



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0113402
(43) 공개일자 2022년08월12일

- (51) 국제특허분류(Int. Cl.)
 HO4N 19/31 (2014.01) HO4N 19/105 (2014.01)
 HO4N 19/139 (2014.01) HO4N 19/172 (2014.01)
 HO4N 19/423 (2014.01) HO4N 19/44 (2014.01)
 HO4N 19/70 (2014.01)
- (52) CPC특허분류
 HO4N 19/31 (2015.01)
 HO4N 19/105 (2015.01)
- (21) 출원번호 10-2022-7020798
- (22) 출원일자(국제) 2022년12월22일
 심사청구일자 없음
- (85) 번역문제출일자 2022년06월17일
- (86) 국제출원번호 PCT/US2020/066583
- (87) 국제공개번호 WO 2021/133788
 국제공개일자 2021년07월01일
- (30) 우선권주장
 62/953,394 2019년12월24일 미국(US)
 17/128,767 2020년12월21일 미국(US)

- (71) 출원인
 켈컴 인코포레이티드
 미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 발명자
 세레진 바딤
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 라마수브라모니안 아다르쉬 크리쉬난
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 코반 무하메드 제이드
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
- (74) 대리인
 특허법인코리아나

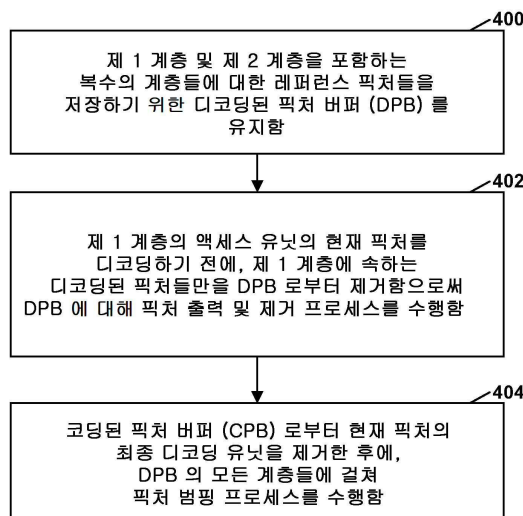
전체 청구항 수 : 총 25 항

(54) 발명의 명칭 다중 계층들에 대한 공유되는 디코더 픽처 버퍼

(57) 요약

다중-계층 비디오 데이터를 디코딩하기 위한 비디오 디코더는, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 것으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는 상기 DPB 를 유지하고; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하며 - 여기서 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 하나 이상의 프로세서들은 또한, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거함 - ; 그리고 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하도록 구성된다.

대표도 - 도7



(52) CPC특허분류

HO4N 19/139 (2015.01)

HO4N 19/172 (2015.01)

HO4N 19/423 (2015.01)

HO4N 19/44 (2015.01)

HO4N 19/70 (2015.01)

명세서

청구범위

청구항 1

다중-계층 비디오 데이터를 디코딩하기 위한 디바이스로서,

다중-계층 비디오 데이터를 저장하도록 구성된 메모리;

회로부에서 구현되는 하나 이상의 프로세서들을 포함하고,

상기 하나 이상의 프로세서들은,

복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 것으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하고;

상기 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 것으로서, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 상기 제 1 계층에 속하는 디코딩된 픽처들만을 상기 DPB 로부터 제거하도록 구성되는, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하며; 그리고

코딩된 픽처 버퍼 (CPB) 로부터 상기 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하도록

구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 2

제 1 항에 있어서,

상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 상기 제 1 계층의 상기 현재 픽처를 디코딩하기 전에 그리고 상기 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 3

제 1 항에 있어서,

상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 상기 CPB 로부터 상기 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 4

제 1 항에 있어서,

상기 DPB 의 모든 계층들에 걸쳐 상기 픽처 범핑 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않은 것으로 마킹된 레퍼런스 픽처를 상기 DPB 로부터 제거하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 5

제 1 항에 있어서,

상기 DPB 의 모든 계층들에 걸쳐 상기 픽처 범핑 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 상기 DPB 로부터 상기 제 2 계층의 적어도 하나의 픽처를 제거하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 6

제 1 항에 있어서,

상기 하나 이상의 프로세서들은 추가로,

상기 DPB 에서 레퍼런스 픽처를 식별하고;

