 항에 있어서,

코딩하는 단계는 디코딩하는 단계를 포함하고,

상기 픽처의 적어도 일부를 디코딩하는 단계는,

인코딩된 비트스트림으로부터 상기 적어도 일부에 대한 변환 계수들을 획득하는 단계;

상기 변환 계수들에 역변환을 적용하여 잔차 데이터를 생성하는 단계; 및

생성된 상기 잔차 데이터에 기초하여 상기 픽처의 상기 적어도 일부를 결정하는 단계

를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 14

비디오 데이터를 코딩하는 디바이스로서,

멀티-계층 비트스트림의 독립적으로 디코딩가능한 비-기본 계층의 픽처의 적어도 일부를 코딩하는 수단;

상기 픽처의 슬라이스 헤더에서, 상기 픽처의 픽처 순서 카운트 (POC) 의 최하위 비트들의 값을 나타내는 픽처 순서 카운트 (POC) 최하위 비트 (LSB) 값을 코딩하는 수단;

슬라이스 헤더에서, 오직 상기 픽처가 0 과 동일한 상기 픽처에 대한 POC LSB 값을 가질 경우에만 상기 픽처의 상기 POC LSB 값에 대한 POC 값 리셋을 표시하는, POC 값 리셋 인덱스를 표시하는 신택스 엘리먼트를 코딩하는

수단; 및

상기 픽처가 0 과 동일하지 않은 POC LSB 값을 가질 경우, 2 이외의 값으로 상기 POC 값 리셋 인덱스 신택스 엘리먼트를 코딩하는 수단으로서, 상기 값 2 는 현재의 픽처에 대한 POC 값의 최상위 비트들 (MSB 들) 및 LSB 들 양자 모두가 리셋될 수 있다는 것을 특징하는, 상기 POC 값 리셋 인덱스 신택스 엘리먼트를 코딩하는 수단을 포함하는, 비디오 데이터를 코딩하는 디바이스.

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

제 14 항에 있어서,

상기 멀티-계층 비트스트림은 HEVC 에 대한 멀티뷰 비디오 코딩 확장 (MV-HEVC) 비디오 코딩 표준 또는 HEVC 에 대한 스케일러블 비디오 코딩 (SVC) 확장 (SHVC) 비디오 코딩 표준 중 하나에 부합하고,

상기 독립적으로 디코딩가능한 비-기본 계층은 HEVC 비디오 코딩 표준에 부합하는, 비디오 데이터를 코딩하는 디바이스.

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

제 14 항에 있어서,

상기 디바이스는 비디오 인코더를 포함하고,

상기 비디오 인코더는 추가로,

예측 데이터와 상기 일부 간의 차이를 표시하는 상기 적어도 일부에 대한 잔차 데이터를 계산하고;

상기 잔차 데이터에 변환을 적용하여 변환 계수들을 생성하고; 그리고

상기 변환 계수들의 표시를 포함하는 비트스트림을 생성하도록
구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 26

제 14 항에 있어서,

상기 디바이스는 비디오 디코더를 포함하고,

상기 비디오 디코더는 추가로,

인코딩된 비트스트림으로부터 상기 적어도 일부에 대한 변환 계수들을 획득하고;

상기 변환 계수들에 역변환을 적용하여 잔차 데이터를 생성하고; 그리고

생성된 상기 잔차 데이터에 기초하여 상기 픽처의 상기 적어도 일부를 결정하도록

구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 27

삭제

청구항 28

삭제

청구항 29

명령들을 저장한 비-일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 실행될 경우, 비디오 데이터를 코딩하는 디바이스의 프로세서로 하여금, 제 1 항, 제 2 항, 제 5 항 내지 제 8 항, 제 12 항 및 제 13 항 중 어느 한 항의 방법을 수행하게 하는, 비-일시적 컴퓨터 판독가능 저장 매체.

발명의 설명

기술 분야

[0001] 본 출원은 2014 년 3 월 17 일자로 출원된 미국 가출원 제 61/954,551 호 및 2014 년 3 월 31 일자로 출원된 미국 특허 출원 제 61/973,104 호의 이점을 청구하며, 이 가출원들은 그들 전체가 참조에 의해 본원에 통합된다.

[0002] 본 개시물은 비디오 코딩에 관한 것이다.

배경 기술

[0003] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인용 디지털 보조기들 (PDA들), 랩탑 또는 데스크탑 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 소위 "스마트 폰들", 비디오 텔레컨퍼런싱 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, 어드밴스드 비디오 코딩 (AVC) 에 의해 정의된 표준들, ITU-T H.265 고 효율 비디오 코딩 (HEVC) 표준, 및 그러한 표준들의 확장들에서 설명된 기술들과 같은 비디오 코딩 기술들을 구현한다. 비디오 디바이스들은 그러한 비디오 코딩 기술들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0004] 비디오 코딩 기술들은 비디오 시퀀스들에 내재한 리던던시를 감소 또는 제거하기 위해 공간 (인트라-픽처) 예측 및/또는 시간 (인터-픽처) 예측을 포함한다. 블록 기반 비디오 코딩에 대해, 비디오 슬라이스 (예를 들어, 비디오 프레임 또는 비디오 프레임의 부분) 는 비디오 블록들로 파티셔닝될 수도 있으며, 이 비디오 블록들은 또한 트리블록들, 코딩 유닛들 (CU들) 및/또는 코딩 노드들로서 지칭될 수도 있다. 픽처의 인트라-코딩된

(I) 슬라이스에서의 비디오 블록들은 동일 픽처의 이웃 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 픽처의 인터-코딩된 (P 또는 B) 슬라이스에서의 비디오 블록들은 동일 픽처의 이웃 블록들에서의 참조 샘플들에 대한 공간 예측, 또는 다른 참조 픽처들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 픽처들은 프레임들로서 지칭될 수도 있으며, 참조 픽처들은 참조 프레임들로서 지칭될 수도 있다. 픽처는 디코딩된 픽처 또는 코딩된 픽처를 지칭할 수도 있다.

[0005] 공간 또는 시간 예측은 코딩된 블록에 대한 예측 블록을 발생한다. 잔차 데이터는 코딩된 원래의 블록과 예측 블록 간의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 포인팅하는 모션 벡터, 및 코딩된 블록과 예측 블록 간의 차이를 나타내는 잔차 데이터에 따라 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라 인코딩된다. 추가적인 압축을 위해, 잔차 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔차 변환 계수들을 발생시킬 수도 있으며, 그 후, 이 잔차 변환 계수들은 양자화될 수도 있다. 2차원 어레이로 초기에 배열되는 양자화된 변환 계수들은 변환 계수들의 1차원 벡터를 생성하기 위해 스캐닝될 수도 있으며, 엔트로피 코딩이 훨씬 더 많은 압축을 달성하도록 적용될 수도 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0006] 본 개시물은 멀티-계층 비트스트림으로부터 비디오 데이터의 계층의 추출을 용이하게 하기 위한 기술들을 설명한다. 예를 들어, 본 개시물의 기술들은 멀티-계층 비트스트림의 독립적인 비-기본 계층의 픽처들에 대한 픽처 순서 카운트 (POC) 값 리세팅을 제어하는데 사용될 수도 있다. 일부 예들에서, 비디오 코더에게 본 개시물의 기술들을 사용하도록 강요하는 것은, 멀티-계층 비트스트림의 비디오 데이터의 계층이 그 계층의 슬라이스 헤더들에서 데이터를 변경시키지 않고 추출되게 할 수도 있다. 추가로, POC 값 리세팅이 발생하는 로케이션들을 제어하는 것은, 비디오 코더가 디코딩된 픽처 버퍼 (DPB) 에서 장기 참조 픽처들을 적절히 식별할 수도 있도록 보장하는 것을 도울 수도 있다.

[0007] 일 예에서, 비디오 데이터를 코딩하는 방법은, 비디오 코더에 의해, 멀티-계층 비트스트림의 독립적으로 디코딩 가능한 비-기본 계층의 픽처의 적어도 일부분을 코딩하는 단계, 및 오직 그 픽처가 0 과 동일한 픽처에 대한 POC 최하위 비트들 (LSB) 값을 갖는 경우에만 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩하는 단계를 포함한다.

[0008] 다른 예에서, 비디오 데이터를 코딩하는 디바이스는, 멀티-계층 비트스트림의 비디오 데이터의 독립적으로 디코딩 가능한 계층을 저장하도록 구성된 메모리를 포함한다. 그 디바이스는 또한, 멀티-계층 비트스트림의 독립적으로 디코딩 가능한 비-기본 계층의 픽처의 적어도 일부분을 코딩하고, 그리고 오직 픽처가 0 과 동일한 픽처에 대한 픽처 순서 카운트 (POC) 최하위 비트들 (LSB) 값을 갖는 경우에만 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩하도록 구성된 비디오 코더를 포함한다.

[0009] 다른 예에서, 비디오 데이터를 코딩하는 디바이스는, 멀티-계층 비트스트림의 독립적으로 디코딩 가능한 비-기본 계층의 픽처의 적어도 일부분을 코딩하는 수단, 및 오직 픽처가 0 과 동일한 픽처에 대한 픽처 순서 카운트 (POC) 최하위 비트들 (LSB) 값을 갖는 경우에만 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩하는 수단을 포함한다.

[0010] 다른 예에서, 비-일시적 컴퓨터 판독가능 저장 매체는 저장된 명령들을 가지고, 그 명령들은, 실행될 경우, 비디오 데이터를 코딩하는 디바이스의 프로세서로 하여금, 멀티-계층 비트스트림의 독립적으로 디코딩 가능한 비-기본 계층의 픽처의 적어도 일부분을 코딩하게 하고, 그리고 오직 픽처가 0 과 동일한 픽처에 대한 픽처 순서 카운트 (POC) 최하위 비트들 (LSB) 값을 갖는 경우에만 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩하게 한다.

[0011] 하나 이상의 예들의 상세들이 첨부 도면들 및 하기의 설명에 개시된다. 다른 특징들, 목적들, 및 이점들은 그 설명 및 도면들로부터, 그리고 청구항들로부터 명백할 것이다.

도면의 간단한 설명

- [0012] 도 1 은 본 개시물의 기술들에 따라 POC 값 정보를 코딩하기 위한 기술들을 활용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 도시한 블록 다이어그램이다.
- 도 2 는 본 개시물의 기술들에 따라 POC 값 정보를 코딩하기 위한 기술들을 구현할 수도 있는 비디오 인코더의 일 예를 도시한 블록 다이어그램이다.
- 도 3 은 본 개시물의 기술들에 따라 POC 값 정보를 코딩하기 위한 기술들을 구현할 수도 있는 비디오 디코더의 일 예를 도시한 블록 다이어그램이다.
- 도 4 는 멀티-계층 비트스트림의 독립적인 비-기본 계층에 대한 POC 값 리세팅을 도시하는 개념 다이어그램이다.
- 도 5 는 멀티-계층 비트스트림의 독립적인 비-기본 계층에 대한 POC 값 리세팅을 도시하는 다른 개념 다이어그램이다.
- 도 6 은 본 개시물의 기술들에 따른 멀티-계층 비트스트림의 비디오 데이터를 인코딩하는 예시적인 방법을 도시한 플로우차트이다.
- 도 7 은 본 개시물의 기술들에 따른 멀티-계층 비트스트림의 비디오 데이터를 디코딩하는 예시적인 방법을 도시한 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

- [0013] 본 개시물의 양태들은 또한, 멀티-계층 비트스트림들에 대한 독립적인 비-기본 계층을 코딩하는 다양한 기술들에 관한 것일 수도 있다. 일부 경우들에 있어서, 그 기술들은 이하 논의되는 것과 같이, 고효율 비디오 코딩 (HEVC) 표준에 대한 멀티-계층 확장들, 예컨대 HEVC 에 대한 멀티뷰 비디오 코딩 확장 (MV-HEVC) 또는 HEVC 에 대한 스케일러블 비디오 코딩 (SVC) 확장 (SHVC) 로 수행될 수도 있다. 그러나, 본 개시물의 기술들은 임의의 특정 비디오 코딩 표준에 제한되는 것이 아니며, 또한 HEVC 에 대한 다른 확장들, 다른 멀티뷰 코딩 표준들 및/또는 다른 멀티-계층 비디오 표준들일 수도 있거나, 또는 대안적으로 이들과 함께 사용될 수도 있다. 추가로, 본 개시물의 기술들은, 이하 설명되는 것과 같이, 독립적으로 또는 조합하여 적용될 수도 있다.
- [0014] 비디오 데이터의 "계층" 은 일반적으로, 뷰, 프레임 레이트, 공간 포맷 또는 분해능, 신호대 잡음비, 등등과 같은 적어도 하나의 공통적인 특징을 갖는 픽처들의 시퀀스를 지칭할 수도 있다. 예를 들어, 계층은 멀티-뷰 비디오 데이터의 특정 뷰 (예컨대, 시각) 과 연관된 비디오 데이터를 포함할 수도 있다. 다른 예로서, 계층은 스케일러블 비디오 데이터의 특정 계층과 연관된 비디오 데이터를 포함할 수도 있다. 따라서, 본 개시물은 비디오 데이터의 계층과 뷰를 상호교환가능하게 지칭할 수도 있다. 즉, 비디오 데이터의 뷰는 비디오 데이터의 계층으로서 지칭될 수도 있고, 그 반대 또한 가능하다. 추가로, (멀티-계층 비디오 코더 또는 멀티-계층 인코더-디코더로도 지칭되는) 멀티-계층 코덱은 멀티-뷰 코덱 또는 스케일러블 코덱 (예컨대, MV-HEVC, SHVC, 또는 다른 멀티-계층 코딩 기술을 사용하여 비디오 데이터를 인코딩 및/또는 디코딩하도록 구성된 코덱) 을 지칭할 수도 있다.
- [0015] 멀티-계층 비트스트림은 기본 계층 및 하나 이상의 비-기본 계층들을 포함할 수도 있다. 기본 계층은 통상적으로 0 과 동일한 계층 식별자를 가질 수도 있다. 비-기본 계층은 0 보다 큰 계층 식별자를 가질 수도 있고, 기본 계층에 포함되지 않는 추가의 비디오 데이터를 제공할 수도 있다. 예를 들어, 멀티-뷰 비디오 데이터의 비-기본 계층은 비디오 데이터의 추가의 뷰를 포함할 수도 있다. 스케일러블 비디오 데이터의 비-기본 계층은 기본 계층에 대하여 증가되거나 감소된 분해능을 갖는 스케일러블 비디오 데이터의 추가 계층을 포함할 수도 있다. 비-기본 계층은 항상 계층으로 상호교환가능하게 지칭될 수도 있다.
- [0016] 계층은 그 계층이 직접 참조 계층을 가지지 않을 경우, "독립적인" 것으로 지칭될 수도 있다. 즉, 독립적인 계층은 다른 계층의 비디오 데이터를 지칭하지 않고, 코딩 (인코딩 또는 디코딩) 될 수도 있다. 독립적인 비-기본 계층은, 독립적인 계층이 속하는 멀티-계층 비트스트림의 다른 계층들과 동일하지 않아야만 하는, 특정 표준의 특정 코딩 프로파일에 부합할 수도 있다. 예시의 목적들을 위한 비-제한된 예에서, 멀티-계층 비트스트림은 MV-HEVC 또는 SHVC 에 부합할 수도 있다. 멀티-계층 비트스트림의 독립적인 비-기본 계층은 (예컨대, 계층간 예측 기술들을 사용하지 않고) 독립적으로 코딩될 수도 있고, HEVC 표준에 부합하지 않을 수도 있다. 따라서, 이하 더 상세히 설명되는 것과 같이, 독립적인 비-기본 계층은 HEVC 표준을 사용하고 멀티-

계층 비트스트림의 다른 계층들은 디코딩하지 않고 추출되고 디코딩될 수도 있다. 일부 경우들에서는, MANE (media-aware network element), 스플라이싱 (splicing) 유닛, 또는 다른 디바이스는 멀티-계층 비트스트림으로부터 독립적인 계층들을 추출할 수도 있다. 그러한 경우들에서, MANE 는 오직 추출된 독립적인 계층만을 디코딩을 위해 비디오 디코더로 송신할 수도 있다. 다른 경우들에서, 다른 컴퓨팅 디바이스는 서브-비트스트림의 추출을 담당할 수도 있다.

[0017] 본 개시물의 기술들은 비디오 코딩을 위한 픽처 순서 카운트 (POC) 값들의 세팅에 적용가능하고, 멀티-계층 비트스트림으로부터 독립적인 비-기본 계층의 추출을 용이하게 하는데 사용될 수도 있다. 예를 들어, 비디오 인코더는 특정 멀티-계층 비디오 코딩 표준, 예컨대 SHVC 또는 MV-HEVC 에 따라 멀티-계층 비트스트림을 인코딩할 수도 있다. 비디오 디코더, MANE (media-aware network element), 스플라이싱 (splicing) 유닛, 또는 다른 디바이스는 그러한 멀티-계층 비트스트림으로부터 독립적인 비-기본 계층들을 추출할 수도 있다. 이하 더 상세히 설명되는 것과 같이, 본 개시물의 기술들은, 일부 예들에서, 계층이 SHVC 또는 MV-HEVC 에 부합하는 비트스트림과 같은 멀티-계층 비트스트림으로부터 추출되게 하고, 독립적인 비-기본 계층의 다른 선택스 또는 슬라이스 헤더들을 변경하지 않고 HEVC 표준을 사용하여 디코딩하도록 구성된 비디오 디코더에 의해 적절히 디코딩되게 한다.

[0018] POC 값들은 픽처들의 디스플레이 순서를 표시할 수도 있고, 픽처를 식별하는데 일반적으로 사용될 수도 있다. 예를 들어, 0 의 POC 값을 갖는 픽처는 1 의 POC 값을 갖는 픽처 이전에 디스플레이된다. 현재 픽처의 블록이 참조 픽처에 대하여 인터-예측될 경우, 참조 픽처는 참조 픽처에 대한 POC 값을 사용하여 식별될 수도 있다. 더 구체적으로, 참조 픽처 리스트에 포함될 참조 픽처들에 대한 POC 값들은 시퀀스 파라미터 세트 (SPS), 픽처 파라미터 세트 (PPS), 및/또는 블록에 대응하는 슬라이스 헤더와 같은 파라미터 세트에서 시그널링될 수도 있다. 이러한 방식으로, 비디오 인코더는 인덱스를 참조 픽처 리스트에서의 참조 픽처의 위치에 대응하는 참조 픽처 리스트로 시그널링함으로써 참조 픽처를 식별할 수도 있고, 비디오 디코더는 (POC 값들에 기초하는) 참조 픽처 리스트를 구성함으로써 및 참조 픽처 리스트에서의 참조 픽처의 위치를 식별하기 위해 참조 인덱스를 사용함으로써 참조 픽처를 식별할 수도 있다.

[0019] POC 값들은 최하위 비트들 (LSB들) 및 최상위 비트들 (MSB들) 로 구성될 수도 있다. 일반적으로, POC LSB 값은 비트스트림에서 시그널링될 수도 있다. 예를 들어, POC LSB 값은 픽처의 슬라이스들의 슬라이스 헤더에 포함될 수도 있다. 비디오 코더 (비디오 인코더 또는 비디오 디코더) 는 비트스트림의 이전에 코딩된 픽처의 POC 값에 기초하여 픽처에 대한 POC MSB 값을 결정할 수도 있다.

[0020] POC 값들은 다중 계층들로부터의 비디오 데이터가 적절히 출력될 것을 보장하기 위해 멀티-계층 비디오 코딩에서 사용될 수도 있다. 예를 들어, 멀티-계층 비디오 코딩 (예컨대, 멀티-뷰 비디오 코딩, 스케일러블 비디오 코딩, 등등) 에서, 뷰 컴포넌트는 (디스플레이 순서 또는 출력 순서와 관련하여) 특정 시간에서의 특정 계층 또는 뷰에 대한 인코딩된 픽처를 포함할 수도 있다. 액세스 유닛은, 일반적으로, 공통의 시간 인스턴스에 대한 모든 뷰 컴포넌트들 (예컨대, 모든 네트워크 추상화 계층 (NAL) 유닛들) 을 포함하는 데이터의 유닛이다. 액세스 유닛의 뷰 컴포넌트들은 통상적으로, 함께 출력되도록 (즉, 실질적으로 동시에 출력되도록) 의도되며, 여기서 픽처를 출력하는 것은 일반적으로 DPB 로부터 픽처들을 전송하는 것 (예컨대, 픽처들을 DPB 로부터 외부 메모리로 저장하는 것, 픽처들을 DPB 로부터 디스플레이로 전송하는 것, DPB 로부터 픽처들을 제거하는 것, 등등) 을 수반한다

[0021] 비디오 코더는 POC 값을 주기적으로 리셋할 수도 있다. 예를 들어, 비디오 코더는 통상적으로 순시적 디코더 리프레시 (IDR) 픽처 (예컨대, 오직 인트라-예측된 슬라이스들만을 포함하는 픽처) 에 대한 POC 값을 리셋할 수도 있다. POC 값을 리셋하는데 부가하여, 비디오 코더는 POC 리셋과 연관된 현재 픽처 이전에 프로세싱되고 DPB 에 저장된 픽처들의 POC 값을 감분할 수도 있다. 감분된 참조 픽처들의 POC 값들은 그러한 픽처들의 적절한 출력 순서를 유지하기 위해 음의 값을 가질 수도 있다.

[0022] 일부 경우들에서, 비디오 코더는 액세스 유닛의 다른 픽처들과의 교차-계층 정렬을 유지하기 위해, 멀티-계층 비트스트림의 액세스 유닛에서 픽처의 POC 값을 리셋할 수도 있다. 단일 계층 비트스트림 (예컨대, HEVC) 에서, 각각의 액세스 유닛은 명시된 분류 규칙에 따라 서로 연관되고 디코딩 순서에 있어서 연속하는 NAL 유닛들의 세트를 포함하며, 각각의 액세스 유닛은 (또한 코딩된 픽처로 지칭되는) 정확히 하나의 픽처를 포함한다. 멀티-계층 비트스트림 (예컨대, 1 초과개의 계층을 갖는 SHVC 또는 MV-HEVC 비트스트림) 에서, (또한 멀티-계층 액세스 유닛들로 알려진) 액세스 유닛들은 하나 이상의 코딩된 픽처들을 포함할 수도 있다. 하나 이상의 코딩된 픽처들이 멀티-계층 액세스 유닛에 포함된다면, 각각의 코딩된 픽처는 서브-비트스트림 내에 배치된다.

예를 들면, 멀티-계층 액세스 유닛은 기본 계층에서 하나의 코딩된 픽처 및 향상 계층에서 하나의 코딩된 픽처를 가질 수도 있다. 따라서, 멀티-계층 액세스 유닛은 모든 연관된 비디오 코딩 계층 (VCL) 및 비-VCL NAL 유닛들을 포함하여, 동일한 출력 시간을 갖는 계층들에 걸친 모든 코딩된 픽처들을 포함한다. 액세스 유닛에서의 코딩된 픽처들은 식별을 위해 POC의 동일한 값과 그리고 액세스 유닛의 픽처들이 동시에 또는 실질적으로 동시에 출력되는 것을 보장하기 위해 연관될 수도 있다.