상기 레퍼런스 픽처에서 상기 현재 픽처의 현재 블록에 대한 예측 블록을 식별하고;

상기 예측 블록에 기초하여 상기 현재 블록을 디코딩하고; 그리고

상기 현재 픽처의 디코딩된 버전을 출력하도록

구성되고, 상기 현재 픽처의 상기 디코딩된 버전은 상기 현재 블록의 디코딩된 버전을 포함하는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 7

제 1 항에 있어서,

상기 DPB 는 이전에 디코딩된 픽처들을 저장하도록 구성되고, 상기 CPB 는 인코딩된 다중-계층 비디오 데이터를 저장하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 8

제 1 항에 있어서,

상기 하나 이상의 프로세서들은 추가로, 상기 메모리에 상기 DPB 및 상기 CPB 를 유지하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 9

제 1 항에 있어서,

상기 디바이스는 무선 통신 디바이스를 포함하고, 인코딩된 다중-계층 비디오 데이터를 수신하도록 구성된 수신기를 더 포함하는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 10

제 9 항에 있어서,

상기 무선 통신 디바이스는 전화 핸드셋을 포함하고, 상기 수신기는, 무선 통신 표준에 따라, 상기 인코딩된 다중-계층 비디오 데이터를 포함하는 신호를 복조하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 11

제 1 항에 있어서,

상기 디바이스는 무선 통신 디바이스를 포함하고, 인코딩된 다중-계층 비디오 데이터를 송신하도록 구성된 송신기를 더 포함하는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 12

제 11 항에 있어서,

상기 무선 통신 디바이스는 전화 핸드셋을 포함하고, 상기 송신기는, 무선 통신 표준에 따라, 상기 인코딩된 다중-계층 비디오 데이터를 포함하는 신호를 변조하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 13

제 1 항에 있어서,

상기 현재 픽처의 디코딩된 버전을 포함하는 디코딩된 다중-계층 비디오 데이터를 디스플레이하도록 구성된 디스플레이를 더 포함하는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 14

제 1 항에 있어서,

상기 디바이스는 카메라, 컴퓨터, 모바일 디바이스, 브로드캐스트 수신기 디바이스, 또는 셋탑 박스 중 하나 이상을 포함하는, 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 15

다중-계층 비디오 데이터를 디코딩하는 방법으로서,

복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 단계로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하는 단계;

상기 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계로서, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 단계는 상기 제 1 계층에 속하는 디코딩된 픽처들만을 상기 DPB 로부터 제거하는 단계를 포함하는, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 단계; 및

코딩된 픽처 버퍼 (CPB) 로부터 상기 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 단계를 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 16

제 15 항에 있어서,

상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 단계는 상기 제 1 계층의 상기 현재 픽처를 디코딩하기 전에 그리고 상기 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 단계를 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 17

제 15 항에 있어서,

상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 단계는 상기 CPB 로부터 상기 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 단계를 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 18

제 15 항에 있어서,

상기 DPB 의 모든 계층들에 걸쳐 상기 픽처 범핑 프로세스를 수행하는 단계는,

출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않는 것으로 마킹된 레퍼런스 픽처를 상기 DPB 로부터 제거하는 단계를 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 19

제 15 항에 있어서,

상기 DPB 의 모든 계층들에 걸쳐 상기 픽처 범핑 프로세스를 수행하는 단계는 상기 DPB 로부터 상기 제 2 계층의 적어도 하나의 픽처를 제거하는 단계를 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 20

제 15 항에 있어서,

상기 DPB 에서 레퍼런스 픽처를 식별하는 단계;

상기 레퍼런스 픽처에서 상기 현재 픽처의 현재 블록에 대한 예측 블록을 식별하는 단계;

상기 예측 블록에 기초하여 상기 현재 블록을 디코딩하는 단계; 및

상기 현재 픽처의 디코딩된 버전을 출력하는 단계를 더 포함하고, 상기 현재 픽처의 상기 디코딩된 버전은 상기 현재 블록의 디코딩된 버전을 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 21

제 20 항에 있어서,

상기 현재 픽처의 상기 디코딩된 버전을 디스플레이하는 단계를 더 포함하는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 22

제 15 항에 있어서,

상기 DPB 는 이전에 디코딩된 픽처들을 저장하도록 구성되고, 상기 CPB 는 인코딩된 다중-계층 비디오 데이터를 저장하도록 구성되는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 23

제 15 항에 있어서,

상기 디코딩하는 방법은 인코딩 프로세스의 부분으로서 수행되는, 다중-계층 비디오 데이터를 디코딩하는 방법.

청구항 24

명령들을 저장한 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 하나 이상의 프로세서들에 의해 실행될 경우, 상기 하나 이상의 프로세서들로 하여금,

복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하게 하는 것으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하게 하고;

상기 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하는 것으로서, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금, 상기 제 1 계층에 속하는 디코딩된 픽처들만을 상기 DPB 로부터 제거하게 하는, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하게 하며; 그리고

코딩된 픽처 버퍼 (CPB) 로부터 상기 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하게 하는, 컴퓨터 판독가능 저장 매체.

청구항 25

다중-계층 비디오 데이터를 디코딩하기 위한 장치로서,

복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 수단로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하는 수단;

상기 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단으로서, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 것은 상기 제 1 계층에 속하는 디코딩된 픽처들만을 상기 DPB 로부터 제거하는 것을 포함하는, 상기 DPB 에 대해 상기 픽처 출력 및 제거 프로세스를 수행하는 수단; 및

코딩된 픽처 버퍼 (CPB) 로부터 상기 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 수단을 포함하는, 다중-계층 비디오 데이터를 디코딩하기 위한 장치.

발명의 설명

기술 분야

[0001] 본 출원은 2019년 12월 24일 출원된 미국 가특허 출원 제 62/953,394 호의 이익을 주장하는, 2020년 12월 21일 출원된 미국 출원 제 17/128,767 호에 대한 우선권을 주장하며, 이들 각각의 전체 내용은 본 명세서에 참조에 의해 통합된다.

[0002] 기술분야

[0003] 본 개시는 비디오 인코딩 및 비디오 디코딩에 관한 것이다.

배경 기술

[0004] 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), ITU-T H.265/고 효율 비디오 코딩 (HEVC) 에 의해 정의된 표준들, 및 그러한 표준들의 확장들에서 설명된 것들과 같은 비디오 코딩 기법들을 구현한다. 비디오 디바이스들은 그러한 비디오 코딩 기법들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0005] 비디오 코딩 기법들은 비디오 시퀀스들에 내재하는 리던던시를 감소 또는 제거하기 위해 공간 (인트라-픽처) 예측 및/또는 시간 (인터-픽처) 예측을 포함한다. 블록 기반 비디오 코딩에 대해, 비디오 슬라이스 (예컨대, 비디오 픽처 또는 비디오 픽처의 부분) 는 비디오 블록들로 파티셔닝될 수도 있고, 이 비디오 블록들은 또한 코딩 트리 유닛들 (CTU들), 코딩 유닛들 (CU들) 및/또는 코딩 노드들로서 지칭될 수도 있다. 픽처의 인트라-코딩된 (I) 슬라이스에서의 비디오 블록들은 동일한 픽처의 이웃하는 블록들에서의 레퍼런스 샘플들에 대한 공간 예측을 사용하여 인코딩된다. 픽처의 인터-코딩된 (P 또는 B) 슬라이스에서의 비디오 블록들은 동일한 픽처의 이웃하는 블록들에서의 레퍼런스 샘플들에 대한 공간 예측, 또는 다른 레퍼런스 픽처들에서의 레퍼런스 샘플들에 대한 시간 예측을 사용할 수도 있다. 픽처들은 프레임들로서 지칭될 수도 있고, 레퍼런스 픽처들은 레퍼런스 프레임들로서 지칭될 수도 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0006] 본 개시물은 다중-계층 비디오 데이터를 디코딩하기 위해 디코딩된 픽처 버퍼를 유지하기 위한 기법들을 설명한다. DPB 를 유지하는 것의 일부로서, 비디오 디코더는 DPB 에 추가될 새로운 픽처들을 위한 공간을 만들기 위해 DPB 로부터 픽처들을 주기적으로 제거할 필요가 있을 수도 있다. 다중-계층 비디오 데이터를 디코딩할 때, 일부 기존의 비디오 디코더들은 서브-DPB들을 유지하며, 여기서 각각의 서브-DPB 는 하나의 계층의 디코딩된 픽처들의 저장을 위한 픽처 저장 버퍼들을 포함한다. 대조적으로, 다중-계층 비디오 데이터를 디코딩할 때, 일부 기존의 비디오 디코더들은 모든 계층들에 대해 공통 DPB 를 유지하며, 이는 복잡성을 감소시키지만 잠재적으로 다른 문제들을 야기한다. 예를 들어, 범핑 프로세스가 코딩되고 있는 픽처와 동일한 계층에 속하는 픽처들에 대해서만 호출되면, 버퍼가 그 계층의 임의의 픽처들을 포함하지 않는 일부 코딩 시나리오들에서, 범핑 프로세스는 새로운 픽처들에 대한 DPB 에서의 공간을 적절하게 생성하지 않을 수도 있다. 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거함으로써 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 것, 및 CPB 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 것을 포함하는 본 개시물의 기법들은, 유리하게 비디오 디코더가 모든 계층들에 대해 공통 DPB 를 활용하는 것을 가능하게 하면서, 필요한 레퍼런스 픽처들이 DPB 로부터 조기에 제거되지 않는 것을 보장하고 또한 DPB 가 과도하게 가득차지 않는 것을 보장할 수도 있다.