[0023] 교차-계층 정렬을 유지하기 위해 POC 값 리세팅에 대하여, 예시의 목적들을 위한 일 예에서, 액세스 유닛은 POC 값 리세팅이 수행되게 하는 기본 계층 IDR 픽처 및 IDR 픽처가 아닌 비-기본 계층 픽처를 포함할 수도 있다. 액세스 유닛의 비-기본 계층 픽처의 POC 값이 기본 계층 픽처의 POC 값에 매칭하는 것을 보장하기 위해, 비디오 코더는 비-기본 계층 픽처의 POC 값을 리셋할 수도 있다.

[0024] 전술된 것과 같이, 독립적인 비-기본 계층은, 비-기본 계층이 멀티-계층 비트스트림으로부터 추출되게 하기 위해 특정 표준과 연관된 특정 코딩 프로파일에 부합할 수도 있다. 그러나, 멀티-계층 비트스트림의 비-기본 계층이 독립적으로 코딩되도록 하기 위한 제공은 몇가지 도전과제들을 제시할 수도 있다. 예를 들어, 이하도 4에 대하여 더 상세히 설명되는 것과 같이, 독립적인 비-기본 계층의 픽처의 POC LSB 값을 리셋하는 것은 참조 픽처 세트 (RPS)의 장기 참조 픽처들 (LTRP들)을 적절히 식별해야 하는 문제를 야기할 수도 있다. 다른 예로서, 독립적인 비-기본 계층의 픽처의 POC LSB 값을 리셋하는 것은, 액세스 유닛이 교차 계층 정렬되지 않는 픽처들을 포함하게 할 수도 있다.

[0025] 본 개시물의 기술들은, 일부 예시들에서, 앞서 언급된 도전과제들을 어드레싱할 수도 있다. 예를 들어, 본 개시물의 기술들은 멀티-계층 비트스트림의 비-기본 계층이 추출되게 하고 (예컨대, 비-기본 계층의 슬라이스 헤더들 또는 다른 신택스를 변경하지 않고) 비-기본 계층에 최소한의 변경들로 디코딩되게 하는데 사용될 수도 있다. 본 개시물의 기술들은 독립적인 비-기본 계층에서 인트라 랜덤 액세스 포인트 (IRAP) 픽처들의 로케이션을 결정하는데 사용될 수도 있다. 이러한 방식으로, 본 개시물의 기술들은 독립적인 비-기본 계층의 픽처들에 대한 POC 값 리세팅을 제어하는데 사용될 수도 있다.

[0026] 본 개시물의 양태들에 따르면, 비디오 코더 (비디오 인코더 또는 비디오 디코더)는 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처를 코딩할 수도 있다. 비디오 코더는 또한, 오직 픽처에 대한 POC LSB 값이 0과 동일할 경우에만 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩할 수도 있다. 일부 예들에서, 비디오 코더는 코딩되고 있는 독립적인 비-기본 계층에서의 픽처가 IDR 픽처일 경우 또는 코딩되고 있는 픽처가 POC 값 리셋을 수행하기 전에 0과 동일한 POC LSB 값을 가질 경우, 오직 향상 계층에 대한 POC 값 리셋을 표시하는 데이터를 코딩할 수도 있다. 이하 더 상세히 설명되는 것과 같이, POC 값 리세팅이 발생하는 로케이션들을 제어하는 것은, 비디오 코더가 DPB에서 장기 참조 픽처들을 적절히 식별하고 예컨대, 본원에서 확인되는 표준과 같은 특정 비디오 코딩 표준과의 비트스트림 부합을 유지할 수도 있도록 보장하는 것을 도울 수도 있다.

[0027] 예시의 목적들을 위한 비-제한적인 예에서, 본 개시물의 기술들은 독립적인 비-기본 계층이 멀티-계층 스케일러블 비트스트림으로부터 추출되게 할 수도 있다. 예를 들어, SHVC 비트스트림은 540p의 분해능을 갖는 기본 계층 및 1080p의 향상 계층을 포함할 수도 있다. 관례상, 기본 계층에서, IDR 픽처는 32개 픽처들까지만 다 포함될 수도 있다. 향상 계층에서, IDR은 상이한 인터벌로, 예컨대 96개 픽처들마다 포함될 수도 있다.

[0028] 본 개시물의 기술들은 향상 계층이 기본 HEVC 표준과의 부합을 유지하도록, 그러한 스케일러블 비트스트림이 형성되는 방식을 제약하는데 사용될 수도 있다. 예를 들어, 본원에서 설명되는 것과 같이, 본 개시물의 기술들은 POC 값 리세팅이 SHVC 비트스트림에 대하여 수행되는 방식을 제어하여, 향상 계층이 SHVC 비트스트림으로부터 추출되고 기본 HEVC 표준에 부합하여 디코더에 의해 디코딩될 수도 있도록 하는 방식을 제어하는데 사용될 수도 있다. 따라서, (HEVC 비트스트림이 아닌) HEVC 비트스트림을 디코딩하는 능력을 갖는 모바일 컴퓨팅 디바이스는 향상 계층의 (본원에서 논의되는 것과 같은 POC 값 정보를 포함하는) 슬라이스 헤더들을 변경하지 않고 향상 계층을 수신하고 디코딩할 수도 있다.

[0029] 도 1은 본 개시물의 기술들에 따라 POC 값들을 관리하기 위한 기술들을 활용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10)을 도시한 블록 다이어그램이다. 도 1에 도시된 바와 같이, 시스템 (10)은, 목적지 디바이스 (14)에 의해 더 나중 시간에 디코딩될 인코딩된 비디오 데이터를 제공하는 소스 디바이스 (12)를 포함한다. 특히, 소스 디바이스 (12)는 비디오 데이터를, 컴퓨터 판독가능 매체 (16)를 통해 목적지 디바이스 (14)에 제공한다.

- [0030] 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋탑 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 경우들에 있어서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신을 위해 장비될 수도 있다.
- [0031] 컴퓨터 판독가능 매체 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동 가능한 임의의 타입의 매체 또는 디바이스를 포함할 수도 있다. 일 예에 있어서, 컴퓨터 판독가능 매체 (16) 는, 소스 디바이스 (12) 로 하여금 인코딩된 비디오 데이터를 직접 목적지 디바이스 (14) 로 실시간으로 송신할 수 있게 하는 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조되고, 목적지 디바이스 (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 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷 기반 네트워크의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.
- [0032] 일부 예들에서, 컴퓨터 판독가능한 매체 (16) 는 저장 디바이스를 포함할 수도 있다. 예를 들어, 저장 디바이스는 하드 드라이브, 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 휘발성 또는 비휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 국부적으로 액세스된 데이터 저장 매체들 중 임의의 데이터 저장 매체를 포함할 수도 있다. 추가의 예에 있어서, 저장 디바이스는, 소스 디바이스 (12) 에 의해 생성된 인코딩된 비디오를 저장할 수도 있고 목적지 디바이스에 의해 액세스될 수도 있는 파일 서버 또는 다른 중간 저장 디바이스에 대응할 수도 있다. 예시적인 파일 서버들은 웹 서버 (예를 들어, 웹 사이트용), FTP 서버, 네트워크 접속형 저장 (NAS) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다.
- [0033] 도 1 의 예에 있어서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20), 및 출력 인터페이스 (22) 를 포함한다. 목적지 디바이스 (14) 는 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 다른 예들에 있어서, 소스 디바이스 및 목적지 디바이스는 다른 컴포넌트들 또는 배열들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (12) 는 외부 카메라와 같은 외부 비디오 소스로부터 비디오 데이터를 수신할 수도 있다. 유사하게, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함하는 것보다 외부 디스플레이 디바이스와 인터페이스할 수도 있다. 따라서, 도 1 의 도시된 시스템 (10) 은 단지 일 예일 뿐이다. 본 개시물의 기술들에 따라 POC 값들을 관리하기 위한 기술들은 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스에 의해 수행될 수도 있다.
- [0034] 비록 일반적으로 본 개시의 기술들이 비디오 인코딩 디바이스에 의해 수행되지만, 그 기술들은 또한, 통상적으로 "CODEC" 로서 지칭되는 비디오 인코더/디코더에 의해 수행될 수도 있다. 더욱이, 본 개시의 기술들은 또한 비디오 프리프로세서에 의해 수행될 수도 있다. 소스 디바이스 (12) 및 목적지 디바이스 (14) 는, 단지, 소스 디바이스 (12) 가 목적지 디바이스 (14) 로의 송신을 위한 코딩된 비디오 데이터를 생성하는 그러한 코딩 디바이스들의 예들일 뿐이다. 일부 예들에 있어서, 디바이스들 (12, 14) 은, 디바이스들 (12, 14) 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 따라서, 시스템 (10) 은 예를 들어, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 전화를 위해, 비디오 디바이스들 (12, 14) 간의 일방향 또는 양방향 비디오 송신을 지원할 수도 있다.
- [0035] 소스 디바이스 (12) 의 비디오 소스 (18) 는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 및/또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스를 포함할 수도 있다. 추가적인 대안으로서, 비디오 소스 (18) 는 컴퓨터 그래픽 기반 데이터를 소스 비디오로서, 또는 라이브 비디오, 아카이브된 비디오, 및 컴퓨터 생성 비디오의 조합으로서 생성할 수도 있다. 일부 경우들에 있어서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 하지만, 상기 언급된 바와 같이, 본 개시에서 설명된 기술들은 일반적으로 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다. 각각의 경우에 있어서, 캡처되거나 사전-캡처되거나 또는 컴퓨터 생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 그 후, 인코딩된 비디오 정보는 출력 인터페이스 (22) 에

의해 컴퓨터 판독가능 매체 (16) 상으로 출력될 수도 있다.

[0036] 전술된 것과 같이, 컴퓨터 판독가능 매체 (16) 는 무선 브로드캐스트 또는 유선 네트워크 송신과 같은 일시적인 매체, 또는 저장 매체 (즉, 비-일시적인 저장 매체), 예컨대 하드 디스크, 플래시 드라이브, 콤팩트 디스크, 디지털 비디오 디스크, 블루레이 디스크, 또는 다른 컴퓨터 판독가능 매체를 포함할 수도 있다. 일부 예들에 있어서, 네트워크 서버 (도시 안됨) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 수신하고, 인코딩된 비디오 데이터를, 예를 들어, 네트워크 송신을 통해 목적지 디바이스 (14) 에 제공할 수도 있다. 유사하게, 디스크 스탬핑 설비와 같은 매체 생성 설비의 컴퓨팅 디바이스는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 수신하고, 인코딩된 비디오 데이터를 포함하는 디스크를 생성할 수도 있다. 따라서, 컴퓨터 판독가능 매체 (16) 는, 다양한 예들에 있어서, 다양한 형태들의 하나 이상의 컴퓨터 판독가능 매체들을 포함하도록 이해될 수도 있다.

[0037] 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 컴퓨터 판독가능 매체 (16) 로부터 정보를 수신하거나 액세스할 수도 있다. 컴퓨터 판독가능 매체 (16) 의 정보는 비디오 인코더 (20) 에 의해 정의되고 또한 비디오 디코더 (30) 에 의해 이용되는 신택스 정보를 포함할 수도 있으며, 이 신택스 정보는 블록들 및 다른 코딩된 유닛들, 예를 들어, 픽처들의 그룹 (GOP들) 의 특성 및/또는 프로세싱을 기술하는 신택스 엘리먼트들을 포함한다. 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 사용자에게 디스플레이하며, 음극선관 (CRT), 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0038] 목적지 디바이스 (14) 는 인코딩된 비디오 데이터에, 인터넷 접속을 포함한 임의의 표준 데이터 접속을 통해 액세스할 수도 있다. 이는 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하기에 적합한 무선 채널 (예를 들어, Wi-Fi 접속), 유선 접속 (예를 들어, DSL, 케이블 모뎀 등), 또는 이들 양자의 조합을 포함할 수도 있다. 인코딩된 비디오 데이터의 저장 디바이스로부터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이들의 조합일 수도 있다. 본 개시의 기술들은 무선 애플리케이션들 또는 설정들로 반드시 한정되는 것은 아니다. 그 기술들은, 공중 경유 (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, HTTP 상으로의 동적 적응 스트리밍 (DASH) 과 같은 인터넷 스트리밍 비디오 송신들, 데이터 저장 매체 상으로 인코딩된 디지털 비디오, 데이터 저장 매체 상으로 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은 다양한 멀티미디어 애플리케이션들 중 임의의 멀티미디어 애플리케이션의 지원으로 비디오 코딩에 적용될 수도 있다. 일부 예들에 있어서, 시스템 (10) 은 비디오 스트리밍, 비디오 플레이어백, 비디오 브로드캐스팅, 및/또는 비디오 전화와 같은 애플리케이션들을 지원하기 위해 일방향 또는 양방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0039] 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 이들의 임의의 조합들과 같은 다양한 적합한 인코더 회로 중 임의의 회로로서 구현될 수도 있다. 기술들이 부분적으로 소프트웨어로 구현될 경우, 디바이스는 적합한 비-일시적인 컴퓨터 판독가능 매체에 소프트웨어에 대한 명령들을 저장하고, 본 개시의 기술들을 수행하기 위해 하나 이상의 프로세서들을 사용하는 하드웨어로 명령들을 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 하나는 개별 디바이스에 있어서 결합된 CODEC 의 부분으로서 통합될 수도 있다. 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 를 포함하는 디바이스는 집적 회로, 마이크로 프로세서, 및/또는 셀룰러 전화기와 같은 무선 통신 디바이스를 포함할 수도 있다.

[0040] 비록 도 1 에 도시되지는 않지만, 일부 양태들에 있어서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양자의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기 위해 적절한 MUX-DEMUX 유닛들 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, MUX-DEMUX 유닛들은 ITU H.223 다중화기 프로토콜, 또는 사용자 데이터그램 프로토콜 (UDP) 과 같은 다른 프로토콜에 따를 수도 있다.

[0041] 본 개시물은 일반적으로 특정 정보를, 비디오 디코더 (30) 와 같은 다른 디바이스로 "시그널링"하는 비디오 인코더 (20) 를 언급할 수도 있다. 하지만, 비디오 인코더 (20) 는, 특정 신택스 엘리먼트들을 비디오 데이터의 다양한 인코딩된 부분들과 연관시킴으로써 정보를 시그널링할 수도 있음이 이해되어야 한다. 즉, 비디오 인코더 (20) 는, 특정 신택스 엘리먼트들을 비디오 데이터의 다양한 인코딩된 부분들의 헤더들에 저장함으로써

데이터를 "시그널링"할 수도 있다. 일부 경우들에 있어서, 그러한 선택스 엘리먼트들은, 비디오 디코더 (30) 에 의해 수신 및 디코딩되기 전에 인코딩 및 저장될 수도 있다. 따라서, 용어 "시그널링" 은 일반적으로, 선택스 엘리먼트들을 인코딩 시 매체에 저장할 경우 (그 후, 이 매체에 저장된 이후 임의의 시간에서 디코딩 디바이스에 의해 추출될 수도 있음) 에 발생할 수도 있는 것과 같이 그러한 통신이 실시간으로 또는 준 실시간으로 또는 시간 기간에 걸쳐 발생하든지 아니든지, 압축된 비디오 데이터를 디코딩하기 위한 선택스 또는 다른 데이터의 통신을 지칭할 수도 있다.

[0042] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 비디오 코딩 표준에 따라 동작할 수도 있다. 최근에, 새로운 비디오 코딩 표준, 즉, 고효율 비디오 코딩 (HEVC) 의 설계는 ITU-T 비디오 코딩 전문가 그룹 (VCEG) 및 ISO/IEC 모션 픽처 전문가 그룹 (MPEG) 의 JCT-VC (Joint Collaboration Team on Video Coding) 에 의해 완결되었다. HEVC 에 대한 멀티뷰 확장, 즉 MV-HEVC 가 또한 JCT-3V 에 의해 개발되고 있다. MV-HEVC 의 최신 워킹 드래프트 (WD) 는 MV-HEVC WD7 로 지칭된다. SHVC 로 명명된 HEVC 에 대한 스케일러블 확장물이 또한 JCT-VC 에 의해 개발되고 있다. SHVC 의 최신 워킹 드래프트 (WD) 는 SHVC WD5 로 지칭된다. 하지만, 본 개시의 기술들은 임의의 특정 코딩 표준으로 한정되지 않는다.

[0043] HEVC 및 다른 비디오 코딩 표준들에 있어서, 비디오 시퀀스는 통상적으로 픽처들의 시리즈를 포함한다. 픽처들은 또한 "프레임들"로서 지칭될 수도 있다. 픽처는 S_L , S_{Cb} , 및 S_{Cr} 로 표기되는 3 개의 샘플 어레이들을 포함할 수도 있다. S_L 은 루마 샘플들의 2 차원 어레이 (즉, 블록) 이다. S_{Cb} 는 Cb 크로미넌스 샘플들의 2 차원 어레이이다. S_{Cr} 는 Cr 크로미넌스 샘플들의 2 차원 어레이이다. 크로미넌스 샘플들은 또한, 본 명세서에서 "크로마" 샘플들로서 지칭될 수도 있다. 다른 경우들에 있어서, 픽처는 단색일 수도 있으며 오직 루마 샘플들의 어레이만을 포함할 수도 있다.

[0044] 픽처의 인코딩된 표현을 생성하기 위해, 비디오 인코더 (20) 는 코딩 트리 블록들 (CTU들) 의 세트를 생성할 수도 있다. CTU들 각각은 루마 샘플들의 코딩 트리 블록, 크로마 샘플들의 2 개의 대응하는 코딩 트리 블록들, 및 코딩 트리 블록들의 샘플들을 코딩하는데 사용된 선택스 구조들일 수도 있다. 코딩 트리 블록은 샘플들의 $N \times N$ 블록일 수도 있다. CTU 는 또한, "트리 블록" 또는 "최대 코딩 유닛" (LCU) 으로서 지칭될 수도 있다. HEVC 의 CTU들은 H.264/AVC 와 같은 다른 표준들의 매크로블록들과 대체로 유사할 수도 있다. 하지만, CTU 는 반드시 특정 사이즈로 한정되지는 않으며, 하나 이상의 코딩 유닛들 (CU들) 을 포함할 수도 있다.

[0045] 슬라이스는 래스터 스캔에 있어서 연속적으로 순서화된 정수 개수의 CTU들을 포함할 수도 있다. 슬라이스는 슬라이스 헤더 및 슬라이스 데이터를 포함할 수도 있다. 슬라이스의 슬라이스 헤더는 슬라이스에 관한 정보를 제공하는 선택스 엘리먼트들을 포함하는 선택스 구조일 수도 있다. 슬라이스 데이터는 슬라이스의 코딩된 CTU들을 포함할 수도 있다.

[0046] 본 개시물은 용어 "비디오 유닛" 또는 "비디오 블록" 또는 "블록" 을 사용하여, 샘플들의 하나 이상의 블록들의 샘플들을 코딩하는데 사용된 선택스 구조들 및 하나 이상의 샘플 블록들을 지칭할 수도 있다. 비디오 유닛들 또는 블록들의 예시적인 타입들은, CTU들, CU들, PU들, 변환 유닛들 (TU들), 매크로블록들, 매크로블록 파티션들, 등등을 포함할 수도 있다. 일부 맥락들에서, PU들이 논의는 매크로블록 파티션들의 매크로블록들의 논의와 상호교환될 수도 있다.

[0047] 코딩된 CTU 를 생성하기 위해, 비디오 인코더 (20) 는 CTU 의 코딩 트리 블록들에 쿼드 트리 파티셔닝을 재귀적으로 수행하여, 코딩 트리 블록들을 코딩 블록들, 따라서, 명칭 "코딩 트리 유닛들" 로 분할할 수도 있다. 코딩 블록은 샘플들의 $N \times N$ 블록이다. CU 는 루마 샘플들의 코딩 블록, 및 루마 샘플 어레이, Cb 샘플 어레이, 및 Cr 샘플 어레이를 갖는 픽처의 크로마 샘플들의 2개의 대응하는 코딩 블록들, 그리고 코딩 블록들의 샘플들을 코딩하는데 사용된 선택스 구조들일 수도 있다. 비디오 인코더 (20) 는 CU 의 코딩 블록을 하나 이상의 예측 블록들로 파티셔닝할 수도 있다. 예측 블록은, 동일한 예측이 적용되는 샘플들의 직사각형 (즉, 정사각형 또는 비-정사각형) 블록이다. CU 의 예측 유닛 (PU) 은 루마 샘플들의 예측 블록, 픽처의 크로마 샘플들의 2 개의 대응하는 예측 블록들, 및 예측 블록 샘플들을 예측하는데 사용된 선택스 구조들일 수도 있다. 비디오 인코더 (20) 는 CU 의 각각의 PU 의 루마, Cb 및 Cr 예측 블록들에 대한 예측 루마, Cb, 및 Cr 블록들을 생성할 수도 있다.

[0048] 비디오 인코더 (20) 는 인트라-예측 또는 인터-예측을 이용하여 PU 에 대한 예측 블록들을 생성할 수도 있다. 비디오 인코더 (20) 가 인트라-예측을 이용하여 PU 의 예측 블록들을 생성하면, 비디오 인코더 (20) 는 PU

와 연관된 픽처의 디코딩된 샘플들에 기초하여 PU 의 예측 블록들을 생성할 수도 있다.

[0049] 비디오 인코더 (20) 가 인터-예측을 이용하여 PU 의 예측 블록들을 생성하면, 비디오 인코더 (20) 는 PU 와 연관된 픽처 이외의 하나 이상의 픽처들의 디코딩된 샘플들에 기초하여 PU 의 예측 블록들을 생성할 수도 있다.

비디오 인코더 (20) 는 PU 의 예측 블록들을 생성하기 위해 단방향 예측 또는 양방향 예측을 사용할 수도 있다. 비디오 인코더 (20) 가 PU 에 대한 예측 블록들을 생성하기 위해 단방향 예측을 사용할 경우, PU 는 단일 모션 벡터 (MV) 를 가질 수도 있다. 비디오 인코더 (20) 가 PU 에 대한 예측 블록들을 생성하기 위해 양방향 예측을 사용할 경우, PU 는 2 개의 모션 벡터들 (MV) 을 가질 수도 있다.