[0007] 본 개시물의 일 예에 따르면, 비디오 데이터를 디코딩하기 위한 디바이스는, 비디오 데이터를 저장하도록 구성된 메모리, 및 회로로 구현된 하나 이상의 프로세서들을 포함하고, 하나 이상의 프로세서들은, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 것으로서, 상기 복수의 계층들은

적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하고; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 것으로서, 여기서 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 하나 이상의 프로세서들은 또한, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하는, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하고; 그리고 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하도록 구성된다.

[0008] 본 개시물의 다른 예에 따르면, 방법은, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 단계로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하는 단계; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계로서, 여기서 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계는, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하는, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계; 및 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 단계를 포함한다.

[0009] 본 개시물의 또 다른 예에 따르면, 컴퓨터 판독가능 저장 매체는 명령들을 저장하고, 상기 명령들은 하나 이상의 프로세서들에 의해 실행될 때, 하나 이상의 프로세서들로 하여금, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하게 하는 것으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하게 하고; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하는 것으로서, 상기 하나 이상의 프로세서들은 추가로, 하나 이상의 프로세서들은 또한, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하는, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하고; 그리고 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하게 한다.

[0010] 본 개시물의 다른 예에 따르면, 다중-계층 비디오 데이터를 디코딩하기 위한 장치는, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 수단으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하는 수단; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단으로서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 것은 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하는 것을 포함하는, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단; 및 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 수단을 포함한다.

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

도면의 간단한 설명

[0012] 도 1 은 본 개시물의 기법들을 수행할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록 다이어그램이다.

도 2a 및 도 2b 는 예시적인 쿼드트리 바이너리 트리 (QTBT) 구조 및 대응하는 코딩 트리 유닛 (CTU) 을 예시하는 개념 다이어그램들이다.

도 3 은 본 개시의 기술들을 수행할 수도 있는 예시적인 비디오 인코더를 예시하는 블록 다이어그램이다.

도 4 는 본 개시물의 기법들을 수행할 수도 있는 예시적인 비디오 디코더를 예시하는 블록 다이어그램이다.

도 5 은 비디오 인코딩 프로세스를 예시한 플로우차트이다.

도 6 는 비디오 디코딩 프로세스를 예시한 플로우차트이다.

도 7 는 비디오 디코딩 프로세스를 예시한 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0013] 비디오 코딩 (예컨대, 비디오 인코딩 및/또는 비디오 디코딩) 은 통상적으로, 동일한 픽처에서의 비디오 데이터의 이미 코딩된 블록 (예컨대, 인트라 예측) 또는 상이한 픽처에서의 비디오 데이터의 이미 코딩된 블록 (예컨대,

대, 인터 예측) 중 어느 하나로부터 비디오 데이터의 블록을 예측하는 것을 수반한다. 일부 사례들에 있어서, 비디오 인코더는 또한, 예측 블록을 원래 블록과 비교함으로써 잔차 데이터를 계산한다. 따라서, 잔차 데이터는 예측 블록과 원래 블록 사이의 차이를 나타낸다. 잔차 데이터를 시그널링하는데 필요한 비트들의 수를 감소시키기 위해, 비디오 인코더는 잔차 데이터를 변환 및 양자화하고, 변환 및 양자화된 잔차 데이터를 인코딩된 비트스트림에서 시그널링한다. 변환 및 양자화 프로세스들에 의해 달성되는 압축은 손실성일 수도 있으며, 이는 변환 및 양자화 프로세스들이 디코딩된 비디오 데이터에 왜곡을 도입할 수도 있다는 것을 의미한다.

[0014] 비디오 디코더는 잔차 데이터를 디코딩하고 예측 블록에 추가하여, 예측 블록 단독보다 더 가깝게 원래 비디오 블록과 매칭하는 복원된 비디오 블록을 생성한다. 잔차 데이터의 변환 및 양자화에 의해 도입된 손실로 인해, 제 1의 복원된 블록은 왜곡 또는 아티팩트들을 가질 수도 있다. 아티팩트 또는 왜곡의 하나의 일반적인 타입은 블록키니스 (blockiness)로서 지칭되며, 여기서, 비디오 데이터를 코딩하는데 사용되는 블록들의 경계들은 가시적이다.

[0015] 디코딩된 비디오의 품질을 더 개선하기 위해, 비디오 디코더는 복원된 비디오 블록들에 대해 하나 이상의 필터링 동작들을 수행할 수 있다. 이들 필터링 동작들의 예들은 더블록킹 필터링, 샘플 적응적 오프셋 (SAO) 필터링, 및 적응적 루프 필터링 (ALF) 을 포함한다. 이들 필터링 동작들에 대한 파라미터들은 비디오 인코더에 의해 결정되고, 인코딩된 비디오 비트스트림에서 명시적으로 시그널링될 수도 있거나, 또는 파라미터들이 인코딩된 비디오 비트스트림에서 명시적으로 시그널링될 필요없이 비디오 디코더에 의해 암시적으로 결정될 수도 있다.

[0016] 비디오 인코더는 비디오 데이터를 네트워크 추상화 계층 (NAL) 유닛들로서 인코딩하고, 비디오 디코더는 비디오 데이터를 NAL 유닛들로서 디코딩한다. NAL 유닛은 일반적으로 뒤따르기 위한 데이터의 타입의 표시, 및 필요에 따라 에물레이션 방지 바이트들로서 산재된 원시 바이트 시퀀스 페이로드 (RBSP) 의 형태로 그 데이터를 포함하는 바이트들을 포함하는 선택스 구조를 지칭한다. NAL 유닛들은 비디오 코딩 계층 (VCL) NAL 유닛들 및 비-VCL NAL 유닛들 양자를 포함할 수도 있다.

[0017] 픽처 유닛 (PU) 은 일반적으로, 명시된 분류 규칙에 따라 서로와 연관되고 디코딩 순서에서 연속적이며 정확히 하나의 코딩된 픽처를 포함하는 NAL 유닛들의 세트를 지칭한다. 액세스 유닛들은 일반적으로 상이한 계층들에 속하고 출력을 위해 동일한 시간과 연관된 코딩된 픽처들을 포함하는 PU들의 세트를 지칭한다.

[0018] 공간적 및 시간적 확장성을 지원하기 위해, 비디오 데이터는 다수의 계층들에서 코딩될 수 있다. 계층은 일반적으로, 동일한 계층 식별 값 (예를 들면, nuh_layer_id 값) 을 모두 갖는 VCL NAL 유닛들의 세트 및 연관된 비-VCL NAL 유닛들을 지칭한다. 다중-계층 비디오 데이터를 디코딩할 때, 비디오 디코더는 모든 계층들 또는 그 계층들의 서브세트만을 디코딩하고 디스플레이할 수도 있다.