[0050] 비디오 인코더 (20) 가 CU 의 하나 이상의 PU들에 대한 예측 루마, Cb, 및 Cr 블록들을 생성한 후, 비디오 인코더 (20) 는 CU 에 대한 루마 잔차 블록을 생성할 수도 있다. CU 의 루마 잔차 블록에서의 각각의 샘플은 CU 의 예측 루마 블록들 중 하나에서의 루마 샘플과 CU 의 원래의 루마 코딩 블록에서의 대응하는 샘플 간의 차이를 나타낸다. 부가적으로, 비디오 인코더 (20) 는 CU 에 대한 Cb 잔차 블록을 생성할 수도 있다. CU 의 Cb 잔차 블록에서의 각각의 샘플은 CU 의 예측 Cb 블록들 중 하나에서의 Cb 샘플과 CU 의 원래의 Cb 코딩 블록에서의 대응하는 샘플 간의 차이를 나타낼 수도 있다. 비디오 인코더 (20) 는 또한, CU 에 대한 Cr 잔차 블록을 생성할 수도 있다. CU 의 Cr 잔차 블록에서의 각각의 샘플은 CU 의 예측 Cr 블록들 중 하나에서의 Cr 샘플과 CU 의 원래의 Cr 코딩 블록에서의 대응하는 샘플 간의 차이를 나타낼 수도 있다.

[0051] 더욱이, 비디오 인코더 (20) 는 쿼드 트리 파티셔닝을 이용하여, CU 의 루마, Cb, 및 Cr 잔차 블록들을 하나 이상의 루마, Cb, 및 Cr 변환 블록들로 분해할 수도 있다. 변환 블록은, 동일한 변환이 적용되는 샘플들의 직사각형 블록일 수도 있다. CU 의 변환 유닛 (TU) 은 루마 샘플들의 변환 블록, 크로마 샘플들의 2 개의 대응하는 변환 블록들, 및 변환 블록 샘플들을 변환하는데 사용된 선택스 구조들일 수도 있다. 따라서, CU 의 각각의 TU 는 루마 변환 블록, Cb 변환 블록, 및 Cr 변환 블록과 연관될 수도 있다. TU 와 연관된 루마 변환 블록은 CU 의 루마 잔차 블록의 서브-블록일 수도 있다. Cb 변환 블록은 CU 의 Cb 잔차 블록의 서브-블록일 수도 있다. Cr 변환 블록은 CU 의 Cr 잔차 블록의 서브-블록일 수도 있다.

[0052] 비디오 인코더 (20) 는 TU 의 루마 변환 블록에 하나 이상의 변환들을 적용하여 TU 에 대한 루마 계수 블록을 생성할 수도 있다. 계수 블록은 변환 계수들의 2 차원 어레이일 수도 있다. 변환 계수는 스칼라량일 수도 있다. 비디오 인코더 (20) 는 TU 의 Cb 변환 블록에 하나 이상의 변환들을 적용하여 TU 에 대한 Cb 계수 블록을 생성할 수도 있다. 비디오 인코더 (20) 는 TU 의 Cr 변환 블록에 하나 이상의 변환들을 적용하여 TU 에 대한 Cr 계수 블록을 생성할 수도 있다.

[0053] 계수 블록 (예를 들어, 루마 계수 블록, Cb 계수 블록 또는 Cr 계수 블록) 을 생성한 후, 비디오 인코더 (20) 는 계수 블록을 양자화할 수도 있다. 양자화는 일반적으로, 변환 계수들이 그 변환 계수들을 나타내는데 사용되는 데이터의 양을 가능하게 감소시키도록 양자화되어 추가 압축을 제공하는 프로세스를 지칭한다. 비디오 인코더 (20) 가 계수 블록을 양자화한 후, 비디오 인코더 (20) 는 양자화된 변환 계수들을 나타내는 선택스 엘리먼트들을 엔트로피 인코딩할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 양자화된 변환 계수들을 나타내는 선택스 엘리먼트들에 컨텍스트-적응형 이진 산술 코딩 (CABAC) 을 수행할 수도 있다. 비디오 인코더 (20) 는 비트스트림에서 엔트로피 인코딩된 선택스 엘리먼트들을 출력할 수도 있다.

[0054] 비디오 인코더 (20) 는 단일 계층 또는 멀티-계층 비트스트림을 출력할 수도 있다. 각각의 계층은 엔트로피 인코딩된 선택스 엘리먼트들을 포함하는 연관된 비트스트림을 갖는다. 비트스트림은 코딩된 픽처들 및 연관된 데이터의 표현을 형성하는 비트들의 시퀀스를 포함할 수도 있다. 비트스트림은 네트워크 추상화 계층 (NAL) 유닛들의 시퀀스를 포함할 수도 있다. NAL 유닛들 각각은 NAL 유닛 헤더를 포함하고 원시 바이트 시퀀스 페이로드 (RBSP) 를 캡슐화한다. NAL 유닛 헤더는, NAL 유닛 타입 코드를 나타내는 선택스 엘리먼트를 포함할 수도 있다. NAL 유닛의 NAL 유닛 헤더에 의해 명시된 NAL 유닛 타입 코드는 NAL 유닛의 타입을 나타낸다. RBSP 는, NAL 유닛 내에서 캡슐화되는 정수 개수의 바이트들을 포함하는 선택스 구조일 수도 있다. 일부 경우들에 있어서, RBSP 는 제로 비트들을 포함한다.

[0055] 상이한 타입들의 NAL 유닛들은 상이한 타입들의 RBSP들을 캡슐화할 수도 있다. 예를 들어, NAL 유닛의 제 1 타입은 픽처 파라미터 세트 (PPS) 에 대한 RBSP 를 캡슐화할 수도 있고, NAL 유닛의 제 2 타입은 코딩된 슬라이스에 대한 RBSP 를 캡슐화할 수도 있으며, NAL 유닛의 제 3 타입은 보충 향상 정보 (SEI) 에 대한 RBSP 를 캡슐화할 수도 있는 등등이다. (파라미터 세트들 및 SEI 메시지들에 대한 RBSP들과 대조적으로) 비디오 코딩 데이터에 대한 RBSP들을 캡슐화하는 NAL 유닛들은 VCL NAL 유닛들로서 지칭될 수도 있다.

- [0056] 비디오 디코더 (30) 는 비디오 인코더 (20) 에 의해 생성된 비트스트림을 수신할 수도 있다. 부가적으로, 비디오 디코더 (30) 는 비트스트림을 파싱하여, 비트스트림으로부터 신택스 엘리먼트들을 디코딩할 수도 있다. 비디오 디코더 (30) 는 비트스트림으로부터 디코딩된 신택스 엘리먼트들에 적어도 부분적으로 기초하여 비디오 데이터의 픽처들을 복원할 수도 있다. 비디오 데이터를 복원하기 위한 프로세스는 일반적으로, 비디오 인코더 (20) 에 의해 수행된 프로세스에 역일 수도 있다.
- [0057] 예를 들어, 비디오 디코더 (30) 는 PU들의 모션 벡터들 (MV들) 을 이용하여 현재 CU 의 PU들에 대한 예측 블록들을 결정할 수도 있다. 부가적으로, 비디오 디코더 (30) 는 현재 CU 의 TU들과 연관된 변환 계수 블록들을 역양자화할 수도 있다. 비디오 디코더 (30) 는 변환 계수 블록들에 역변환들을 수행하여 현재 CU 의 TU들과 연관된 변환 블록들을 복원할 수도 있다. 비디오 디코더 (30) 는 현재 CU 의 PU들에 대한 예측 샘플 블록들의 샘플들을, 현재 CU 의 TU들의 변환 블록들의 대응하는 샘플들에 부가함으로써, 현재 CU 의 코딩 블록들을 복원할 수도 있다. 픽처의 각각의 CU 에 대한 코딩 블록들을 복원함으로써, 비디오 디코더 (30) 는 픽처를 복원할 수도 있다.
- [0058] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 픽처의 식별을 위해 POC 값을 비디오 시퀀스의 각각의 픽처에 할당할 수도 있다. 비디오 인코더 (20) 는, 예컨대 슬라이스 헤더 및/또는 PPS 또는 SPS 와 같은 파라미터 세트에서 참조 픽처들에 대한 POC 값들을 나타내는 데이터를 시그널링함으로써, 참조 픽처 리스트에 포함될 참조 픽처들을 나타내는 데이터를 인코딩할 수도 있다 (그리고 비디오 디코더 (30) 는 디코딩할 수도 있다). 특정 코딩된 픽처에 대하여 신택스 엘리먼트 PicOrderCntVal 로 표현되는 POC 값은, 동일한 코딩된 비디오 시퀀스에서의 다른 픽처들에 대한 픽처 출력 프로세스에서 픽처의 상대적인 순서를 표시한다. 감분된 참조 픽처들의 POC 값들은 그러한 픽처들의 적절한 출력 순서를 유지하기 위해 음의 값을 가질 수도 있다.
- [0059] POC 값은 POC LSB 값 및 POC MSB 값을 포함한다. POC LSB 값은 각각의 픽처에 대하여 증가할 수도 있고, 0 의 초기 값으로부터 최대 값 (예컨대, 4 내지 16 비트들을 사용하여 명시된 최대 값) 으로의 출력 순서를 명시한다. 8 비트 POC LSB 값의 예에서, 예시의 목적들을 위해, 255 의 POC LSB 값 이후에, POC LSB 값은 255 부터 0 까지 떨어질 수도 있고, POC LSB 값은 0 부터 증가하는 것을 계속할 수도 있다. POC LSB 값은 슬라이스 헤더에서 시그널링될 수도 있는 반면, POC MSB 값은 현재 픽처의 NAL 유닛 타입 및 POC-앵커 픽처들로 지칭되는 디코딩 순서에서의 이전 픽처의 MSB 및 LSB 에 기초하여 비디오 디코더 (30) 에 의해 계산될 수도 있다. SHVC 및/또는 MV-HEVC 에 대하여, 추가의 정보가 POC MSB 의 계산을 보조하기 위해 시그널링될 수도 있다. 그러한 예들에서, 추가의 정보가 존재할 경우, POC MSB 의 계산은 디코딩 순서에 있어서 이전 픽처에 의존하지 않는다.
- [0060] 멀티-계층 비트스트림이 포함할 수도 있는 액세스 유닛은 IRAP 픽처뿐만 아니라, IRAP 픽처들이 아닌 하나 이상의 다른 픽처들 (즉, 비-IRAP 픽처들) 이다. IRAP 픽처는 오직 인트라-코딩된 (I) 슬라이스들을 포함하고, 절단된 링크 액세스 (BLA) 픽처, 클린 랜덤 액세스 (CRA) 픽처 또는 IDR 픽처일 수도 있다. 각각의 계층에 대하여, 코딩된 비디오 시퀀스에 대한 비트스트림에서의 제 1 픽처는 IRAP 픽처이다. IRAP 픽처들은 또한, 멀티-계층 비트스트림의 각 계층에서의 다양한 포인트들에서 제공된다. 멀티-계층 액세스 유닛은, IRAP 픽처들이 각 계층 내에서 동일한 인터벌들로 배치되지 않을 수도 있기 때문에, IRAP 픽처들 및 비-IRAP 픽처들을 포함할 수도 있다. IRAP 픽처들이 통상적으로 인트라-코딩된 비-IRAP 픽처들보다 더 큰 비트 페이로드를 요구하기 때문에, 비디오 인코더 (20) 는 기본 계층과 비교할 때, 향상 계층에서 더 큰 인터벌들로 IRAP 픽처들을 배치할 수도 있다.
- [0061] 예를 들어, 8 의 GOP 에 대하여, 비디오 인코더 (20) 는 기본 계층에서 매 32 개 픽처들마다 IRAP 픽처들을 배치할 수도 있고, 향상 계층에서 매 96 개 픽처들마다 IRAP 픽처들을 배치할 수도 있다. (기본 계층과 같은) 하나의 계층이 IRAP 픽처를 가지고 다른 계층이 IRAP 픽처를 가지지 않는 멀티-계층 비트스트림에서의 지점에서, 멀티-계층 액세스 유닛은 비-정렬된 액세스 유닛으로 지칭될 수도 있다. 후속하는 멀티-계층 액세스 유닛들은, POC 가 리셋될 때까지, 비-정렬되는 것을 계속한다. 이하 더 상세히 설명되는 것과 같이, 멀티-계층 액세스 유닛의 모든 계층들에 걸친 IRAP 픽처의 제공은 정렬을 리셋할 수도 있다.
- [0062] 일부 경우들에서, POC 값들은 멀티-계층 액세스 유닛에서 픽처들의 POC 정렬을 유지하기 위해, 멀티-계층 액세스 유닛의 픽처들에 대하여 리셋될 수도 있다. 예를 들어, 표준 부합 HEVC 디코더는 추출된 계층에서 픽처들의 슬라이스 헤더들에서 참조 픽처 세트 (RPS) 정보를 업데이트하지 않고, 멀티-계층 비트스트림의 (향상 계층으로 또한 알려진) 비-기본 계층을 추출하고 디코딩하도록 구성될 수도 있다. 향상 계층은, 예컨대 MV-HEVC 비트스트림에서의 대안적인 뷰 또는 SHVC 비트스트림에서의 스케일링된 비디오일 수도 있다. 계층-특

정 POC 리세팅 주기는 슬라이스 세그먼트 헤더들에서 시그널링된 POC 리세팅 주기 식별자에 기초하여 명시될 수도 있다. 즉, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 슬라이스 세그먼트 헤더들에서 POC 리세팅 주기 식별자를 나타내는 데이터를 개별적으로 코딩할 수도 있다.

[0063] 적어도 하나의 IRAP 픽처를 포함하는 액세스 유닛에 속하는 (IDR 픽처 타입의 픽처들을 포함하는) 각각의 비-IRAP 픽처는 비-IRAP 픽처를 포함하는 계층에서 시작될 수도 있는 POC 리세팅 주기의 시작부일 수도 있다. 즉, 비디오 인코더 (20) 는 비-IRAP 픽처가 새로운 POC 리세팅 주기의 시작부인 것을 표시하기 위해, 적어도 하나의 IRAP 픽처를 포함하는 액세스 유닛의 비-IRAP 픽처에 대한 POC 리세팅 타입을 세팅할 수도 있다. 그러한 액세스 유닛에서, 각각의 픽처는 그 후, 계층들의 각각에서 POC 리세팅 주기의 시작부일 것이다.

[0064] 비디오 인코더 (20) 는, 비디오 디코더 (30) 가 슬라이스 세그먼트 헤더를 갖는 슬라이스를 포함하는 계층에서 픽처들의 POC 값들 도출하기 위해 사용할 수도 있는, 슬라이스 세그먼트 헤더에서의 POC LSB 값을 시그널링할 수도 있다. 슬라이스 세그먼트 헤더에서 시그널링된 POC LSB 값은 델타 POC 값의 도출을 위해 사용될 수도 있고, 이는 DPB 에서 동일-계층 픽처들의 POC 값들을 업데이트하기 위해 및 또한 현재 픽처의 POC 값의 POC MSB 의 도출을 위해 사용될 수도 있다.

[0065] POC 리세팅 주기 동안 POC 리세팅을 수행할 때, 비디오 인코더 (20) 는 슬라이스 세그먼트 헤더에서 POC 리세팅 타입을 시그널링할 수도 있다. POC 리세팅 타입은, POC 값들이 완전히 리셋될 것인지 또는 부분적으로 리셋될 것인지를 여부를 표시할 수도 있는, POC 값 리셋 인덱스 (poc_reset_idc) 를 사용하여 표시될 수도 있다. 예를 들어, MV-HEVC 또는 SHVC 에서, POC 리세팅 타입은 poc_reset_idc 인덱스 엘리먼트를 사용하여 표시될 수도 있다. 그러한 예에서, 0 과 동일한 poc_reset_idc 인덱스는 현재 픽처에 대한 POC 값의 MSB들 및 LSB들이 리셋되지 않는 것을 명시한다. 1 과 동일한 poc_reset_idc 인덱스 엘리먼트는 오직 현재 픽처에 대한 POC 값의 MSB들만이 리셋될 수도 있는 것을 명시한다. 2 와 동일한 poc_reset_idc 인덱스 엘리먼트는 현재 픽처에 대한 POC 값의 MSB들 및 LSB들 양자가 리셋될 수도 있는 것을 명시한다. 3 과 동일한 poc_reset_idc 인덱스 엘리먼트는 현재 픽처에 대한 POC 값의 오직 MSB들 또는 MSB들 및 LSB들 양자가 리셋될 수도 있는 것을 명시하고, 추가의 POC 정보가 시그널링된다.

[0066] 독립적인 비-기본 계층이 독립적으로 디코딩가능하도록 하기 위해, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 비트스트림을 형성하고 디코딩할 때, 특정 제한들을 적용할 수도 있다. 예시의 목적들을 위한 비-제한적인 예에서, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 MV-HEVC 또는 SHVC 를 사용하여 멀티-계층 비트스트림을 코딩할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 독립적인 비-기본 계층을 코딩할 때 하나 이상의 제약들을 적용하여 비-기본 계층이 추출되게 하고 HEVC 부합 디코더를 사용하여 디코딩되게 할 수도 있다. 예를 들어, 독립적인 비-기본 계층이 HEVC 부합 디코더에 의해 성공적으로 디코딩되도록 하기 위해, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 HEVC 에 부합하는 독립적인 비-기본 계층에 대한 인덱스를 사용할 수도 있다. 예를 들어, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 HEVC 표준에 부합하도록 슬라이스 헤더들 및 독립적인 비-기본 계층의 다른 인덱스를 코딩할 수도 있다. 이러한 방식으로, HEVC 부합 비디오 디코더는 슬라이스 헤더들의 데이터를 수정하거나 다른 트랜스코딩 동작들을 수행하지 않고 독립적인 비-기본 계층을 디코딩할 수도 있다.

[0067] 일부 경우들에 있어서, 독립적인 비-기본 계층의 픽처의 POC LSB 값을 리셋하는 것은 RPS 에서 장기 참조 픽처 (LTRP) 를 식별하는 것을 시도할 때, 문제를 야기할 수도 있다. 예를 들어, 전송된 것과 같이, 비디오 인코더 (20) 는, 예컨대 슬라이스 헤더 및/또는 PPS 또는 SPS 와 같은 파라미터 세트에서 참조 픽처들에 대한 POC 값들을 나타내는 데이터를 시그널링함으로써, 참조 픽처 리스트에 포함될 참조 픽처들을 나타내는 데이터를 인코딩할 수도 있다 (그리고 비디오 디코더 (30) 는 디코딩할 수도 있다). 추가로, 참조 픽처 리스트에 포함될 참조 픽처들을 나타내는 데이터를 포함시키기 전에, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 POC 리셋과 연관된 현재 픽처 이전에 프로세싱되고 DPB 에 저장된 픽처들의 POC 값들을 감분할 수도 있다.

[0068] LTRP들은 개별 LTRP들의 POC LSB 값에 기초하여 식별될 수도 있다. 전송된 것과 같이, 비디오 인코더 (20) 는 통상적으로 POC 값 리셋을 설명하기 위해 POC 값 리세팅을 수행할 시 LTRP들의 POC LSB 값들을 통상적으로 조정할 (즉, 감분할) 수도 있다. 따라서, 비디오 인코더 (20) 는 통상적으로 감분된 LTRP들의 POC LSB 값들을 슬라이스 헤더들에 (및/또는 PPS 또는 SPS 예) 포함시킬 수도 있다. (예컨대, POC 값 리셋 이후) 값이 음인 LTRP들에 대하여, 비디오 인코더 (20) 또는 비디오 디코더 (30) 는 변환 알고리즘을 사용하여 음의 POC LSB 값을 양의 POC LSB 값으로 변환할 수도 있다.

[0069] 그러나, 일부 비디오 디코더들은 POC 값 리세팅과 연관된 인덱스를 디코딩하도록 구성되지 않을 수도 있다.

예시의 목적들을 위한 비-제한적인 예에서, HEVC 부합 비디오 디코더는 멀티-계층 비트스트림으로부터 추출된 독립적인 비-기본 계층을 수신할 수도 있다. HEVC 부합 비디오 디코더는 POC 값 리세팅 실택스를 적절히 해석하지 않을 수도 있다. 그러한 경우들에서, HEVC 부합 비디오 디코더는 POC 값 리세팅을 수행하지 않을 수도 있다. 예를 들어, HEVC 부합 비디오 디코더는 POC 값 리세팅 정보를 삭제할 수도 있고, HEVC 부합 비디오 디코더의 DPB 에 저장된 픽처들의 POC 값들을 감분하지 않을 수도 있다.

[0070] 앞서 설명된 예에서, HEVC 부합 비디오 디코더는, HEVC 부합 비디오 디코더가 POC 값 리세팅을 수행하거나 DPB 에 저장된 픽처들의 POC 값들을 감분하도록 구성되지 않을 수도 있기 때문에, 슬라이스 헤더에서 시그널링된 LTRP들의 POC LSB 값들에 기초하여 LTRP들을 적절히 식별하지 않을 수도 있다. 즉, HEVC 부합 비디오 디코더는 (POC 값 리세팅에 의해) 픽처들의 슬라이스 헤더들에서 비디오 인코더 (20) 에 의해 감분된 LTRP들의 POC 값들을 수신할 수도 있다. 그러나, 슬라이스 헤더에서 시그널링된 LTRP들의 POC 값들은, HEVC 부합 비디오 디코더가 DPB 의 픽처들의 POC 값들을 감분하지 않았기 때문에, DPB 에서의 참조 픽처들과 적절히 정렬하지 않을 수도 있다. 따라서, HEVC 부합 비디오 디코더는 수신된 POC 값에 기초하여 DPB 에서 LTRP들을 정확히 식별하지 않을 수도 있다.

[0071] 예시의 목적들을 위한 일 예에서, 비디오 인코더 (20) 는 10 및 20 의 POC LSB 값들을 갖는 LTRP들을 사용하는 멀티-계층 비트스트림의 독립적인 비-기본 계층의 픽처를 인코딩할 수도 있다. 비디오 인코더 (20) 는 60 의 POC LSB 값을 갖는 픽처를 POC 값 리세팅하는 것을 수행할 수도 있다. 따라서, 비디오 인코더 (20) 는 DPB 에 저장된 픽처들의 POC 값들을 60 만큼 감분하고, 감분된 POC LSB 값들 (예컨대, -50 및 -40) 을 사용하여 LTRP들을 시그널링할 수도 있다.

[0072] 그러나, 비디오 디코더가 (전술된 HEVC 부합 비디오 디코더와 같은) POC 값 리세팅을 수행하지 않고 비트스트림을 디코딩하도록 구성되는 경우들에서, 비디오 디코더는 DPB 에 저장된 픽처들의 POC LSB 값들을 감분하지 않을 수도 있다. 예를 들어, 멀티-계층 비트스트림의 독립적인 비-기본 계층을 추출할 시, 비디오 디코더는 슬라이스 헤더로부터 -50 및 -40 의 LTRP POC LSB 값들의 표시들을 디코딩할 수도 있다. 그러나, DPB 에 저장된 대응하는 픽처들은, 비디오 디코더 (30) 가 POC 값 리세팅을 수행하지 않았기 때문에, 10 및 20 의 값들을 여전히 가질 수도 있다. 따라서, 비디오 디코더는 시그널링된 POC LSB 값들에 기초하여 DPB 에서 LTRP들을 적절히 식별하지 않을 수도 있다.

[0073] 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 POC 값 리세팅이 수행되는 방식을 제어할 수도 있다. 그 기술들은, 일부 경우들에 있어서, 슬라이스 헤더에서 시그널링된 LTRP들의 POC 값들이 독립적인 비-기본 계층을 디코딩할 경우에 비디오 디코더의 DPB 에 저장된 참조 픽처들의 POC 값들과 일치성을 유지하게 할 수도 있다.

[0074] 일 예에서, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처를 코딩 (인코딩 또는 디코딩) 할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 또한, 오직 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에만 픽처의 POC LSB 값에 대한 POC 값 리세팅을 표시하는 데이터를 코딩할 수도 있다. 그러한 경우들에서, POC 값 리세팅을 수행하는 것은, POC LSB 값이 이미 0 과 동일하기 때문에, 비디오 인코더 (20) 또는 비디오 디코더 (30) 가 DPB 에 저장된 픽처들의 POC 값들을 감분할 것을 요구하지 않는다.

[0075] 예시의 목적들을 위해 전술된 예로 리턴하여, 비디오 인코더 (20) 및 비디오 디코더 (30) 가 (예컨대, 앞서 언급된 60 의 POC LSB 값보다는) 0 의 POC LSB 값을 갖는 픽처에 POC 값 리세팅을 수행한다면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는, POC LSB 값과 0 간의 차이가 0 이기 때문에, DPB 에 저장된 픽처들의 POC LSB 값들을 감분하지 않는다. 따라서, 슬라이스 헤더에서 시그널링된 LTRP들의 POC LSB 값들은 비디오 디코더 (30) 에서 DPB 에 저장된 픽처들의 POC LSB 값들과 일치한다.

[0076] 일부 예들에서, 본 개시물의 양태들에 따라, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 POC 값 리세팅이 오직 0 과 동일한 POC LSB 값을 가지는 픽처들에 대해서만 수행되도록, 제약들을 적용할 수도 있다. 일 예에서, 독립적인 비-기본 계층 (layerA) 이 추출될 수 있고 layerA 에서 픽처들의 VCL NAL 유닛의 슬라이스 헤더를 변경하지 않고 특정 표준과의 부합을 유지 (예컨대, HEVC 의 프로파일과의 부합을 유지) 하도록 하기 위해, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 다음 제약들을 적용할 수도 있다: (1) layerA 에서 IDR 픽처들의 모든 슬라이스 세그먼트 헤더에 대하여, slice_pic_order_cnt_lsb 실택스 엘리먼트가 존재하지 않아야 한다; 및 (2) 계층에서 각각의 픽처는 POC 값의 MSB 및 LSB 값들 양자를 리세팅하는 POC 리세팅 픽처에 선행하는 장기 참조 픽처를 가지지 않아야 한다.

- [0077] 다른 예에서, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 다음 제약들을 적용할 수도 있다: (1) layerA 에서 IDR 픽처들의 모든 슬라이스 세그먼트 헤더에 대하여, slice_pic_order_cnt_lsb 선택스 엘리먼트가 존재하지 않아야 한다; 및 (2) layerA 에서 POC 값의 MSB 및 LSB 값들 양자를 리셋하는 POC 리세팅 픽처는, 그러한 POC 리세팅 픽처가 (리셋 이전에) 0 과 동일한 POC LSB 를 가지지 않는다면, 존재하지 않아야 한다.
- [0078] 다른 예에서, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 다음 제약들을 적용할 수도 있다: (1) layerA 에서 IDR 픽처들의 모든 슬라이스 세그먼트 헤더들에 대하여, slice_pic_order_cnt_lsb 선택스 엘리먼트가 존재하지 않아야 한다; (2) layerA 에서 POC 값의 MSB 및 LSB 값들 양자를 리셋하는 POC 리세팅 픽처는, 그러한 POC 리세팅 픽처가 또한 IDR 픽처가 아니라면, 존재하지 않아야 한다; 및 (3) layerA 에서 POC 값의 MSB 및 LSB 값들을 리셋하는 것을 표시하는 3 과 동일한 poc_reset_idc 선택스 엘리먼트를 갖는 POC 리세팅 픽처는, 그 리세팅 픽처들이 또한 IDR 픽처가 아니라면, 존재하지 않아야 한다.
- [0079] 여전히 다른 예에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 이하 더 상세히 설명되는 것과 같이, poc_lsb_not_present_flag[i] 의 값에 대한 base_layer_parameter_set_compatibility_flag[i] 의 존재를 컨디셔닝할 수도 있다. 또 다른 예에서, 비디오 인코더 (20) 는 각각의 독립적인 비-기본 계층에 대한 플래그를, 계층이 추출 이후 특정 표준 (예컨대, HEVC) 과의 호환성을 유지하는지의 여부를 표시하도록 시그널링할 수도 있다 (그리고 비디오 디코더 (30) 는 디코딩할 수도 있다).
- [0080] 본 게시물의 기술들은 또한, 멀티-계층 액세스 유닛의 픽처들의 POC 값 정렬을 유지하는 것 (예컨대, 액세스 유닛의 모든 픽처들이 동일한 POC 값을 갖도록 유지하는 것) 에 관한 것이다. 예를 들어, POC 값 리세팅은 통상적으로 IDR 픽처들에 대하여 수행되지만, 일부 경우들에서 멀티-계층 코딩에 있어서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 IDR 픽처의 POC 값을 리셋하지 않을 수도 있다. 예시의 목적들을 위한 일 예에서, 액세스 유닛은 IDR 픽처가 아닌 기본 계층 픽처 및 IDR 픽처인 비-기본 계층 픽처를 포함할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 가 통상적으로 비-기본 계층의 IDR 픽처에 대하여 POC 값 리세팅을 수행할 수도 있지만, 액세스 유닛에서 POC 값 정렬을 유지하기 위해, 비디오 인코더 (20) 는 대신, 비-기본 계층 픽처에 대한 기본 계층의 POC LSB 값을 시그널링할 수도 있다 (그리고 비디오 디코더 (30) 는 디코딩할 수도 있다). 예를 들어, 비디오 인코더 (20) 는 (0 과 동일하지 않은) 기본 계층의 POC LSB 값의 표시를 비-기본 계층 픽처의 슬라이스 헤더에 포함시킬 수도 있다.
- [0081] 그러나, 전술된 예는 독립적인 비-기본 계층을 코딩할 때, 도전과제를 제시할 수도 있다. 예시의 목적들을 위한 비-제한적인 예에서, HEVC 부합 비디오 디코더는 IDR 픽처의 POC LSB 값을 0 과 동일하도록 리셋할 수도 있다. 그러한 예에서, HEVC 부합 비디오 디코더는 POC LSB 값이 0 이기 때문에, IDR 픽처의 슬라이스들의 슬라이스 헤더들로부터 POC LSB 값의 표시를 디코딩하도록 구성되지 않을 수도 있다. 그러나, HEVC 부합 비디오 디코더는 HEVC 부합 비디오 디코더가 IDR 픽처들의 슬라이스 헤더들로부터 POC LSB 값을 디코딩하도록 구성되지 않을 수도 있기 때문에, POC LSB 값이 IDR 픽처의 슬라이스 헤더들에서 시그널링될 경우 오작동할 수도 있다. 그 후에, 디코더는 시퀀스에서 다음 IRAP 픽처에 앞서 스킵할 수도 있고, 이는 사용자 경험에 있어서 연속성을 저하시킬 수도 있다.
- [0082] 앞서 언급된 MV-HEVC 및 SHVC 는 서브-비트스트림이 poc_lsb_not_present_flag 를 포함하는 하나 이상의 선택스 엘리먼트들을 사용하여 독립적으로 디코딩가능한지 여부를 표시할 수도 있다. 일부 경우들에 있어서, 1 과 동일한 poc_lsb_not_present_flag 는 poc_lsb_not_present_flag 와 연관된 계층이 예컨대, HEVC 부합 비디오 디코더에 의해 독립적으로 디코딩가능한 것을 표시할 수도 있다. 대조적으로, 0 과 동일한 poc_lsb_not_present_flag 는 슬라이스 헤더들에 포함된 선택스와 같은 특정 선택스가 그 계층이 독립적으로 디코딩되기 위해 수정되어야만 할 수도 있음을 표시할 수도 있다.
- [0083] 일반적으로, poc_lsb_not_present_flag 는 POC LSB 값이 IDR 픽처의 슬라이스 헤더에 포함되는지 여부를 표시할 수도 있다. SHVC 및 MV-HEVC 에 따르면, 1 과 동일한 poc_lsb_not_present_flag 는 POC LSB 값이 IDR 픽처들의 슬라이스 헤더들에 존재하지 않는 것을 명시한다. 0 과 동일한 poc_lsb_not_present_flag 는 POC LSB 값이 IDR 픽처들의 슬라이스 헤더들에 존재할 수도 있거나 존재하지 않을 수도 있는 것을 명시한다. poc_lsb_not_present_flag 는, POC LSB 값이 슬라이스 헤더에 포함될 수도 있다는 표시를 제공하기 때문에, 비-기본 계층의 상대적으로 더 용이한 추출 및 디코딩을 허용할 수도 있다.
- [0084] 앞서 언급된 SHVC 및 MV-HEVC 표준들에 대하여, 1 과 동일한 poc_lsb_not_present_flag (예컨대, POC LSB 값이 슬라이스 헤더에 포함되지 않는 것을 표시함) 를 세팅하는 것에 관한 유일한 명시적인 제약은, 코딩되는 계층이 독립적인 계층이라는 것이다. 그러나, POC LSB 값이 슬라이스 헤더에 포함되지 않는 것을 표시하는 것은,

이하 도 5 의 예에 대하여 더 상세히 설명되는 것과 같이, 0 의 POC LSB 값을 가지지 않는 독립적인 계층의 픽처들에 대하여 문제를 생성할 수도 있다.

[0085] 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를, 오직 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에만 코딩할 수도 있다. 예를 들면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는, 계층에서의 IDR 픽처의 POC LSB 값이 0 과 동일한 것으로 추론될 수도 있지 않다면, poc_lsb_not_present_flag [i] 의 값이 (예컨대 POC LSB 값이 슬라이스 헤더에 포함되지 않는 것을 표시하는) 1 과 동일하지 않도록 제약을 적용한다. 독립적인 비-기본 계층 픽처의 POC LSB 값은, 동일한 액세스 유닛에서 기본 계층 픽처의 POC LSB 값이 IDR 픽처일 경우 또는 비-기본 계층 픽처의 POC LSB 값이 (예컨대, 0 으로부터 최대 POC LSB 값까지 POC LSB 값들의 정상적인 진행 이후에) 0 의 값일 경우에, 0 과 동일한 것으로 추론될 수도 있다.

[0086] 도 2 는 본 개시물의 기술들에 따라 POC 값들을 관리하기 위한 기술들을 구현할 수도 있는 비디오 인코더 (20) 의 일 예를 도시한 블록 다이어그램이다. 특히, 비디오 인코더 (20) 는 본 개시물의 기술들에 따라 멀티-계층 비트스트림의 픽처들의 POC 값들을 관리하도록 구성될 수도 있다. 상기 언급된 바와 같이, 비디오 인코더 (20) 는 멀티뷰 및/또는 스케일러블 비디오 코딩을 수행하도록 적용될 수도 있다. 예를 들어, 비디오 인코더 (20) 는, SHVC 또는 MV-HEVC 와 같은 하나 이상의 비디오 코딩 표준 확장들에 부합하는 비트스트림을 인코딩하도록 구성될 수도 있다. 하지만, 특정 코딩 표준들이 참조되는 한편, 그 기술들은 임의의 하나의 코딩 표준에 특정되지 않고 장래의 및/또는 아직 개발되지 않은 표준들로 구현될 수도 있음이 이해되어야 한다.

[0087] 비디오 인코더 (20) 는 비디오 슬라이스들 내에서 비디오 블록들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라-코딩은 소정의 비디오 프레임 또는 픽처 내 비디오에 있어서 공간 리던던시를 감소 또는 제거하기 위해 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 인접한 프레임들 또는 픽처들 내 비디오에 있어서 시간 리던던시를 감소 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I 모드) 는 수개의 공간 기반 코딩 모드들 중 임의의 모드를 지칭할 수도 있다. 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 과 같은 인터-모드들은 수 개의 시간 기반 코딩 모드들 중 임의의 모드를 지칭할 수도 있다.

[0088] 도 2 에 도시된 바와 같이, 비디오 인코더 (20) 는 인코딩될 비디오 프레임 내의 현재 비디오 블록을 수신한다. 도 2 의 예에 있어서, 비디오 인코더 (20) 는 비디오 데이터 메모리 (38), 모드 선택 유닛 (40), 참조 픽처 메모리 (64), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 모드 선택 유닛 (40) 은 차례로, 모션 보상 유닛 (44), 모션 추정 유닛 (42), 인트라-예측 유닛 (46), 및 파티션 유닛 (48) 을 포함한다. 비디오 블록 복원을 위해, 비디오 인코더 (20) 는 또한 역양자화 유닛 (58), 역변환 유닛 (60), 및 합산기 (62) 를 포함한다. 디블로킹 (deblocking) 필터 (도 2 에 도시 안됨) 가 또한, 블록 경계들을 필터링하여 복원된 비디오로부터 블록키니스 아티팩트들을 제거하기 위해 포함될 수도 있다. 요구된다면, 디블로킹 필터는 통상적으로 합산기 (62) 의 출력을 필터링할 것이다. 부가적인 필터 (인 루프 (in loop) 또는 포스트 루프) 가 또한 디블로킹 필터에 부가하여 이용될 수도 있다. 그러한 필터들은 간략화를 위해 도시되지 않지만, 요구된다면, (인-루프 필터로서) 합산기 (50) 의 출력을 필터링할 수도 있다.

[0089] 인코딩 프로세스 동안, 비디오 인코더 (20) 는 코딩될 비디오 프레임 또는 슬라이스를 수신한다. 그 프레임 또는 슬라이스는 다수의 비디오 블록들로 분할될 수도 있다. 비디오 데이터 메모리 (38) 는 비디오 인코더 (20) 의 컴포넌트들에 의해 인코딩될 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (38) 에 저장된 비디오 데이터는, 예컨대 비디오 소스 (18) 로부터 획득될 수도 있다. 참조 픽처 메모리 (64) 는 예컨대, 인트라-코딩 또는 인터-코딩 모드들에서 비디오 인코더 (20) 에 의해 비디오 데이터를 인코딩하는데 사용하기 위해 참조 비디오 데이터를 저장하는 DPB 로 지칭될 수도 있다. 비디오 데이터 메모리 (38) 및 참조 픽처 메모리 (64) 는 동기식 DRAM (SDRAM) 을 포함한 동적 랜덤 액세스 메모리 (DRAM), 자기저항성 RAM (MRAM), 저항성 RAM (RRAM), 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들 중 임의의 메모리 디바이스에 의해 형성될 수도 있다. 비디오 데이터 메모리 (38) 및 참조 픽처 메모리 (64) 는 동일한 메모리 디바이스 또는 분리된 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에 있어서, 비디오 데이터 메모리 (38) 는 비디오 인코더 (20) 의 다른 컴포넌트들과 온-칩형이거나 그 컴포넌트들에 대하여 오프-칩형일 수도 있다.

[0090] 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 시간 예측을 제공하기 위해 하나 이상의 참조 프레임들에 있어서의 하나 이상의 블록들에 대해 수신된 비디오 블록의 인터-예측 코딩을 수행한다. 대안적으로, 인트라-예

측 유닛 (46) 은 공간 예측을 제공하기 위해 코딩될 블록과 동일한 프레임 또는 슬라이스에 있어서의 하나 이상의 이웃한 블록들에 대해 수신된 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다. 비디오 인코더 (20) 는, 예를 들어, 비디오 데이터의 각각의 블록에 대한 적당한 코딩 모드를 선택하기 위해 다중의 코딩 패스들을 수행할 수도 있다.

[0091] 더욱이, 파티션 유닛 (48) 은 이전 코딩 패스들에 있어서의 이전 파티셔닝 방식들의 평가에 기초하여 비디오 데이터의 블록들을 서브-블록들로 파티셔닝할 수도 있다. 예를 들어, 파티션 유닛 (48) 은, 레이트-왜곡 분석 (예를 들어, 레이트-왜곡 최적화) 에 기초하여, 처음에 프레임 또는 슬라이스를 LCU들로 파티셔닝하고, LCU들 각각을 서브-CU들로 파티셔닝할 수도 있다. 모드 선택 유닛 (40) 은 추가로, LCU 의 서브-CU들로의 파티셔닝을 표시하는 쿼드트리 데이터 구조를 생성할 수도 있다. 쿼드트리의 리프 노드 CU들은 하나 이상의 PU들 및 하나 이상의 TU들을 포함할 수도 있다.

[0092] 모드 선택 유닛 (40) 은, 예를 들어, 여러 결과들에 기초하여 코딩 모드들 중 하나, 즉, 인트라 모드 또는 인터 모드를 선택할 수도 있고, 결과적인 인트라- 또는 인터-코딩된 블록을 합산기 (50) 에 제공하여 잔차 블록 데이터를 생성하고, 합산기 (62) 에 제공하여 참조 프레임으로서의 사용을 위한 인코딩된 블록을 복원한다. 모드 선택 유닛 (40) 은 또한, 모션 벡터들, 인트라 모드 표시자들, 파티션 정보, 및 다른 그러한 선택스 정보와 같은 선택스 엘리먼트들을 엔트로피 인코딩 유닛 (56) 에 제공한다.

[0093] 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적을 위해 별개로 도시된다. 모션 추정 유닛 (42) 에 의해 수행된 모션 추정은, 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 현재 슬라이스 (또는 다른 코딩된 유닛) 내에서 인코딩되고 있는 현재 블록에 대한 참조 프레임 (또는 다른 코딩된 유닛) 내의 예측 블록에 대한 현재 비디오 프레임 또는 픽처 내의 비디오 블록의 PU 의 변위를 표시할 수도 있다. 예측 블록은 픽셀 차이의 관점에서 인코딩될 블록과 밀접하게 매칭되도록 발견되는 블록이며, 이 픽셀 차이는 절대 차이의 합 (SAD), 제곱 차이의 합 (SSD), 또는 다른 상이한 메트릭들에 의해 결정될 수도 있다.

[0094] 모션 추정 유닛 (42) 은 인터-코딩된 슬라이스에 있어서의 비디오 블록의 PU 에 대한 모션 벡터를, 그 PU 의 포지션을 참조 픽처의 예측 블록의 포지션과 비교함으로써 계산한다. 참조 픽처는 제 1 참조 픽처 리스트 (리스트 0) 또는 제 2 참조 픽처 리스트 (리스트 1) 로부터 선택될 수도 있으며, 이 리스트들 각각은 참조 픽처 메모리 (64) 에 저장된 하나 이상의 참조 픽처들을 식별한다. 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다.

[0095] 모션 보상 유닛 (44) 에 의해 수행된 모션 보상은 모션 추정 유닛 (42) 에 의해 결정된 모션 벡터에 기초하여 예측 블록을 페치 또는 생성하는 것을 수반할 수도 있다. 다시, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 일부 예들에 있어서 기능적으로 통합될 수도 있다. 현재 비디오 블록의 PU 에 대한 모션 벡터를 수신할 시, 모션 보상 유닛 (44) 은, 모션 벡터가 참조 픽처 리스트들 중 하나에 포인팅하는 예측 블록을 로케이팅할 수도 있다. 합산기 (50) 는, 하기에서 논의되는 바와 같이, 인코딩되고 있는 현재 비디오 블록의 픽셀 값들로부터 예측 블록의 픽셀 값들을 감산하여 픽셀 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다.

일반적으로, 모션 추정 유닛 (42) 은 루마 컴포넌트들에 대한 모션 추정을 수행하고, 모션 보상 유닛 (44) 은 크로마 컴포넌트들 및 루마 컴포넌트들 양자에 대한 루마 컴포넌트들에 기초하여 계산된 모션 벡터들을 이용한다. 모드 선택 유닛 (40) 은 또한, 비디오 슬라이스의 비디오 블록들을 디코딩함에 있어서 비디오 디코더 (30) 에 의한 사용을 위해 비디오 블록들 및 비디오 슬라이스와 연관된 선택스 엘리먼트들을 생성할 수도 있다.

[0096] 모션 추정 유닛 (42) 은 참조 픽처 메모리 (64) 의 DPB 에 저장된 참조 픽처들을 탐색한다. 모션 선택 유닛 (40) 은 픽처, 픽처의 슬라이스, 및/또는 픽처들의 시퀀스를 인코딩하는데 어느 참조 픽처들이 사용될지를 결정할 수도 있다. 비디오 인코더 (20) 는 픽처, 슬라이스, 또는 시퀀스를 인코딩하는데 사용된 참조 픽처들에 대한 픽처 순서 카운트 (POC) 값들을 인코딩할 수도 있다. 비디오 인코더 (20) 는 PPS들 및/또는 SPS들과 같은 슬라이서 헤더들 및/또는 파라미터 세트들에서 참조 픽처 리스트에 포함될 참조 픽처들을 표시하는 POC 값들을 인코딩할 수도 있다.

[0097] 이러한 방식으로, 비디오 디코더 (30) 와 같은 비디오 디코더는 슬라이스 헤더 및/또는 파라미터 세트(들)에 표시된 참조 픽처들을 포함시킴으로써, 참조 픽처 리스트를 재생할 수도 있다. 추가로, 모션 추정 유닛 (42) 에 의해 생성된 모션 벡터를 사용하여 블록을 인코딩한 후에, 비디오 인코더 (20) 는 그 블록에 대한 모션 정보를 인코딩할 수도 있고, 여기서 모션 정보는 모션 벡터를 나타내는 데이터, 참조 픽처 리스트에 대한 식별자, 및 참조 픽처 리스트에서 참조 픽처를 식별하는 참조 인덱스를 포함할 수도 있다.