[0019] 비디오 디코더는 코딩된 픽처 버퍼 (CPB) 를 유지, 예를 들어, 저장 및 업데이트할 수도 있다. CPB 는, 예를 들어, 특정된 디코딩 순서에서 디코딩 유닛들 (DU들) 을 포함하는 선입선출 (first-in first-out) 버퍼일 수도 있다. 비디오 디코더는 또한 디코딩된 픽처 버퍼 (DPB) 를 유지할 수도 있다. DPB 는 일반적으로 참조, 출력 재정렬, 또는 특정된 출력 지연을 위해 디코딩된 픽처들을 유지하는 버퍼 또는 메모리를 지칭한다. DU 는 액세스 유닛 (AU) 내의 하나 이상의 VCL NAL 유닛들 및 연관된 비-VCL NAL 유닛들을 지칭할 수도 있다. 일부 경우들에서, DU 는 또한 AU 일 수도 있다.

[0020] DPB 를 유지하는 것의 일부로서, 비디오 디코더는 DPB 에 추가될 새로운 픽처들을 위한 공간을 만들기 위해 DPB 로부터 픽처들을 주기적으로 제거할 필요가 있을 수도 있다. 다중-계층 비디오 데이터를 디코딩할 때, 일부 기존의 비디오 디코더들은 서브-DPB들을 유지하며, 여기서 각각의 서브-DPB 는 하나의 계층의 디코딩된 픽처들의 저장을 위한 픽처 저장 버퍼들을 포함한다. 대조적으로, 다중-계층 비디오 데이터를 디코딩할 때, 일부 기존의 비디오 디코더들은 모든 계층들에 대해 공통 DPB 를 유지하며, 이는 복잡성을 감소시키지만 잠재적으로 다른 문제들을 야기한다. 예를 들어, 범핑 프로세스가 코딩되고 있는 픽처와 동일한 계층에 속하는 픽처들에 대해서만 호출되면, 버퍼가 그 계층의 임의의 픽처들을 포함하지 않는 일부 코딩 시나리오들에서, 범핑 프로세스는 새로운 픽처들에 대한 DPB 에서의 공간을 적절하게 생성하지 않을 수도 있다. 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거함으로써 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 것, 및 CPB 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 것을 포함하는 본 개시물의 기법들은, 유리하게 비디오 디코더가 모든 계층들에 대해 공통 DPB 를 활용하는 것을 가능하게 하면서, 필요한 레퍼런스 픽처들이 DPB 로부터 조기에 제거되지 않는 것을 보장하고 또한 DPB 가

과도하게 가득차지 않는 것을 보장할 수도 있다.