- [0098] 일부 예들에 있어서, 비디오 인코더 (20) 는 참조 픽처 메모리 (64) 에 저장된 참조 픽처들의 서브-정수 픽셀 포지션들에 대한 값들을 계산할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 참조 픽처의 1/4 픽셀 포지션들, 1/8 픽셀 포지션들, 또는 다른 분수 픽셀 포지션들의 값들을 보간할 수도 있다. 따라서, 모션 추정 유닛 (42) 은 풀 픽셀 포지션들 및 분수 픽셀 포지션들에 대한 모션 탐색을 수행하고, 분수 픽셀 정밀도로 모션 벡터를 출력할 수도 있다.
- [0099] 비디오 인코더 (20) 는 또한, POC 리세팅 주기들을 형성하고, 픽처들의 슬라이스들에 대한 POC 리세팅 주기 식별자들의 값들을 인코딩하여 슬라이스들을 개별 POC 리세팅 주기들에 할당할 수도 있다. 비디오 인코더 (20) 는 비디오 데이터의 각 계층 내에서 슬라이스들을 POC 리세팅 주기들에 개별적으로 할당할 수도 있다. 추가로, 비디오 인코더 (20) 는 슬라이스들에 대한 POC 리세팅 타입들을 결정하고, 슬라이스들에 대한 POC 리세팅 타입들의 표시들에 대한 값들 (예컨대, 2-비트 값들) 을 인코딩할 수도 있다.
- [0100] 도 1 에 대하여 논의된 것과 같이, 비디오 인코더 (20) 는 적용가능한 비디오 코딩 표준에 의해 확립된 제약들에 따라 구성될 수도 있다. 예를 들어, 비디오 인코더 (20) 는 각각의 POC 리세팅 주기에서 제 1 픽처가 POC 값들이 완전히 리셋될 것인지 또는 부분적으로 리셋될 것인지를 표시하는 POC 리세팅 타입 (예컨대, poc_reset_idc) 를 가지는 것을 보장할 수도 있다. 일반적으로, 전술된 것과 같이, 비디오 인코더 (20) 는 코딩 순서에서 현재 픽처 이전의 픽처의 POC MSB 값에 기초하여, 또는 슬라이스 헤더에 시그널링된 추가의 정보 (예컨대, POC MSB 사이클) 에 기초하여 현재 픽처에 대한 POC MSB 값을 결정할 수도 있다.
- [0101] 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 는 오직 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에만 독립적인 비-기본 계층의 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 인코딩할 수도 있다. 예를 들면, 비디오 인코더 (20) 는 poc_reset_idc 신택스 엘리먼트를 인코딩할 수도 있다. 비디오 인코더 (20) 는, 0 과 동일한 poc_reset_idc 신택스 엘리먼트를, 현재 픽처에 대한 POC 값의 MSB들 및 LSB들이 리셋되지 않는 것을 명시하도록 세팅할 수도 있다. 비디오 인코더 (20) 는, 1 과 동일한 poc_reset_idc 신택스 엘리먼트를, 오직 현재 픽처에 대한 POC 값의 MSB들만이 리셋될 수도 있는 것을 명시하도록 세팅할 수도 있다. 비디오 인코더 (20) 는, 2 와 동일한 poc_reset_idc 신택스 엘리먼트를, 현재 픽처에 대한 POC 값의 MSB들 및 LSB들 양자가 리셋될 수도 있는 것을 명시하도록 세팅할 수도 있다. 비디오 인코더 (20) 는, 3 과 동일한 poc_reset_idc 신택스 엘리먼트를, 현재 픽처에 대한 POC 값의 오직 MSB들 또는 MSB들 및 LSB들 양자가 리셋될 수도 있는 것 및 추가의 픽처 순서 카운트 정보가 시그널링될 수도 있는 것을 명시하도록 세팅할 수도 있다.
- [0102] 비디오 인코더 (20) 는 poc_reset_idc 값이 하나 이상의 제약들에 기초하여 세팅되는 방식을 제어할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 poc_reset_idc 신택스 엘리먼트가 결과적인 비트스트림이 특정 표준에 부합하는 것을 보장하도록 세팅되는 방식을 제어할 수도 있다. 본 개시물의 양태들에 따르면, 비디오 인코더는 독립적인 비-기본 계층이 예컨대, 독립적인 비-기본 계층의 코딩된 슬라이스들의 슬라이스 헤더들에 포함된 정보를 변경/업데이트하지 않고, 독립적으로 디코딩되게 하는 방식으로 멀티-계층 비트스트림의 독립적인 비-기본 계층에서 픽처에 대한 poc_reset_idc 신택스 엘리먼트를 세팅할 수도 있다.
- [0103] 예를 들어, 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 는 오직 코딩되고 있는 픽처가 0 과 동일한 POC LSB 값을 가질 경우에만 2 와 동일한 poc_reset_idc 신택스 엘리먼트를 인코딩할 수도 있다. 즉, 코딩되고 있는 특정 계층에 대한 poc_lsb_not_present_flag 가 (예컨대, 그 계층이 독립적인 비-기본 계층인 것을 표시하는) 1 과 동일하고 픽처의 POC LSB 값을 표시하는 slice_pic_order_cnt_lsb 신택스 엘리먼트가 0 보다 클 경우, 비디오 인코더 (20) 는 poc_reset_idc 의 값을 2 와 동일한 것으로 세팅하지 않을 수도 있다.
- [0104] 비디오 인코더 (20) 가 3 과 동일하도록 poc_reset_idc 신택스 엘리먼트를 인코딩할 경우, 비디오 인코더 (20) 는 추가로 full_poc_reset_flag 및 poc_lsb_val 신택스 엘리먼트를 인코딩할 수도 있다. MV-HEVC 및 SHVC 에서, 1 과 동일한 full_poc_reset_flag 는, 동일한 계층에서 디코딩 순서에 있어서 이전 픽처가 동일한 POC 리세팅 주기에 속하지 않을 경우, 현재 픽처에 대한 POC 값의 MSB들 및 LSB들 양자가 리셋되는 것을 명시한다. 0 과 동일한 full_poc_reset_flag 는, 동일한 계층에서 디코딩 순서에 있어서 이전 픽처가 동일한 POC 리세팅 주기에 속하지 않을 경우, 현재 픽처에 대한 POC 값의 오직 MSB들만이 리셋되는 것을 명시한다. 추가로, poc_lsb_val 신택스 엘리먼트는 현재 픽처의 픽처 순서 카운트를 도출하는데 사용될 수도 있는 값을 명시한다.
- [0105] 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 는 인코딩되고 있는 계층이 독립적인 비-기본 계층인지 여부에 기초하여 poc_lsb_val 이 인코딩되는 방식을 제어할 수도 있다. 예를 들어, 인코딩되고 있는 계층에 대한 poc_lsb_not_present_flag 가 1 과 동일하고, full_poc_reset_flag 가 1 과 동일할 경우, 비디오 인코더

(20)는 poc_lsb_val의 값을 0과 동일하게 세팅할 수도 있다. 이러한 방식으로, 비디오 인코더 (20)는 POC 값 리세팅이 오직 0과 동일한 POC LSB 값을 갖는 픽처들에 대해서만 수행되는 것을 보장할 수도 있고, 이는 참조 픽처들이 도 4에 대하여 이하 더 상세히 설명되는 것과 같이, 참조 픽처 세트에서 적절히 식별되게 할 수도 있다.

[0106] 인트라-예측 유닛 (46)은 현재 블록을, 상기 설명된 바와 같은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44)에 의해 수행된 인터-예측에 대한 대안으로서 인트라-예측할 수도 있다. 특히, 인트라-예측 유닛 (46)은 현재 블록을 인코딩하는데 이용하기 위한 인트라-예측 모드를 결정할 수도 있다. 일부 예들에 있어서, 인트라-예측 유닛 (46)은 예를 들어 별도의 인코딩 패스들 동안에 다양한 인트라-예측 모드들을 이용하여 현재 블록을 인코딩할 수도 있으며, 인트라-예측 유닛 (46) (또는 일부 예들에서는 모드 선택 유닛 (40))은 테스트된 모드들로부터의 이용을 위해 적절한 인트라-예측 모드를 선택할 수도 있다.

[0107] 예를 들어, 인트라-예측 유닛 (46)은 다양한 테스트된 인트라-예측 모드들에 대한 레이트-왜곡 분석을 이용하여 레이트-왜곡 값들을 계산하고, 테스트된 모드들 중 최상의 레이트-왜곡 특성들을 갖는 인트라-예측 모드를 선택할 수도 있다. 레이트-왜곡 분석은 일반적으로, 인코딩된 블록과 그 인코딩된 블록을 생성하도록 인코딩되었던 원래의 인코딩되지 않은 블록 간의 왜곡 (또는 에러)의 양뿐 아니라 인코딩된 블록을 생성하는데 사용된 비트 레이트 (즉, 비트들의 수)를 결정한다. 인트라-예측 유닛 (46)은 다양한 인코딩된 블록들에 대한 왜곡 및 레이트들로부터의 비율들을 계산하여, 어느 인트라-예측 모드가 그 블록에 대한 최상의 레이트-왜곡 값을 나타내는지 결정할 수도 있다.

[0108] 블록에 대한 인트라-예측 모드를 선택한 이후, 인트라-예측 유닛 (46)은 블록에 대한 선택된 인트라-예측 모드를 표시하는 정보를 엔트로피 인코딩 유닛 (56)에 제공할 수도 있다. 엔트로피 인코딩 유닛 (56)은 선택된 인트라-예측 모드를 표시한 정보를 인코딩할 수도 있다. 비디오 인코더 (20)는 송신된 비트스트림에, 복수의 인트라-예측 모드 인덱스 테이블들 및 복수의 수정된 인트라-예측 모드 인덱스 테이블들 (코드워드 매핑 테이블들로서도 또한 지칭됨)을 포함할 수도 있는 구성 데이터, 다양한 블록들에 대한 인코딩 컨텍스트들의 정의들, 및 컨텍스트들 각각에 대한 사용을 위한 가장 가능성있는 인트라-예측 모드, 인트라-예측 모드 인덱스 테이블, 및 수정된 인트라-예측 모드 인덱스 테이블의 표시들을 포함할 수도 있다.

[0109] 비디오 인코더 (20)는 모드 선택 유닛 (40)으로부터의 예측 데이터를, 코딩되는 원래의 비디오 블록으로부터 감산함으로써 잔차 비디오 블록을 형성한다. 합산기 (50)는 이러한 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 변환 프로세싱 유닛 (52)은 이산 코사인 변환 (DCT) 또는 개념적으로 유사한 변환과 같은 변환을 잔차 블록에 적용하여, 잔차 변환 계수 값들을 포함하는 비디오 블록을 생성한다. 변환 프로세싱 유닛 (52)은, DCT와 개념적으로 유사한 다른 변환들을 수행할 수도 있다. 웨이블릿 변환들, 정수 변환들, 서브-대역 변환들, 또는 다른 타입들의 변환들이 또한 이용될 수 있다. 어떠한 경우라도, 변환 프로세싱 유닛 (52)은 변환을 잔차 블록에 적용하여, 잔차 변환 계수들의 블록을 생성한다. 그 변환은 잔차 정보를 픽셀 값 도메인으로부터 주파수 도메인과 같은 변환 도메인으로 컨버팅할 수도 있다.

[0110] 변환 프로세싱 유닛 (52)은 결과적인 변환 계수들을 양자화 유닛 (54)으로 전송할 수도 있다. 양자화 유닛 (54)은 비트 레이트를 추가로 감소시키기 위해 변환 계수들을 양자화한다. 양자화 프로세스는 그 계수들의 일부 또는 그 모두와 연관되는 비트 심도를 감소시킬 수도 있다. 양자화의 정도는 양자화 파라미터를 조정함으로써 수정될 수도 있다. 그 후, 일부 예들에 있어서, 양자화 유닛 (54)은, 양자화된 변환 계수들을 포함하는 매트릭스의 스캔을 수행할 수도 있다. 대안적으로, 엔트로피 인코딩 유닛 (56)은 스캔을 수행할 수도 있다.

[0111] 양자화 이후, 엔트로피 인코딩 유닛 (56)은 양자화된 변환 계수들을 엔트로피 코딩한다. 예를 들어, 엔트로피 인코딩 유닛 (56)은 컨텍스트 적응 가변 길이 코딩 (CAVLC), 컨텍스트 적응 바이너리 산술 코딩 (CABAC), 신택스 기반 컨텍스트 적응 바이너리 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 엔트로피 코딩 기술을 수행할 수도 있다. 컨텍스트 기반 엔트로피 코딩의 경우, 컨텍스트는 이웃한 블록들에 기초할 수도 있다. 엔트로피 인코딩 유닛 (56)에 의한 엔트로피 코딩 이후, 인코딩된 비트스트림은 다른 디바이스 (예를 들어, 비디오 디코더 (30))로 송신되거나 또는 더 나중의 송신 또는 취출을 위해 아카이브될 수도 있다.

[0112] 역양자화 유닛 (58) 및 역변환 유닛 (60)은, 예를 들어, 참조 블록으로서의 더 나중 사용을 위해 픽셀 도메인에서 잔차 블록을 복원하도록, 각각, 역양자화 및 역변환을 적용한다. 모션 보상 유닛 (44)은 참조 픽처 메모리 (64)의 프레임들 중 하나의 예측 블록에 잔차 블록을 부가함으로써 참조 블록을 계산할 수도 있다.

모션 보상 유닛 (44) 은 또한, 하나 이상의 보간 필터들을 복원된 잔차 블록에 적용하여, 모션 추정에서 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 복원된 잔차 블록을, 모션 보상 유닛 (44) 에 의해 생성된 모션 보상된 예측 블록에 부가하여, 참조 픽처 메모리 (64) 로의 저장을 위한 복원된 비디오 블록을 생성한다. 복원된 비디오 블록은, 후속 비디오 프레임에서의 블록을 인터-코딩하기 위해 참조 블록으로서 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 사용될 수도 있다.

[0113] 이러한 방식으로, 도 2 의 비디오 인코더 (20) 는, 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처의 적어도 일부분을 인코딩하고, 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터들, 오직 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 갖는 경우에만 인코딩하도록 구성된 비디오 인코더의 일 예를 나타낸다.

[0114] 도 3 은 본 개시물의 기술들에 따라 POC 값들을 관리하기 위한 기술들을 구현할 수도 있는 비디오 디코더 (30) 의 일 예를 도시한 블록 다이어그램이다. 상기 언급된 바와 같이, 비디오 디코더 (30) 는 멀티뷰 및/또는 스케일러블 비디오 코딩을 수행하도록 적용될 수도 있다. 예를 들어, 비디오 디코더 (30) 는, SHVC 또는 MV-HEVC 와 같은 하나 이상의 비디오 코딩 표준 확장들에 부합하는 비트스트림을 디코딩하도록 구성될 수도 있다. 하지만, 특정 코딩 표준들이 참조되는 한편, 그 기술들은 임의의 하나의 코딩 표준에 특정되지 않고 장래의 및/또는 아직 개발되지 않은 표준들로 구현될 수도 있음이 이해되어야 한다.

[0115] 도 3 의 예에 있어서, 비디오 디코더 (30) 는 비디오 데이터 메모리 (68), 엔트로피 디코딩 유닛 (70), 모션 보상 유닛 (72), 인트라-예측 유닛 (74), 역양자화 유닛 (76), 역변환 유닛 (78), 참조 픽처 메모리 (82) 및 합산기 (80) 를 포함한다. 일부 예들에 있어서, 비디오 디코더 (30) 는 비디오 인코더 (20) (도 2) 에 대하여 설명된 인코딩 패스에 일반적으로 상호적인 디코딩 패스를 수행할 수도 있다. 모션 보상 유닛 (72) 은 엔트로피 디코딩 유닛 (70) 으로부터 수신된 모션 벡터들에 기초하여 예측 데이터를 생성할 수도 있지만, 인트라-예측 유닛 (74) 은 엔트로피 디코딩 유닛 (70) 으로부터 수신된 인트라-예측 모드 표시자들에 기초하여 예측 데이터를 생성할 수도 있다.

[0116] 비디오 데이터 메모리 (68) 는 비디오 디코더 (30) 의 컴포넌트들에 의해 디코딩될, 인코딩된 비디오 비트스트림과 같은 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (68) 에 저장된 비디오 데이터는, 예컨대 카메라와 같은 로컬 비디오 소스로부터의 예컨대 컴퓨터 판독가능 매체로부터, 비디오 데이터의 유선 또는 무선 네트워크 통신을 통해 또는 물리적 데이터를 저장 매체에 액세스함으로써 획득될 수도 있다. 비디오 데이터 메모리 (68) 는 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터를 저장하는 코딩된 픽처 버퍼 (CPB) 를 형성할 수도 있다. 참조 픽처 메모리 (82) 는 예컨대, 인트라-코딩 또는 인터-코딩 모드들에서 비디오 디코더 (30) 에 의해 비디오 데이터를 디코딩하는데 사용하기 위해 참조 비디오 데이터를 저장하는 DPB 로 지칭될 수도 있다. 비디오 데이터 메모리 (68) 및 참조 픽처 메모리 (82) 는 동기식 DRAM (SDRAM) 을 포함한 동적 랜덤 액세스 메모리 (DRAM), 자기저항성 RAM (MRAM), 저항성 RAM (RRAM), 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들 중 임의의 메모리 디바이스에 의해 형성될 수도 있다. 비디오 데이터 메모리 (68) 및 참조 픽처 메모리 (82) 는 동일한 메모리 디바이스 또는 분리된 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에 있어서, 비디오 데이터 메모리 (68) 는 비디오 디코더 (30) 의 다른 컴포넌트들과 온-칩형이거나 그 컴포넌트들에 대하여 오프-칩형일 수도 있다.

[0117] 디코딩 프로세스 동안, 비디오 디코더 (30) 는, 인코딩된 비디오 슬라이스의 비디오 블록들 및 관련 신택스 엘리먼트들을 나타내는 인코딩된 비디오 비트스트림을 비디오 인코더 (20) 로부터 수신한다. 비디오 디코더 (30) 의 엔트로피 디코딩 유닛 (70) 은 비트스트림을 엔트로피 디코딩하여, 양자화된 계수들, 모션 벡터들 또는 인트라-예측 모드 표시자들, 및 다른 신택스 엘리먼트들을 생성한다. 엔트로피 디코딩 유닛 (70) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 모션 보상 유닛 (72) 으로 포워딩한다. 비디오 디코더 (30) 는 신택스 엘리먼트들을 비디오 슬라이스 레벨 및/또는 비디오 블록 레벨로 수신할 수도 있다.

[0118] 비디오 슬라이스가 인트라-코딩된 (I) 슬라이스로서 코딩된 경우, 인트라 예측 유닛 (74) 은 현재 프레임 또는 픽처의 이전에 디코딩된 블록들로부터의 데이터 및 시그널링된 인트라-예측 모드에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 데이터를 생성할 수도 있다. 비디오 프레임이 인터-코딩된 (즉, B, P 또는 GPB) 슬라이스로서 코딩된 경우, 모션 보상 유닛 (72) 은 엔트로피 디코딩 유닛 (70) 으로부터 수신된 모션 벡터들 및 다른 신택스 엘리먼트들에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 블록들을 생성한다. 예측 블록들은 참조 픽처 리스트들 중 하나 내에서의 참조 픽처들 중 하나로부터 생성될 수도 있다. 비디오 디코더 (30) 는 참조 픽처 메모리 (82) 의 DPB 에 저장된 참조 픽처들에 기초한 디폴트 구성 기술들을 이용하여, 참조 프레임 리스트들, 즉, 리스트 0 (L0) 및 리스트 1 (L1) 을 구성할 수도 있다.

- [0119] 더 구체적으로, 비디오 디코더 (30) 는 PPS 에 대응하는 픽처의 슬라이스들이 POC 리세팅 정보를 포함하는지 여부를 표시하는 PPS 를 디코딩할 수도 있다. 슬라이스들이 POC 리세팅 정보를 포함한다고 가정하면, 비디오 디코더 (30) 는 POC 리세팅 정보를 포함하는 픽처의 슬라이스의 슬라이스 세그먼트 헤더를 디코딩할 수도 있다. POC 리세팅 정보는 POC 리세팅 주기 식별자 및 POC 리세팅 타입을 포함할 수도 있다.
- [0120] POC 리세팅 주기 식별자는 슬라이스가 대응하는 POC 리세팅 주기를 표시할 수도 있다. POC 리세팅 타입은 슬라이스를 포함하는 픽처의 POC 값이 완전히 리셋되는지 또는 오직 POC 값의 MSB 만이 리셋되는지 여부를 표시할 수도 있다. 추가로, POC 리세팅 타입은 어떤 POC 리셋도 수행되지 않을 것임을 또는 추가의 정보가 시그널링될 것임을 표시할 수도 있다. POC 리세팅 타입이 추가의 정보가 시그널링된다고 표시한다면, 비디오 디코더 (30) 는 POC LSB 값 및 POC 값이 완전히 리셋되는지 또는 오직 MSB 만이 리셋되는지 여부를 표시할 수도 있는 추가의 정보를 디코딩할 수도 있다.
- [0121] 본 개시물의 양태들에 따르면, 비디오 디코더 (30) 는 독립적으로 디코딩가능한 멀티-계층 비트스트림의 비-기본 계층을 디코딩하도록 구성될 수도 있다. 예를 들어, 본 개시물의 양태들에 따르면, 비디오 디코더 (30) 는 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를, 오직 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에만 디코딩할 수도 있다.
- [0122] 일부 예시들에서, 비디오 디코더 (30) 는 poc_reset_idc 신택스 엘리먼트를 디코딩하도록 구성될 수도 있다. 본 개시물의 양태들에 따르면, 비디오 디코더 (30) 는 하나 이상의 제약들에 기초하여 poc_reset_idc 신택스 엘리먼트를 디코딩할 수도 있다. 예를 들어, 비디오 디코더 (30) 는 특정 표준에 부합하는 poc_reset_idc 신택스 엘리먼트를 디코딩할 수도 있다. 예를 들어, 본 개시물의 양태들에 따르면, 비디오 디코더 (30) 는 오직 코딩되고 있는 픽처가 0 과 동일한 POC LSB 값을 가질 경우에만 2 와 동일할 poc_reset_idc 신택스 엘리먼트를 디코딩할 수도 있다. 즉, 코딩되고 있는 특정 계층에 대한 poc_lsb_not_present_flag 가 (예컨대, 그 계층이 독립적인 비-기본 계층인 것을 표시하는) 1 과 동일하고 픽처의 POC LSB 값을 표시하는 slice_pic_order_cnt_lsb 신택스 엘리먼트가 0 보다 클 경우, 비디오 디코더 (30) 는 2 와 동일할 poc_reset_idc 의 값을 디코딩하지 않을 수도 있다.
- [0123] 비디오 디코더 (30) 가 3 과 동일한 poc_reset_idc 신택스 엘리먼트를 디코딩할 경우, 비디오 디코더 (30) 는 추가로 full_poc_reset_flag 및 poc_lsb_val 신택스 엘리먼트를 디코딩할 수도 있다. 본 개시물의 양태들에 따르면, 디코딩되고 있는 계층에 대한 poc_lsb_not_present_flag 가 1 과 동일하고, full_poc_reset_flag 가 1 과 동일할 경우, 비디오 디코더 (30) 는 0 과 동일한 poc_lsb_val 의 값을 디코딩할 수도 있다. 이러한 방식으로, 비디오 디코더 (30) 는 POC 값 리세팅이 오직 0 과 동일한 POC LSB 값을 갖는 픽처들에 대해서만 수행되는 것을 보장할 수도 있고, 이는 참조 픽처들이 도 4 에 대하여 이하 더 상세히 설명되는 것과 같은, 참조 픽처 세트에서 적절히 식별되게 할 수도 있다.
- [0124] 비디오 디코더 (30) 는 또한, 참조 픽처 리스트에 포함될 참조 픽처들의 POC 값들을 디코딩할 수도 있다. 비디오 디코더 (30) 는 슬라이스 세그먼트 헤더들에서 및/또는 PPS 또는 SPS 와 같은 파라미터 세트들로부터 이들 POC 값들을 디코딩할 수도 있다. 그 후에, 비디오 디코더 (30) 는 디코딩된 POC 값들에 의해 식별된 참조 픽처들을 포함하는 참조 픽처 리스트를 구성할 수도 있다.
- [0125] 슬라이스에 대한 참조 픽처 리스트를 구성한 후에, 비디오 디코더 (30) 는 그 슬라이스의 블록들을 디코딩할 수도 있다. 모션 보상 유닛 (72) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 파싱함으로써 현재 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 그 예측 정보를 사용하여, 디코딩되는 현재 비디오 블록에 대한 예측 블록들을 생성한다. 예를 들면, 모션 보상 유닛 (72) 은, 비디오 슬라이스의 비디오 블록들을 코딩하기 위해 사용되는 예측 모드 (예를 들면, 인트라-예측 또는 인터-예측), 인터-예측 슬라이스 타입 (예를 들면, B 슬라이스, P 슬라이스 또는 GPB 슬라이스), 슬라이스의 각각의 인터-인코딩된 비디오 블록에 대한 모션 벡터들, 슬라이스의 각각의 인터-코딩된 비디오 블록에 대한 인터-예측 상태, 및 현재의 비디오 슬라이스의 비디오 블록들을 디코딩하기 위한 다른 정보를 결정하기 위해, 수신된 신택스 엘리먼트들의 일부를 사용한다. 인터-예측된 블록에 대한 모션 정보는 블록을 예측하기 위해 사용할 참조 픽처를 참조 픽처 리스트에서 식별하기 위해 참조 픽처 리스트 식별자 및 참조 인덱스를 포함할 수도 있다.
- [0126] 모션 보상 유닛 (72) 은 또한, 보간 필터들에 기초하여 보간을 수행할 수도 있다. 모션 보상 유닛 (72) 은 비디오 블록들의 인코딩 동안 비디오 인코더 (20) 에 의해 사용된 바와 같은 보간 필터들을 이용하여, 참조 블록들의 서브-정수 픽셀들에 대한 보간된 값들을 계산할 수도 있다. 이 경우, 모션 보상 유닛 (72) 은 수신된 신택스 엘리먼트들로부터 비디오 인코더 (20) 에 의해 사용된 보간 필터들을 결정하고, 보간 필터들을 이용

하여 예측 블록들을 생성할 수도 있다.

- [0127] 역양자화 유닛 (76) 은 비트스트림에서 제공되고 엔트로피 디코딩 유닛 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화, 즉, 양자화해제한다. 역양자화 프로세스는 비디오 슬라이스에 있어서의 각각의 비디오 블록에 대해 비디오 디코더 (30) 에 의해 계산된 양자화 파라미터 (QP_V) 의 이용을 포함하여, 적용되어야 하는 양자화의 정도 및 유사하게 역양자화의 정도를 결정할 수도 있다.
- [0128] 역변환 유닛 (78) 은 픽셀 도메인에서 잔차 블록들을 생성시키기 위해, 역변환, 예를 들어, 역 DCT, 정수 역변환, 또는 개념적으로 유사한 역변환 프로세스를 변환 계수들에 적용한다.
- [0129] 모션 보상 유닛 (72) 이 모션 벡터들 및 다른 신택스 엘리먼트들에 기초하여 현재 비디오 블록에 대한 예측 블록을 생성한 이후, 비디오 디코더 (78) 는 역변환 유닛 (72) 으로부터의 잔차 블록들을 모션 보상 유닛 (30) 에 의해 생성된 대응하는 예측 블록들과 합산함으로써 디코딩된 비디오 블록을 형성한다. 합산기 (80) 는 이러한 합산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 요구된다면, 디블로킹 필터가 또한, 블록 키니스 아티팩트들을 제거하기 위해, 디코딩된 블록들을 필터링하는데 적용될 수도 있다. (코딩 루프에 있어서 또는 코딩 루프 이후에) 다른 루프 필터들이 또한 픽셀 천이들을 평활하게 하거나 그렇지 않으면 비디오 품질을 개선하기 위해 이용될 수도 있다. 그 후, 소정의 프레임 또는 픽처에 있어서의 디코딩된 비디오 블록들이 참조 픽처 메모리 (82) 에 저장되고, 이 참조 픽처 메모리는 후속적인 모션 보상을 위해 사용되는 참조 픽처들을 저장한다. 참조 픽처 메모리 (82) 는 또한, 도 1 의 디스플레이 디바이스 (32) 와 같은 디스플레이 이 디바이스 상으로의 더 나중의 프리젠테이션을 위해 디코딩된 비디오를 저장한다.
- [0130] 이러한 방식으로, 도 3 의 비디오 디코더 (30) 는, 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처의 적어도 일부분을 디코딩하고, 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를, 오직 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 갖는 경우에만 디코딩하도록 구성된 비디오 디코더의 일 예를 나타낸다.
- [0131] 도 4 는 멀티-계층 비트스트림의 독립적인 비-기본 계층에 대한 POC 값 리세팅을 도시하는 개념 다이어그램이다. 도 4 의 예에서, 멀티-계층 비트스트림은 기본 계층 (계층 0; 90) 및 독립적인 비-기본 계층 (계층 1; 92) 을 포함한다. 일반적으로, 도 4 에서 설명된 예시적인 다이어그램의 상부 절반은 예컨대, 비디오 인코더 (20) 와 같은 비디오 인코더에 의해 인코딩된 (어구 "계층 1 의 추출 이전" 으로 표시된) 것과 같이, 추출 이전의 멀티-계층 비트스트림을 도시한다. 도 4 에 설명된 예시적인 다이어그램의 하부 절반은 예컨대, 비디오 디코더 (30) 와 같은 비디오 디코더에 의한 디코딩을 위한 (어구 "계층 1 의 추출 이후" 로 표시된) 멀티-계층 비트스트림으로부터의 추출 이후에 비-기본 계층 (92) 을 포함하는 서브-비트스트림을 도시한다.
- [0132] 추출 이전에, 비디오 인코더 (20) 는 특정 비디오 코딩 표준, 예컨대 MV-HEVC 또는 SHVC 을 사용하여 기본 계층 (90) 및 비-기본 계층 (92) 을 인코딩할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 IDR 픽처 (94) 에 대하여 POC 값 리세팅을 수행하도록 구성될 수도 있다. 비디오 인코더는 2 와 동일한 POC 값 리셋 인덱스 (poc_reset_idc) 를 시그널링할 수도 있다. POC 값 리세팅 이전에, 픽처 (94) 는 0 의 POC MSB 값과 20 의 POC LSB 값을 갖는다. POC 값 리세팅 이후에, 픽처 (94) 는 0 의 POC MSB 값과 0 의 POC LSB 값을 갖는다. 추가로, POC 값들의 교차-계층 정렬을 유지하기 위해, 비디오 인코더 (20) 는 비-기본 계층 (92) 에 포함된 픽처 (96) 의 POC 값 리세팅을 수행할 수도 있다. 따라서, POC 값 리세팅 이전에, 픽처 (96) 는 0 의 POC MSB 값과 20 의 POC LSB 값을 갖는다. POC 값 리세팅 이후에, 픽처 (96) 는 0 의 POC MSB 값과 0 의 POC LSB 값을 갖는다.
- [0133] POC 값 리세팅을 수행할 때, 비디오 인코더 (20) 는 또한 POC 값 리세팅을 설명하기 위해 DPB 에 저장된 픽처들의 POC LSB 값들을 감분할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 DPB 에 저장된 각각의 참조 픽처의 POC LSB 값을 20 (예컨대, 원래의 POC LSB 값과 리셋 POC LSB 값 간의 차이) 만큼 감분할 수도 있다. 따라서, POC 값 리세팅 이전에, 참조 픽처 (98) 는 0 의 POC MSB 값과 3 의 POC LSB 값을 갖는다. POC 값 리세팅 이후에, 참조 픽처 (98) 는 0 의 POC MSB 값과 -17 의 POC LSB 값을 갖는다. 유사하게, POC 값 리세팅 이전에, 참조 픽처 (100) 는 0 의 POC MSB 값과 3 의 POC LSB 값을 갖는다. POC 값 리세팅 이후에, 참조 픽처 (100) 는 0 의 POC MSB 값과 -17 의 POC LSB 값을 갖는다.
- [0134] 비디오 인코더 (20) 는 또한 인터-예측을 사용하여 비-기본 계층 (92) 의 픽처 (102) 를 인코딩한다. 예를 들어, 비디오 인코더 (20) 는 참조 픽처 (100) 를 LTRP 로서 식별하고, 픽처 (102) (예컨대, 장기 참조 픽처 세트 (LT-RPS)) 를 인코딩하기 위해 참조 픽처 (100) 를 RPS 에 포함시킨다. 비디오 인코더 (20) 는 또한, 참

조 픽처 (100) 를 LTRP 로서 식별하기 위해 픽처 (102) 의 슬라이스의 슬라이스 헤더에서 또는 픽처 (102) 에 의해 참조되는 파라미터 세트에서 참조 픽처 (100) 의 POC LSB 값을 시그널링할 수도 있다. 즉, 비디오 인코더 (20) 는 POC 값 리세팅을 수행한 후에 참조 픽처 (100) 의 POC LSB 값을 시그널링하여, 시그널링된 POC LSB 값이 (변환 알고리즘을 사용한 -17 의 양의 수로의 변환 이후) 15 가 되도록 할 수도 있다.

[0135] 전술된 것과 같은 추출 이후에, 비디오 디코더 (30) 는 비-기본 계층을 인코딩하는데 사용된 멀티-계층 표준과 반드시 동일할 필요는 없는 특정 단일 계층 표준, 예컨대 HEVC 에 부합함으로써 비-기본 계층 (92) 을 디코딩하도록 구성될 수도 있다. 따라서, 비디오 디코더 (30) 는 비-기본 계층 (92) 을 디코딩하기 위한 표준과 연관된 신택스를 사용하고, 다른 신택스를 삭제하거나 무시할 수도 있다. HEVC 및/또는 HEVC 의 멀티-계층 확장들에 부합하는 비디오 디코더 (30) 가 본원에서 참조되지만, 본 개시물의 기술들은 이러한 방식에 반드시 제한될 필요는 없으며 다른 표준들과 함께 사용될 수도 있음이 이해되어야만 한다.

[0136] 픽처 (102) 를 디코딩하기 위해, 비디오 디코더 (30) 는 비디오 인코더 (20) 에 의해 수행되는 것과 역의 방식으로 인터-예측을 수행하도록 구성될 수도 있다. 따라서, 비디오 디코더 (30) 는 LT-RPS 로부터 하나 이상의 픽처들을 포함할 수도 있는 참조 픽처 리스트를 구성할 수도 있다. 예를 들면, 비디오 디코더 (30) 는 참조 픽처 (100) 를 포함하는 하나 이상의 장기 참조 픽처들을 식별하기 위해 픽처 (102) 로 지칭되는 파라미터 세트 또는 픽처 (102) 의 슬라이스의 슬라이스 헤더를 디코딩할 수도 있다.

[0137] 전술된 것과 같이, 참조 픽처 (100) 의 POC LSB 값은 픽처 (102) 의 슬라이스 헤더에서, 15 (예컨대, POC 값 리세팅을 수행한 후에, -17 의 POC MSB 값에 대응함) 와 동일한 것으로 시그널링될 수도 있다. 그러나, 비디오 디코더 (30) 는 POC 값 리세팅을 수행하도록 구성되지 않을 수도 있다. 예를 들어, 픽처 (96) 를 디코딩할 경우, 비디오 디코더 (30) 는 poc_reset_idc 신택스 엘리먼트를 디코딩하지 않을 수도 있고, DPB 에 저장된 픽처들의 POC LSB 값들을 감분하지 않을 수도 있다. 따라서, 참조 픽처 (100) 의 POC LSB 값은 3 과 동일하다. 그러나, 참조 픽처 (100) 는 슬라이스 헤더에서 15 의 POC LSB 값으로 식별된다. 따라서, 도 4 의 예에서, 비디오 디코더 (30) 는, 참조 픽처 (100) 의 POC LSB 값이 슬라이스 헤더에서 시그널링된 POC LSB 값과 매칭하지 않기 때문에, 참조 픽처 (100) 를 DPB 에 로케이팅할 수도 있다.

[0138] 도 4 의 예에 의해 예시된 것과 같이, 슬라이스 헤더 정보를 변경하지 않고 (비-기본 계층 (92) 과 같은) HEVC 디코더에 의해 디코딩될 수 있는 독립적인 향상 계층을 추출하기 위해, 비디오 디코더 (30) 는 LTRP들이 계층에서 픽처들의 예측을 위해 사용되는지 여부 및 POC LSB 와 POC MSB 양자를 리셋하는 픽처들이 계층에 존재하는지 여부를 고려할 수도 있다. 이들 2 개의 파라미터들은 계층에서 픽처들의 슬라이스 헤더들을 변경해야할 필요 없이 멀티-계층 비트스트림으로부터 독립적인 향상 계층을 추출하는 것이 가능한지 여부를 결정하기 위해 고려될 수도 있다.

[0139] 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 POC 값 리세팅이 독립적인 비-기본 계층들에 대하여 수행되는 방식을 제어할 수도 있다. 그 기술들은, 일부 경우들에 있어서, 슬라이스 헤더에서 시그널링된 LTRP들의 POC 값들이 비디오 디코더의 DPB 에 저장된 참조 픽처들의 POC 값들과 일치성을 유지하게 할 수도 있다. 일 예에서, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처를 코딩 (인코딩 또는 디코딩) 할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 또한, 오직 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에만 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를 코딩할 수도 있다. 그러한 경우들에서, POC 값 리셋을 수행하는 것은, POC LSB 값이 이미 0 과 동일하기 때문에, 비디오 인코더 (20) 또는 비디오 디코더 (30) 가 DPB 에 저장된 픽처들의 POC 값들을 감분할 것을 요구하지 않는다.

[0140] 본 개시물의 다양한 기술들에 따라, POC 값 리세팅을 관리하기 위한 예시적인 신택스 및 시맨틱들이 이하 설명된다. 설명되는 신택스 및 시맨틱들은 예컨대, 앞서 언급된 MV-HEVC 및 SHVC 의 신택스 및 시맨틱들과 관련된다. 예시적인 설명에서, 이하의 신택스 표들 및 시맨틱들, 즉 MV-HEVC 및 SHVC 에 대한 추가들은 *이탤릭체* 를 사용하여 표현되고, 삭제들은 "제거됨" 이 선행하는 괄호 안의 텍스트 (예컨대, [제거됨: 제거된 텍스트]) 를 사용하여 표현된다. 일반적으로, "요건들" 에 관한 언급들은 표준 또는 표준 확장의 텍스트의 부분을 형성하지만, 본 개시물의 기술들의 목적들을 위한 요건은 아닌 것으로 이해되어야 한다. 일부 경우들에서, 그러한 "요건들" 은 적용가능한 것으로 결정되고, 그 후에 예컨대, 그 결정들에 기초하여 비디오 코더에 의해 고수될 수도 있는 비트스트림 제약들을 포함할 수도 있다.

[0141] 제 1 예에서, 이하 변경들은 비디오 파라미터 세트 (VPS) 비디오 사용가능성 정보 (VUI) 에서 신택스 엘리먼트 `base_layer_parameter_set_compatibility_flag[i]` 의 시맨틱들에 수행될 수도 있다:

- [0142] 1 과 동일한 **base_layer_parameter_set_compatibility_flag[i]** 는 하기의 제약들을 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 를 갖는 계층에 적용하는 것을 명시한다. 0 과 동일한 **base_layer_parameter_set_compatibility_flag[i]** 는 하기의 제약들을 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 를 갖는 계층에 적용할 수도 있거나 적용하지 않을 수도 있는 것을 명시한다. 존재하지 않을 경우, **base_layer_parameter_set_compatibility_flag[i]** 의 값은 0 과 동일한 것으로 추론된다.
- [0143] - VPS 를 지칭하는 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값을 갖는 각각의 코딩된 슬라이스 세그먼트 NAL 유닛은 0 과 동일한 **nuh_layer_id** 값을 갖는 PPS 를 지칭할 것이다.
- [0144] - VPS 를 지칭하는 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값을 갖는 각각의 코딩된 슬라이스 세그먼트 NAL 유닛은 0 과 동일한 **nuh_layer_id** 값을 갖는 SPS 를 지칭할 것이다.
- [0145] - **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값에 대한 활성 SPS 의 각각 **chroma_format_idc**, **separate_colour_plane_flag**, **pic_width_in_luma_samples**, **pic_height_in_luma_samples**, **bit_depth_luma_minus8**, 및 **bit_depth_chroma_minus8** 의 값들은 활성 VPS 에서 **vps_rep_format_idx[i]-th rep_format()** 의 각각 **chroma_format_idc**, **separate_colour_plane_flag**, **pic_width_in_luma_samples**, **pic_height_in_luma_samples**, **bit_depth_luma_minus8**, 및 **bit_depth_chroma_minus8** 의 값들과 동일할 것이다.
- [0146] - **poc_lsb_not_present_flag[i]** 의 값은 1 과 동일할 것이다.
- [0147] - VPS 를 지칭하는 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 를 갖는 픽처 **picA** 를 포함하는 액세스 유닛이 또한 0 과 동일한 **nuh_layer_id** 및 0 보다 큰 **slice_pic_order_cnt_lsb** 을 갖는 픽처를 포함할 경우, 픽처 **picA** 는 비-IDR 픽처일 것이다.
- [0148] **num_long_term_sps** 또는 **num_long_term_pics**, 또는 이들 양자의 비-제로 값을 갖는 VPS 를 지칭하는 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값을 갖는 코딩된 슬라이스 세그먼트 NAL 유닛, 및 **slice_pic_order_cnt_lsb** 이 0 과 동일하지 않을 경우, **poc_reset_idc** 는 2 와 동일하지 않을 것이다.
- [0149] 제 2 예에서, **base_layer_parameter_set_compatibility_flag[i]** 대 **base_layer_compatibility_flag[i]** 의 선택스 엘리먼트의 명칭. 추가로, 이하 변경들은 (하기에서 **base_layer_compatibility_flag[i]** 로 표시되는) VPS VUI 에서 선택스 엘리먼트 **base_layer_parameter_set_compatibility_flag[i]** 의 시맨틱들에 수행될 수도 있다:
- [0150] 1 과 동일한 **base_layer_compatibility_flag[i]** 는 하기의 제약들을 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 에 적용하는 것을 명시한다. 0 과 동일한 **base_layer_compatibility_flag[i]** 는 하기의 제약들을 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 를 갖는 계층에 적용할 수도 있거나 적용하지 않을 수도 있는 것을 명시한다. 존재하지 않을 경우, **base_layer_compatibility_flag[i]** 의 값은 0 과 동일한 것으로 추론된다.
- [0151] - VPS 를 지칭하는 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값을 갖는 각각의 코딩된 슬라이스 세그먼트 NAL 유닛은 0 과 동일한 **nuh_layer_id** 값을 갖는 PPS 를 지칭할 것이다.
- [0152] - VPS 를 지칭하는 **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값을 갖는 각각의 코딩된 슬라이스 세그먼트 NAL 유닛은 0 과 동일한 **nuh_layer_id** 값을 갖는 SPS 를 지칭할 것이다.
- [0153] - **layer_id_in_nuh[i]** 과 동일한 **nuh_layer_id** 값을 갖는 계층에 대한 활성 SPS 의 각각 **chroma_format_idc**, **separate_colour_plane_flag**, **pic_width_in_luma_samples**, **pic_height_in_luma_samples**, **bit_depth_luma_minus8**, 및 **bit_depth_chroma_minus8** 의 값들은 활성 VPS 에서 **vps_rep_format_idx[i]-th rep_format()** 의 각각 **chroma_format_idc**, **separate_colour_plane_flag**, **pic_width_in_luma_samples**, **pic_height_in_luma_samples**, **bit_depth_luma_minus8**, 및 **bit_depth_chroma_minus8** 의 값들과 동일할 것이다.
- [0154] - **poc_lsb_not_present_flag[i]** 의 값은 1 과 동일할 것이다.
- [0155] - **layer_id_in_nuh[i]** 와 동일한 **nuh_layer_id** 를 가지고 VPS 를 지칭하는 임의의 픽처 **picA** 에 대하여, 이하 내용들이 적용된다:
- [0156] - **slice_pic_order_cnt_lsb** 가 0 보다 클 경우, **poc_reset_idc** 는 2 와 동일하지 않을 것이다.
- [0157] - **full_poc_reset_flag** 이 1 과 동일할 경우, **poc_lsb_val** 는 0 과 동일할 것이다.

[0158] 본원에 포함된 신택스는 단지 예시의 목적들을 위한 것이다. 즉, 앞의 예는 base_layer_compatibility 플래그에 대하여 설명되지만, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 다른 상이한 신택스 엘리먼트에 대하여 또는 다른 방식으로 본원에서 설명된 제약들을 적용할 수도 있다.

[0159] 제 3 예에서, poc_lsb_not_present_flag[i] 의 시맨틱들에 제약이 추가되어, 0 과 동일한 nuh_layer_id 를 갖는 비-IDR 픽처 및 layer_id_in_nuh[i] 와 동일한 nuh_layer_id 를 갖는 IDR 픽처를 포함하는 적어도 하나의 액세스 유닛이 존재할 경우, poc_lsb_not_present_flag[i] 이 1 과 동일하지 않게 한다. 신택스 엘리먼트 base_layer_parameter_set_compatibility_flag[i] 의 시그널링에 조건이 추가되어, 신택스 엘리먼트는 오직 poc_lsb_not_present_flag[i] 의 값이 1 과 동일할 경우에만 존재하게 된다. base_layer_parameter_set_compatibility_flag[i] 에 대한 값은, 그 값이 존재하지 않을 경우 0 과 동일한 것으로 추론된다.

[0160] 앞서 언급된 변경을 갖는 신택스 엘리먼트 base_layer_parameter_set_compatibility_flag[i] 의 시그널링은 이하 표에서 설명되는 것과 같이 수정될 수도 있다:

vps_vui(){	디스크립터
...	
for(i = 1; i <= MaxLayersMinus1; i++)	
if(!poc_lsb_not_present_flag[i])	
base_layer_parameter_set_compatibility_flag[i]	u(1)
...	
}	

[0161]

[0162] 상기 표에서, 1 과 동일한 poc_lsb_not_present_flag[i] 는, slice_pic_order_cnt_lsb 신택스 엘리먼트가 CVS 에서 layer_id_in_nuh[i] 와 동일한 nuh_layer_id 를 갖는 IDR 픽처들의 슬라이스 헤더들에 존재하지 않는 것을 명시한다. 추가로, 0 과 동일한 poc_lsb_not_present_flag[i] 는, slice_pic_order_cnt_lsb 신택스 엘리먼트가 CVS 에서 layer_id_in_nuh[i] 와 동일한 nuh_layer_id 를 갖는 IDR 픽처들의 슬라이스 헤더들에 존재할 수도 있거나 존재하지 않을 수도 있는 것을 명시한다. 존재하지 않을 경우, poc_lsb_not_present_flag[i] 의 값은 0 과 동일한 것으로 추론된다.

[0163] 추가로, 1 과 동일한 base_layer_parameter_set_compatibility_flag[i] 는 하기의 제약들을 layer_id_in_nuh[i] 과 동일한 nuh_layer_id 를 갖는 계층에 적용하는 것을 명시한다. 0 과 동일한 base_layer_parameter_set_compatibility_flag[i] 는 하기의 제약들을 layer_id_in_nuh[i] 과 동일한 nuh_layer_id 를 갖는 계층에 적용할 수도 있거나 적용하지 않을 수도 있는 것을 명시한다. 일반적으로, base_layer_parameter_set_compatibility_flag[i] 는 항상 계층이 기본 계층의 파라미터 세트들 (예컨대, PPS 및 SPS) 을 사용할지 여부를 표시할 수도 있다.

[0164] 제 4 예에서, 본 개시물의 양태들에 따라, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 독립적인 계층이 그 계층에서 픽처들의 슬라이스 헤더들에서의 정보를 업데이트할 필요 없이 추출되고 HEVC 부합 디코더에 의해 디코딩될 수도 있는지 여부를 표시하는, 각각의 계층에 대한 새로운 플래그를 구현하도록 구성될 수도 있다. 그러한 플래그와 연관된 시그널링은 이하 표 1 에서 설명되는 것과 같을 수도 있다:

표 1

vps_vui(){	디스크립터
...	
for(i = 1; i <= MaxLayersMinus1; i++)	
if(NumDirectRefLayers[layer_id_in_nuh[i]] == 0)	
slice_segment_header_update_not_required_for_extraction_flag[i]	u(1)
...	
}	

[0165]

[0166]

앞의 표 1의 예에서, 1과 동일한 slice_segment_header_update_not_required_for_extraction_flag[i]는, VPS를 지칭하는 각각의 CVS에서 layer_id_in_nuh[i]와 동일한 nuh_layer_id를 갖는 픽처들의 슬라이스 세그먼트 헤더들이 서브-비트스트림 추출 프로세스가 입력 TargetDecLayerIdList에서의 최저 계층이 되는 layer_id_in_nuh[i]와 동일한 nuh_layer_id를 갖는 계층으로 인보크될 경우, 업데이트될 필요가 없는 것을 명시한다. 추가로, 0과 동일한 slice_segment_header_update_not_required_for_extraction_flag[i]는, CVS에서 layer_id_in_nuh[i]와 동일한 nuh_layer_id를 갖는 픽처들의 슬라이스 세그먼트 헤더들이 서브-비트스트림 추출 프로세스가 입력 TargetDecLayerIdList에서의 최저 계층이 되는 layer_id_in_nuh[i]와 동일한 nuh_layer_id를 갖는 계층으로 인보크될 경우, 업데이트될 필요가 있을 수도 있거나 없을 수도 있는 것을 명시한다. 존재하지 않을 경우, slice_segment_header_update_not_required_for_extraction_flag[i]는 0과 동일한 것으로 추론된다.

[0167]

도 5는 멀티-계층 비트스트림의 독립적인 비-기본 계층에 대한 POC 값 리세팅을 도시하는 다른 개념 다이어그램이다. 도 5의 예에서, 멀티-계층 비트스트림은 기본 계층(계층 0; 110) 및 독립적인 비-기본 계층(계층 1; 112)을 포함한다. 기본 계층(110)은 IDR 픽처(114) 및 픽처(116)를 포함한다. 비-기본 계층(112)은 기본 계층(110)의 픽처(114)와 동일한 액세스 유닛에 있는 픽처(118) 및 기본 계층(110)의 픽처(116)와 동일한 액세스 유닛에 있는 IDR 픽처(120)를 포함한다.

[0168]

비디오 디코더(30)와 같은 비디오 디코더는 특정 비디오 코딩 표준, 예컨대 MV-HEVC 또는 SHVC를 사용하여 기본 계층(110) 및 비-기본 계층(112)을 디코딩하도록 구성될 수도 있다. 비디오 디코더(30)는 픽처(114)가 IDR 픽처이기 때문에, 기본 계층(110)의 픽처(114)에 대하여 POC 값 리세팅을 수행하도록 구성될 수도 있다. 비디오 디코더는 픽처(114)에 대하여 2와 동일한 POC 값 리셋 인덱스(poc_reset_idc)를 디코딩할 수도 있다. 추가로, POC 값들의 교차-계층 정렬을 유지하기 위해, 비디오 디코더(30)는 또한, 비-기본 계층(112)에 포함된 픽처(118)의 POC 값 리세팅을 수행할 수도 있다. 따라서, POC 값 리세팅 이후에, 기본 계층(110)의 픽처(114)와 비-기본 계층(112)의 픽처(118)는 0과 동일한 POC LSB 값을 갖는다.

[0169]

SHVC 및 MV-HEVC에 포함된 것과 같은 신택스 엘리먼트 poc_lsb_not_present_flag[i]는, POC LSB 정보가 비-기본 계층(112)과 같은 독립적인 향상 계층의 IDR 픽처에 속하는 슬라이스들의 슬라이스 세그먼트 헤더들에 포함되지 않게 한다. 신택스 엘리먼트는 비-기본 계층(112)이 그 계층에서 픽처들의 슬라이스 세그먼트 헤더들을 수정하지 않고 오직 NAL 유닛 헤더들에서 nuh_layer_id만을 변경시킴으로써, 멀티-계층 비트스트림으로부터 추출되게 하고(예컨대, MV-HEVC 또는 SHVC에 대하여 구성된 디코더보다) HEVC 디코더에 의해 디코딩되게 하는 것을 도울 수도 있다. 예를 들어, SHVC 및 MV-HEVC의 워킹 드래프트들에서 현재 설명되고 있는 것과 같이, 1과 동일한 poc_lsb_not_present_flag는 POC LSB 값이 IDR 픽처들의 슬라이스 헤더들에 존재하지 않는 것을 명시한다. 0과 동일한 poc_lsb_not_present_flag는 POC LSB 값이 IDR 픽처들의 슬라이스 헤더들에 존재할 수도 있거나 존재하지 않을 수도 있는 것을 명시한다.

[0170]

앞서 언급된 MV-HEVC 및 SHVC에 포함된 poc_lsb_not_present_flag[i]에 대한 오직 명시적인 제약은, poc_lsb_not_present_flag[i]의 값이 비-기본 계층에 대하여, 오직 그 계층이 독립적인(예컨대, 계층간 예

측이 그 계층의 비디오 데이터를 예측하는데 사용되지 않는) 경우에만 1 로 세팅될 수도 있다는 것이다. 그러나, POC LSB 값이 비-기본 계층 (112) 과 같은 독립적인 비-기본 계층의 슬라이스 헤더에 포함되지 않는 것을 표시하는 것은, 0 의 POC LSB 값을 가지지 않는 픽처들에 대하여 문제를 생성할 수도 있다.

[0171] 예를 들어, 도 5 의 예에 도시된 것과 같이, 비-기본 계층 (112) 이 1 과 동일하게 세팅된 poc_lsb_not_present_flag[i] 과 연관되어, POC LSB 값이 HEVC 헤더와의 부합성을 유지하기 위해 IDR 픽처들의 슬라이스 헤더들에 포함되지 않을 수도 있게 한다. 픽처 (120) 는 POC 값 리세팅이 수행되는 (예컨대, poc_reset_idc 이 1 과 동일한) IDR 픽처이다. 따라서, 픽처 (120) 의 슬라이스들의 슬라이스 헤더들은 POC LSB 값의 표시를 포함하지 않을 수도 있다. 비디오 디코더 (30) 는 픽처 (120) 의 POC LSB 값이 0 과 동일한 것을 추론 (예컨대, 자동으로 결정) 할 수도 있다.

[0172] 그러나, 픽처 (120) 의 POC LSB 값이 0 과 동일한 것을 추론하는 것은, (비-기본 계층 (112) 의 픽처 (120) 와 동일한 액세스 유닛에 있는) 기본 계층 (110) 의 픽처 (116) 의 POC LSB 값이 0 과 동일하지 않기 때문에, 멀티-계층 비트스트림이 HEVC 표준에 부합하지 않게 할 수도 있다. 오히려, 기본 계층 (110) 의 픽처 (116) 의 POC LSB 값은 4 이다. 따라서, 픽처들 (116 및 120) 을 포함하는 액세스 유닛은 비트스트림 부합에 대하여 허가되지 않은, 상이한 POC 값들을 갖는 픽처들을 포함한다.

[0173] 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를, 오직 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에만 코딩할 수도 있다. 예를 들면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는, 계층에서의 IDR 픽처의 POC LSB 값이 0 과 동일한 것으로 추론될 수도 있지 않다면, poc_lsb_not_present_flag [i] 의 값이 (예컨대 POC LSB 값이 슬라이스 헤더에 포함되지 않는 것을 표시하는) 1 과 동일하지 않도록 제약을 적용한다. 독립적인 비-기본 계층 픽처의 POC LSB 값은, 동일한 액세스 유닛의 기본 계층 픽처의 POC LSB 값이 IDR 픽처일 경우 또는 비-기본 계층 픽처의 POC LSB 값이 0 의 값일 (예컨대, POC LSB 값들의 정상적인 진행 이후에 0 과 동일하게 롤 오버할) 경우에, 0 과 동일한 것으로 추론될 수도 있다.

[0174] 도 5 의 예에서, 본 개시물의 양태들에 따라, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 IDR 픽처인 픽처 (120) 의 POC 값 리세팅이 0 과 동일한 것으로 추론될 수 없기 때문에, 비-기본 계층 (112) 에 대하여 1 과 동일한 poc_lsb_not_present_flag [i] 을 코딩하지 않을 수도 있다. 반대로, 픽처 (120) 의 POC LSB 값은 4 이다. 0 과 동일한 poc_lsb_not_present_flag 는 비-기본 계층 (112) 이 픽처 (120) 의 슬라이스 헤더에 포함된 선택스 엘리먼트들과 같은 하나 이상의 선택스 엘리먼트들을 수정하지 않고, 예컨대 HEVC 부합 비디오 디코더를 사용하여 독립적으로 디코딩되지 않을 수 있는 것을 표시할 수도 있다.

[0175] 이러한 방식으로, 본 개시물의 양태들에 따라, poc_lsb_not_present_flag[i] 의 값은, 멀티-계층 비트스트림이 비-제로 POC LSB 값을 갖는 비-IDR 픽처인 기본 계층 픽처 및 IDR 픽처인 (VPS 에서 i 번째 계층에 대응하는) 향상 계층 픽처를 가지는 하나 이상의 액세스 유닛들을 포함할 경우, 1 과 동일하지 않도록 제약될 수도 있다. 일 예에서, 이하 제약이 MV-HEVC 및 SHVC 의 poc_lsb_not_present_flag[i] 의 시맨틱들의 부분으로서 부가되며 (예컨대, poc_lsb_not_present_flag[i] 의 시맨틱에 대한 주의로서), 여기서 추가들은 이탤릭체를 사용하여 표현된다:

[0176] *비트스트림 부합의 조건은, VPS 를 지칭하는 모든 CVS들 내에서, 0 과 동일한 nuh_layer_id 및 0 보다 큰 slice_pic_order_cnt_lsb 를 갖는 비-IDR 픽처를 포함하고 layer_id_in_nuh[i] 와 동일한 nuh_layer_id 를 갖는 IDR 픽처를 포함하는 적어도 하나의 액세스 유닛이 존재할 경우, poc_lsb_not_present_flag[i] 이 1 과 동일하지 않을 것이라는 점이다.*

[0177] 다른 예에서, 이하 제약이 MV-HEVC 및 SHVC 의 poc_lsb_not_present_flag[i] 의 시맨틱들의 부분으로서 부가되며 (예컨대, poc_lsb_not_present_flag[i] 의 시맨틱에 대한 주의로서), 여기서 추가들은 이탤릭체를 사용하여 표현된다:

[0178] *비트스트림 부합의 조건은, VPS 를 지칭하는 모든 CVS들 내에서, 0 과 동일한 nuh_layer_id 를 갖는 비-IDR 픽처 및 layer_id_in_nuh[i] 와 동일한 nuh_layer_id 를 갖는 IDR 픽처를 포함하는 적어도 하나의 액세스 유닛이 존재할 경우, poc_lsb_not_present_flag[i] 이 1 과 동일하지 않을 것이라는 점이다.*

[0179] 다른 예에서, 이하 제약이 MV-HEVC 및 SHVC 의 poc_lsb_not_present_flag[i] 의 시맨틱들의 부분으로서 부가되며 (예컨대, poc_lsb_not_present_flag[i] 의 시맨틱에 대한 주의로서), 여기서 추가들은 이탤릭체를 사용하여 표현된다:

- [0180] 비트스트림 부합의 요건은, `poc_lsb_not_present_flag[i]` 이 1 과 동일할 경우, VPS 를 지칭하는 모든 CVS들 내에서, 액세스 유닛이 `layer_id_in_nuh[i]` 와 동일한 `nuh_layer_id` 를 갖는 IDR 픽처를 포함하는 적어도 하나의 액세스 유닛이 존재할 경우, `poc_reset_idc` 이 액세스 유닛에서 모든 픽처들에 대하여 2 와 동일하지 않다면, `slice_pic_order_cnt_lsb` 의 값이 액세스 유닛에서의 모든 픽처들에 대하여 0 과 동일해야 한다는 점이다.
- [0181] 도 6 은 본 개시물의 기술들에 따른 멀티-계층 비트스트림의 비디오 데이터를 인코딩하는 예시적인 방법을 도시한 플로우차트이다. 추가로, 도 6 의 방법은 현재 블록을 인코딩하는 것을 포함한다. 현재 블록은 현재 CU 또는 현재 CU 의 일부를 포함할 수도 있다. 비디오 인코더 (20; 도 1 및 도 2) 에 대하여 설명되었지만, 다른 디바이스들이 도 6 의 방법과 유사한 방법을 수행하도록 구성될 수도 있음을 이해해야 한다.
- [0182] 이러한 예에서, 비디오 인코더 (20) 는 현재 블록을 포함하는 픽처에 대한 POC 값을 리셋할지 여부를 결정할 수도 있다 (130). POC 값 리세팅이 수행될 경우, 비디오 인코더 (20) 는 POC 리세팅 타입을 결정하고 픽처의 POC 값을 리셋할 수도 있다 (132). 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 는 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를, 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에 오직 인코딩할 수도 있다.
- [0183] 예를 들어, 도 2 에 대하여 앞서 설명된 것과 같이, 비디오 인코더 (20) 는 `poc_reset_idc` 선택스 엘리먼트를 인코딩할 수도 있다. 본 개시물의 양태들에 따르면, 비디오 인코더 (20) 는 `poc_reset_idc` 값이 하나 이상의 제약들에 기초하여 인코딩되는 방식을 제어할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 오직 코딩되고 있는 픽처가 0 과 동일한 POC LSB 값을 가질 경우에만 2 와 동일한 `poc_reset_idc` 선택스 엘리먼트를 인코딩할 수도 있다. 즉, 코딩되고 있는 특정 계층에 대한 `poc_lsb_not_present_flag` 가 (예컨대, 그 계층이 독립적인 비-기본 계층인 것을 표시하는) 1 과 동일하고 픽처의 POC LSB 값을 표시하는 `slice_pic_order_cnt_lsb` 선택스 엘리먼트가 0 보다 클 경우, 비디오 인코더 (20) 는 `poc_reset_idc` 의 값을 2 와 동일한 것으로 세팅하지 않을 수도 있다.
- [0184] 비디오 인코더 (20) 가 3 과 동일하도록 `poc_reset_idc` 선택스 엘리먼트를 인코딩할 경우, 비디오 인코더 (20) 는 추가로 `full_poc_reset_flag` 및 `poc_lsb_val` 선택스 엘리먼트를 인코딩할 수도 있다. 본 개시물의 양태들에 따르면, 인코딩되고 있는 계층에 대한 (인코딩되고 있는 계층이 독립적인 비-기본 계층인 것을 표시할 수도 있는) `poc_lsb_not_present_flag` 가 1 과 동일하고, `full_poc_reset_flag` 가 1 과 동일할 경우, 비디오 인코더 (20) 는 `poc_lsb_val` 의 값을 0 과 동일하게 세팅할 수도 있다. 이러한 방식으로, 비디오 인코더 (20) 는 POC 값 리세팅이 오직 0 과 동일한 POC LSB 값을 갖는 픽처들에 대해서만 수행되는 것을 보장할 수도 있고, 이는 참조 픽처들이 본원에서 설명되는 것과 같이, 적절히 식별되게 할 수도 있다.
- [0185] 그 후, 비디오 인코더 (20) 는 참조 픽처 리스트를 형성할 수도 있다 (134). 비디오 인코더 (20) 는 추가로, 참조 픽처 리스트에 포함된 참조 픽처들의 POC 값들을 시그널링한다 (136). 예를 들어, 비디오 인코더 (20) 는 SPS 또는 PPS 와 같은 파라미터 세트에서 및/또는 슬라이스에 대한 슬라이스 헤더에서 참조 픽처들에 대한 POC 값들 (또는 POC 값들의 일부들, 예컨대 POC LSB 값들) 을 인코딩할 수도 있다. 일부 참조 픽처들 (예컨대, 장기 참조 픽처들) 은 파라미터 세트에서 시그널링될 수도 있는 반면, 다른 참조 픽처들 (예컨대, 단기 참조 픽처들) 은 슬라이스 헤더에서 시그널링될 수도 있다.
- [0186] 예컨대, 현재 슬라이스의 블록들에 대한 최적 레이트-왜곡 특징들을 산출하는 참조 픽처들의 세트를 결정하기 위해, 참조 픽처 리스트를 형성하고 어떤 픽처들이 참조 픽처 리스트에 포함되는지를 시그널링하는 단계들은 몇몇 상이한 인코딩 패스들에 걸쳐 다수 회 수행될 수도 있음이 이해되어야 한다. 즉, 비디오 인코더 (20) 는 현재 슬라이스에서의 모든 블록들의 특징들에 기초하고 단일 블록의 개별 특징들에 대해서는 기초하지 않고, 참조 픽처 리스트에 포함된 참조 픽처들의 세트를 선택할 수도 있다.
- [0187] 그 후에, 비디오 인코더 (20) 는 현재 슬라이스의 현재 블록을 예측할 수도 있다 (138). 예를 들어, 비디오 인코더 (20) 는 현재 블록에 대하여 하나 이상의 예측 유닛들 (PU들) 을 계산할 수도 있다. 더 구체적으로, 모션 추정 유닛 (42) 은 예컨대, SAD, SSD, MAD, MSD, 또는 다른 에러 계산 메트릭들에 기초하여, 참조 블록으로서 사용된 매칭하는 블록을 식별하기 위해 참조 픽처 리스트의 참조 픽처들 중에서 현재 블록에 대한 모션 탐색을 수행할 수도 있다. 모션 추정 유닛 (42) 은 모션 탐색에 기초하여 현재 블록에 대한 모션 벡터를 생성할 수도 있고, 비디오 인코더 (20) 는 그 후, 블록에 대한 모션 정보를 인코딩할 수도 있으며 (140), 이는 모두 함께 참조 블록을 식별하는 모션 벡터에 대한 모션 벡터 차이 값들, 참조 픽처 리스트 식별자, 및 참조 인덱스를 포함할 수도 있다.

- [0188] 그 후, 비디오 인코더 (20) 는 예컨대, 변환 유닛 (TU) 을 생성하기 위해 현재 블록에 대한 잔차 블록을 계산할 수도 있다 (142). 잔차 블록을 계산하기 위해, 비디오 인코더 (20) 는 원래의 코딩되지 않은 블록과 현재 블록에 대한 예측된 블록 간의 차이를 계산할 수도 있다. 그 후, 비디오 인코더 (20) 는 잔차 블록의 계수들을 변환 및 양자화할 수도 있다 (144). 다음으로, 비디오 인코더 (20) 는 잔차 블록의 양자화된 변환 계수들을 스캔할 수도 있다 (146). 스캔 동안, 또는 스캔 이후에, 비디오 인코더 (20) 는 계수들을 엔트로피 인코딩할 수도 있다 (148). 예를 들어, 비디오 인코더 (20) 는 CAVLC 또는 CABAC 를 사용하여 계수들을 인코딩할 수도 있다. 비디오 인코더 (20) 는 그 후, 블록의 엔트로피 코딩된 데이터를 출력할 수도 있다 (150).
- [0189] 이러한 방식으로, 도 6 의 방법은, 비디오 인코더에 의해, 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처의 적어도 일부분을 인코딩하는 것, 및 비디오 코더에 의해, 픽처의 픽처 순서 카운트 (POC) 최하위 비트들 (LSB) 값에 대한 POC 값 리셋을 표시하는 데이터를, 오직 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 갖는 경우에만 인코딩하는 것을 포함하는 방법의 일 예를 나타낸다.
- [0190] 도 7 은 본 개시물의 기술들에 따른 멀티-계층 비트스트림의 비디오 데이터를 디코딩하는 예시적인 방법을 도시한 플로우차트이다. 추가로, 도 7 의 방법은 현재 블록을 디코딩하는 것을 포함한다. 현재 블록은 현재 CU 또는 현재 CU 의 일부를 포함할 수도 있다. 비디오 디코더 (30; 도 1 및 도 3) 에 대하여 설명되었지만, 다른 디바이스들이 도 7 의 방법과 유사한 방법을 수행하도록 구성될 수도 있음을 이해해야 한다.
- [0191] 이러한 예에서, 비디오 디코더 (30) 는 현재 블록을 포함하는 픽처에 대한 POC 값을 리셋할지 여부를 결정할 수도 있다 (160). POC 값 리세팅이 수행될 경우, 비디오 디코더 (30) 는 POC 리세팅을 표시하는 데이터를 디코딩할 수도 있다 (162). 일부 예시들에서, 비디오 디코더 (30) 는 POC 리세팅 타입을 표시하는 데이터를 디코딩할 수도 있다. 본 개시물의 양태들에 따르면, 비디오 디코더 (30) 는 픽처의 POC LSB 값에 대한 POC 값 리셋을 표시하는 데이터를, 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 가질 경우에 오직 인코딩할 수도 있다.
- [0192] 예를 들어, 도 3 에 대하여 앞서 설명된 것과 같이, 비디오 디코더 (30) 는 poc_reset_idc 신택스 엘리먼트를 디코딩할 수도 있다. 본 개시물의 양태들에 따르면, 비디오 디코더 (30) 는 하나 이상의 제약들에 기초하여 poc_reset_idc 를 수신 및 디코딩할 수도 있다. 예를 들어, 비디오 디코더 (30) 는 오직 코딩되고 있는 픽처가 0 과 동일한 POC LSB 값을 가질 경우에만 2 의 값을 갖는 poc_reset_idc 신택스 엘리먼트를 디코딩할 수도 있다. 즉, 코딩되고 있는 특정 계층에 대한 poc_lsb_not_present_flag 가 (예컨대, 그 계층이 독립적인 비-기본 계층인 것을 표시하는) 1 과 동일하고 픽처의 POC LSB 값을 표시하는 slice_pic_order_cnt_lsb 신택스 엘리먼트가 0 보다 클 경우, 비디오 디코더 (30) 는 2 와 동일한 poc_reset_idc 의 값을 디코딩하지 않을 수도 있다.
- [0193] 비디오 디코더 (30) 가 3 과 동일한 poc_reset_idc 신택스 엘리먼트를 디코딩할 경우, 비디오 디코더 (30) 는 추가로 full_poc_reset_flag 및 poc_lsb_val 신택스 엘리먼트를 디코딩할 수도 있다. 본 개시물의 양태들에 따르면, 인코딩되고 있는 계층에 대한 (인코딩되고 있는 계층이 독립적인 비-기본 계층인 것을 표시할 수도 있는) poc_lsb_not_present_flag 가 1 과 동일하고, full_poc_reset_flag 가 1 과 동일할 경우, 비디오 디코더 (30) 는 0 과 동일한 poc_lsb_val 의 값을 디코딩할 수도 있다. 이러한 방식으로, 비디오 디코더 (30) 는 POC 값 리세팅이 오직 0 과 동일한 POC LSB 값을 갖는 픽처들에 대해서만 수행되는 것을 보장할 수도 있고, 이는 참조 픽처들이 본원에서 설명되는 것과 같이, 적절히 식별되게 할 수도 있다.
- [0194] 그 후에, 비디오 디코더 (30) 는 픽처의 POC 값을 리셋하고 예컨대, 슬라이스 세그먼트 헤더 및/또는 PPS 또는 SPS 와 같은 파라미터 세트로부터 참조 픽처들의 POC 값들을 디코딩한다 (164). 비디오 디코더 (30) 는 그 후, 참조 픽처 리스트를 형성한다 (166).
- [0195] 다음으로, 비디오 디코더 (30) 는 현재 블록에 대한 모션 정보를 디코딩한다 (168). 모션 정보가 예컨대, 참조 픽처 리스트 식별자 및 참조 인덱스를 참조 픽처 리스트 내로 포함할 수도 있다. 그 후에, 비디오 디코더 (30) 는 예컨대, 현재 블록에 대한 예측 블록을 계산하기 위해 인터-예측 모드를 사용하여 현재 블록을 예측한다 (170). 더 구체적으로, 비디오 디코더 (30) 는 어떤 참조 픽처 리스트를 사용할지를 식별하기 위해 참조 픽처 리스트 식별자를 사용하고, 참조 픽처 리스트에서 참조 인덱스를 식별하기 위해 참조 인덱스를 사용한다. 그 후, 비디오 디코더 (30) 는 현재 블록에 대한 모션 벡터를 디코딩하고, 식별된 참조 픽처에서 참조 블록을 식별한다.

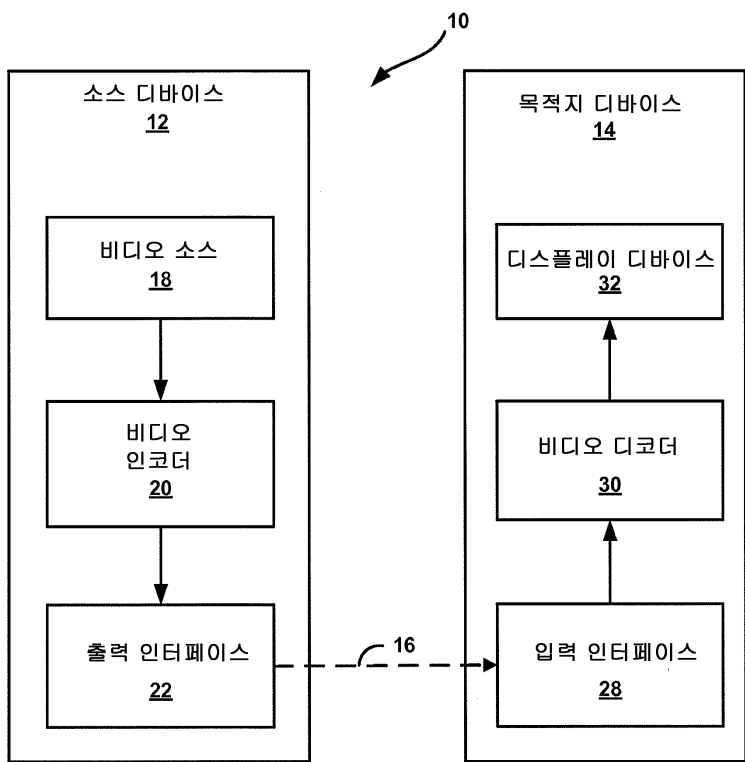
- [0196] 비디오 디코더 (30) 는 또한, 현재 블록에 대응하는 잔차 블록의 계수들에 대한 엔트로피 코딩된 데이터와 같은, 현재 블록에 대한 엔트로피 코딩된 데이터를 수신할 수도 있다 (172). 비디오 디코더 (30) 는 잔차 블록의 계수들을 재생하기 위해 엔트로피 코딩된 데이터를 엔트로피 디코딩할 수도 있다 (174). 그 후에, 비디오 디코더 (30) 는 양자화된 변환 계수들을 블록을 생성하기 위해, 재생된 계수들을 역 스캔할 수도 있다 (176). 즉, 역 스캔을 사용하여, 비디오 디코더 (30) 는 1 차원 벡터를 2 차원 행렬로 변환한다. 그 후, 비디오 디코더 (30) 는 계수들을 역양자화 및 역변환하여 잔차 블록을 생성할 수도 있다 (178). 비디오 디코더 (30) 는 예측된 블록과 잔차 블록을 결합함으로써 현재 블록을 결과적으로 디코딩할 수도 있다 (180).
- [0197] 이러한 방식으로, 도 7 의 방법은, 비디오 디코더에 의해, 멀티-계층 비트스트림의 독립적으로 디코딩가능한 계층의 픽처의 적어도 일부분을 디코딩하는 것, 및 비디오 디코더에 의해, 픽처의 픽처 순서 카운트 (POC) 최하위 비트들 (LSB) 값에 대한 POC 값 리셋을 표시하는 데이터를, 오직 그 픽처가 0 과 동일한 픽처에 대한 POC LSB 값을 갖는 경우에만 디코딩하는 것을 포함하는 방법의 일 예를 나타낸다.
- [0198] 예에 의존하여, 본 명세서에서 설명된 기술들의 임의의 특정 작동들 또는 이벤트들은 상이한 시퀀스로 수행될 수 있고, 전체적으로 부가되거나 병합되거나 또는 제거될 수도 있음 (예를 들어, 설명된 모든 작동들 또는 이벤트들이 그 기술들의 실시를 위해 필수적인 것은 아님) 이 인식되어야 한다. 더욱이, 특정 실시예들에 있어서, 행위들 또는 이벤트들은 순차적인 것보다는, 예를 들어, 멀티-스레드 프로세싱, 인터럽트 프로세싱 또는 다중의 프로세서들을 통해 동시에 수행될 수도 있다.
- [0199] 하나 이상의 실시예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되면, 상기 기능들은 컴퓨터 판독가능한 매체 상에서 하나 이상의 명령들 또는 코드로서 저장되거나 전송되며 하드웨어 기반의 처리 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능한 매체들은, 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체들, 또는 예를 들어, 통신 프로토콜에 따라 일 장소에서 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이러한 방식으로, 컴퓨터 판독가능 매체는 일반적으로 (1) 비-일시적인 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 캐리어파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는, 본 개시에서 설명된 기술들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위한 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수도 있다.
- [0200] 비-제한적인 예로서, 이러한 컴퓨터 판독가능 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 스토리지, 자기 디스크 스토리지, 또는 다른 자기 스토리지 디바이스들, 플래시 메모리, 또는 요구되는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하기 위해 사용될 수 있으며 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속물이 컴퓨터 판독가능 매체로 적절히 칭해진다. 예를 들면, 명령들이 동축 케이블, 광섬유 케이블, 연선, 디지털 가입자 회선 (DSL), 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들을 이용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 송신된다면, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 매체의 정의 내에 포함된다. 하지만, 컴퓨터 판독가능 저장 매체 및 데이터 저장 매체는 커넥션들, 캐리어파들, 신호들, 또는 다른 일시적 매체들을 포함하지 않지만 대신 비-일시적인 유형의 저장 매체들로 지향됨을 이해해야 한다. 본 명세서에서 사용된 바와 같은 디스크 (disk) 및 디스크 (disc) 는 콤팩트 디스크 (CD), 레이저 디스크, 광학 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루레이 디스크를 포함하며, 여기서, 디스크 (disk) 는 통상적으로 데이터를 자기적으로 재생하지만 디스크 (disc) 는 레이저들을 이용하여 데이터를 광학적으로 재생한다. 위의 조합들도 컴퓨터 판독가능 매체들의 범위 내에 포함되어야 한다.
- [0201] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 등가의 집적된 또는 별도의 로직 회로와 같은 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 본 명세서에서 사용되는 바와 같은 용어 "프로세서" 는 본 명세서에서 설명된 기술들의 구현에 적절한 전술한 구조 또는 임의의 다른 구조 중 임의의 구조를 지칭할 수도 있다. 부가적으로, 일부 양태들에 있어서, 본 명세서에서 설명된 기능은 인코딩 및 디코딩을 위해 구성되고 결합된 코덱에서 통합된 전용 하드웨어 모듈 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 그 기술들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.
- [0202] 본 개시의 기술들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예를 들어, 칩 세트) 를 포함하여 매우 다양한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들 또는 유닛들이 개시된 기

술들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시에서 설명되지만, 반드시 상이한 하드웨어 유닛들에 의한 실현을 요구하지는 않는다. 오히려, 상기 설명된 바와 같이, 다양한 유닛들은 적절한 소프트웨어 및/또는 펌웨어와 함께 상기 설명된 바와 같은 하나 이상의 프로세서들을 포함하여 코텍 하드웨어 유닛으로 결합되거나 또는 상호운용식 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

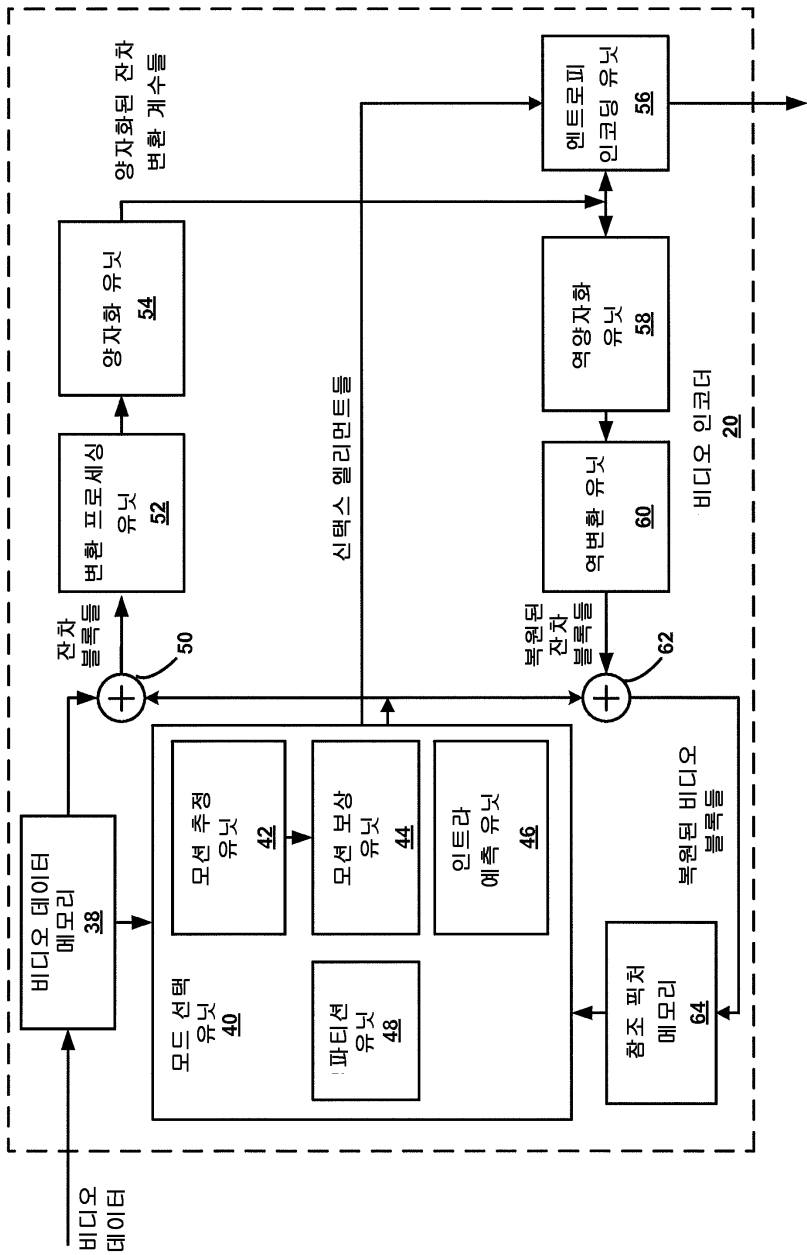
다양한 예들이 설명되었다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

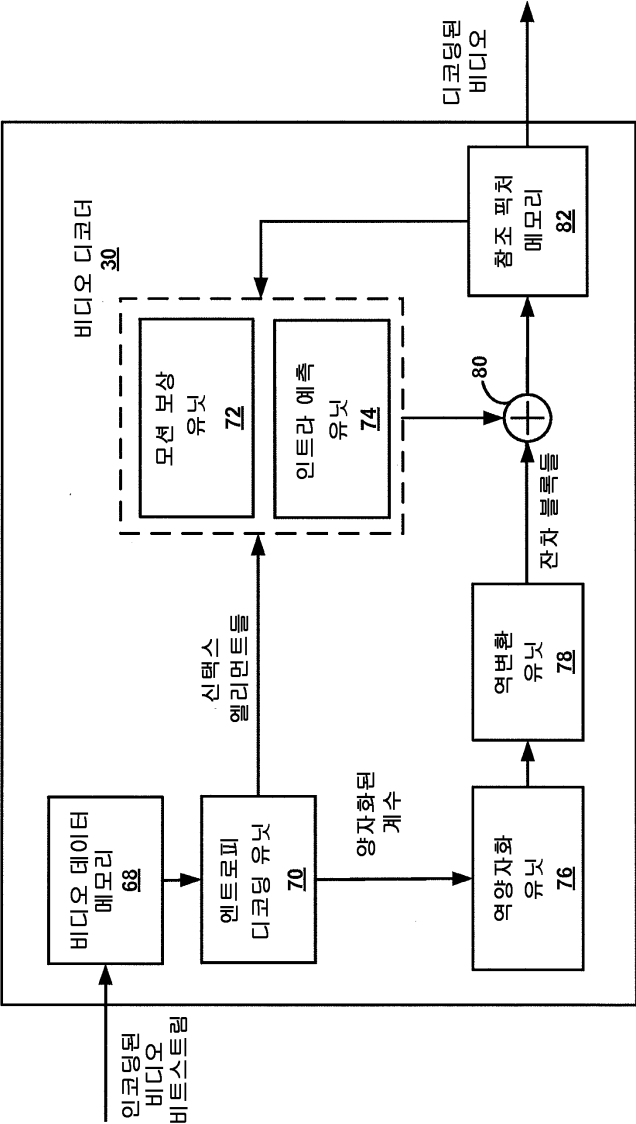
도면1



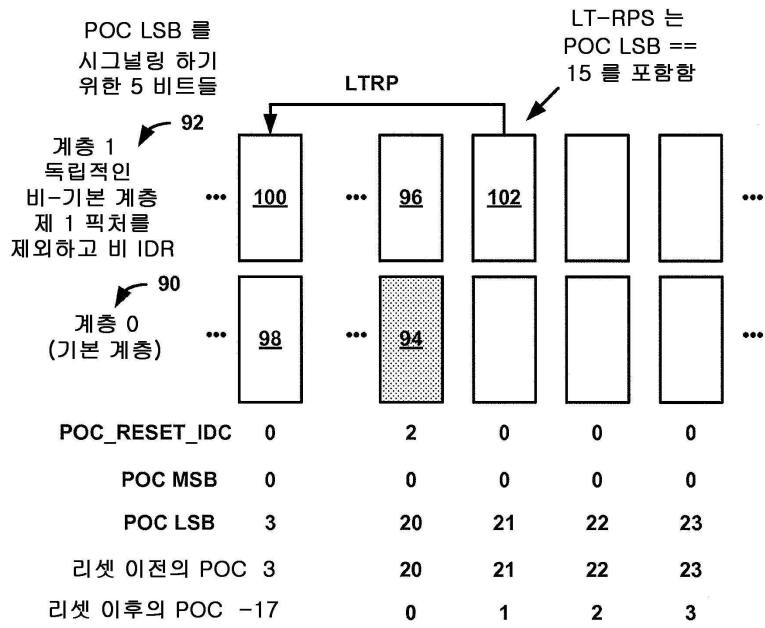
도면2



도면3

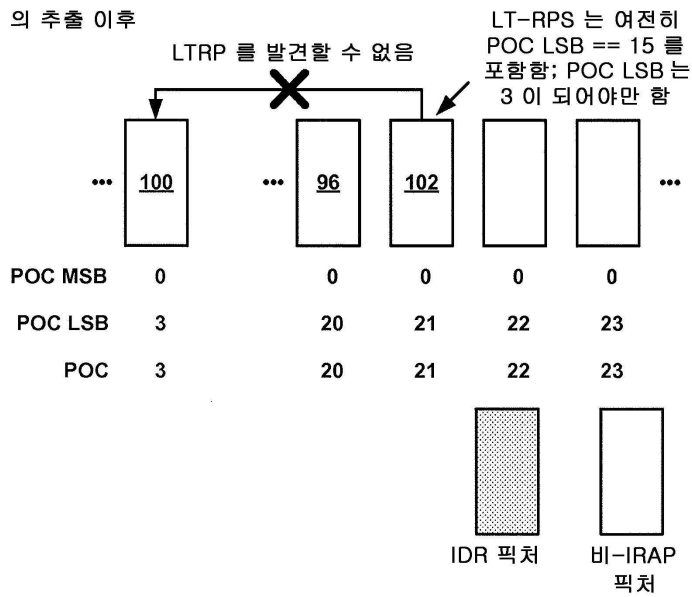


도면4

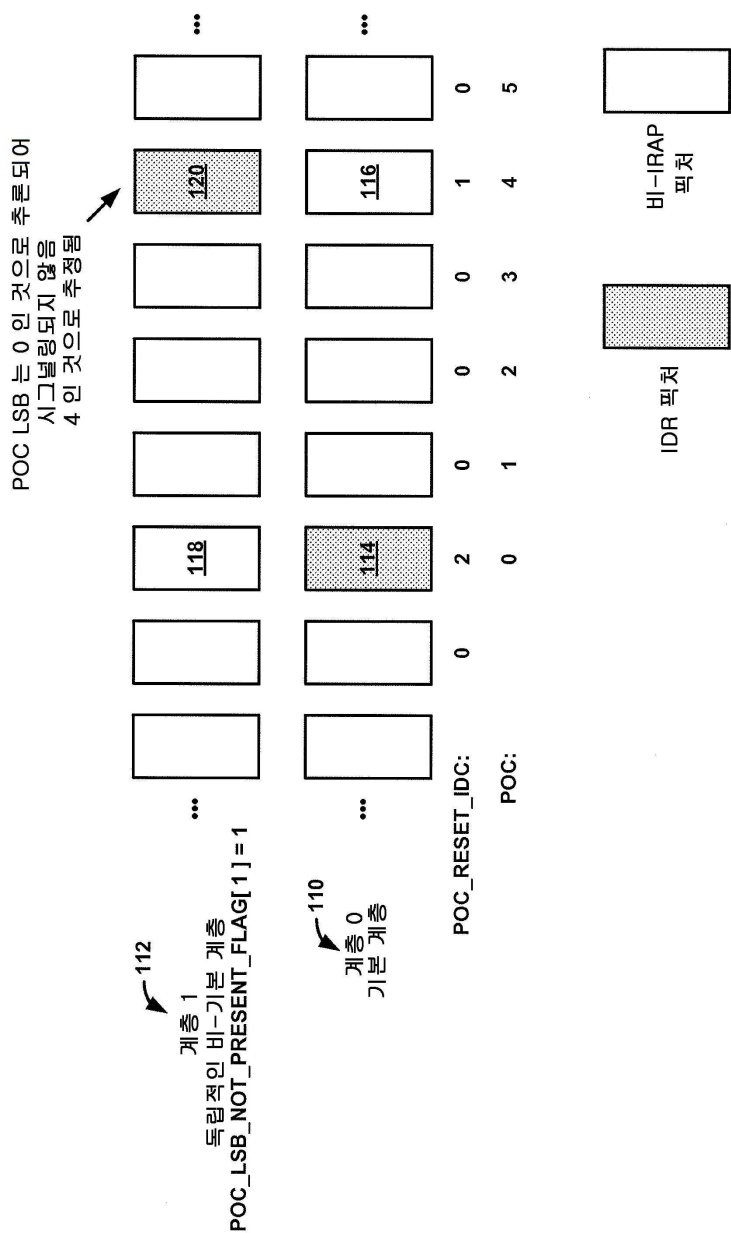


계층 1 의 추출 이전

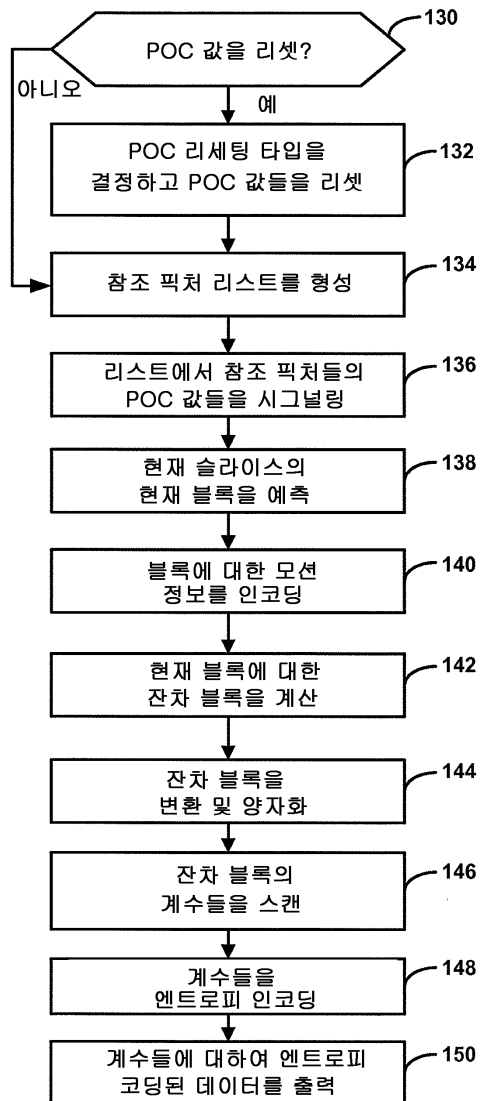
계층 1 의 추출 이후



도면5



도면6



도면7