- [0021] 본 개시물에 설명된 특정 기법들은 비디오 디코딩과 관련하여 설명될 수도 있다. 그러나, 반대로 특정되지 않는 한, 이들 기법들은 또한 비디오 인코더에 의해 수행될 수도 있다는 것이 이해되어야 한다. 예를 들어, 비디오 인코더는 통상적으로, 비디오 데이터를 인코딩하는 방법을 결정하는 프로세스들의 부분으로서 비디오 디코딩 (복원으로 또한 지칭됨) 을 수행한다. 예를 들어, 비디오 데이터의 블록들을 인코딩하는 방법을 결정하는 것의 일부로서, 비디오 인코더는 비디오 디코더와 동일한 DPB 업데이트 프로세스들을 구현하도록 구성될 수도 있어서, 비디오 인코더 및 비디오 디코더가 동일한 DPB들을 유지하게 한다.
- [0022] 도 1 은 본 개시물의 기법들을 수행할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (100) 을 예시하는 블록 다이어그램이다. 본 개시의 기법들은 일반적으로 비디오 데이터를 코딩 (인코딩 및/또는 디코딩) 하는 것과 관련된다. 일반적으로, 비디오 데이터는 비디오를 프로세싱하기 위한 임의의 데이터를 포함한다. 따라서, 비디오 데이터는 원시, 인코딩되지 않은 비디오, 인코딩된 비디오, 디코딩된 (예를 들어, 복원된) 비디오, 및 비디오 메타데이터, 이를 테면 시그널링 데이터를 포함할 수도 있다.
- [0023] 도 1 에 도시된 바와 같이, 시스템 (100) 은 이 예에서 목적지 디바이스 (116) 에 의해 디코딩 및 디스플레이될 인코딩된 비디오 데이터를 제공하는 소스 디바이스 (102) 를 포함한다. 특히, 소스 디바이스 (102) 는 컴퓨터 판독 가능 매체 (110) 를 통해 목적지 디바이스 (116) 에 비디오 데이터를 제공한다. 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 모바일 디바이스들, 태블릿 컴퓨터들, 셋-탑 박스들, 전화기 핸드셋들, 예를 들어 스마트폰들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스, 브로드캐스트 수신기 디바이스 등을포함한, 광범위한 디바이스들 중 임의의 것을 포함할 수도 있다. 일부 경우들에서, 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 무선 통신을 위해 장비될 수도 있고, 따라서 무선 통신 디바이스들로 지칭될 수도 있다.
- [0024] 도 1 의 예에서, 소스 디바이스 (102) 는 비디오 소스 (104), 메모리 (106), 비디오 인코더 (200), 및 출력 인터페이스 (108) 를 포함한다. 목적지 디바이스 (116) 는 입력 인터페이스 (122), 비디오 디코더 (300), 메모리 (120), 및 디스플레이 디바이스 (118) 를 포함한다. 본 개시에 따르면, 소스 디바이스 (102) 의 비디오 인코더 (200) 및 목적지 디바이스 (116) 의 비디오 디코더 (300) 는 다중 계층들에 대한 공유되는 디코딩된 픽처 버퍼를 위한 기법들을 적용하도록 구성될 수도 있다. 따라서, 소스 디바이스 (102) 는 비디오 인코딩 디바이스의 예를 나타내는 한편, 목적지 디바이스 (116) 는 비디오 디코딩 디바이스의 예를 나타낸다. 다른 예들에서, 소스 디바이스 및 목적지 디바이스는 다른 컴포넌트들 또는 배열들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (102) 는 외부 카메라와 같은 외부 비디오 소스로부터 비디오 데이터를 수신할 수도 있다. 마찬가지로, 목적지 디바이스 (116) 는 통합된 디스플레이 디바이스를 포함하기 보다는, 외부 디스플레이 디바이스와 인터페이스할 수도 있다.
- [0025] 도 1 에 도시된 바와 같은 시스템 (100) 은 단지 하나의 예이다. 일반적으로, 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스는 다중 계층들에 대한 공유된 디코딩된 픽처 버퍼에 대한 기법들을 수행할 수도 있다. 소스 디바이스 (102) 및 목적지 디바이스 (116) 는, 소스 디바이스 (102) 가 목적지 디바이스 (116) 로의 송신을 위해 코딩된 비디오 데이터를 생성하는 그러한 코딩 디바이스들의 예들일 뿐이다. 본 개시는 데이터의 코딩 (인코딩 및/또는 디코딩) 을 수행하는 디바이스로서 "코딩" 디바이스를 참조한다. 따라서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 코딩 디바이스들, 특히 각각 비디오 인코더 및 비디오 디코더의 예들을 나타낸다. 일부 예들에서, 소스 디바이스 (102) 및 목적지 디바이스 (116) 는, 소스 디바이스 (102) 및 목적지 디바이스 (116) 의 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 이로써, 시스템 (100) 은 예를 들어, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 텔레포니를 위해, 소스 디바이스 (102) 와 목적지 디바이스 (116) 사이의 일방향 또는 양방향 비디오 송신을 지원할 수도 있다.
- [0026] 일반적으로, 비디오 소스 (104) 는 비디오 데이터 (즉, 원시, 인코딩되지 않은 비디오 데이터) 의 소스를 나타내며 픽처들에 대한 데이터를 인코딩하는 비디오 인코더 (200) 에 비디오 데이터의 순차적인 일련의 픽처들 (또한 "프레임들" 로서 지칭됨) 을 제공한다. 소스 디바이스 (102) 의 비디오 소스 (104) 는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 원시 비디오를 포함하는 비디오 아카이브, 및/또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스를 포함할 수도 있다. 추가의 대안으로서, 비디오 소스 (104) 는 컴퓨터 그래픽 기반 데이터를 소스 비디오로서, 또는 라이브 비디오, 아카이브된 비디오,

및 컴퓨터 생성된 비디오의 조합으로서 생성할 수도 있다. 각각의 경우에 있어서, 비디오 인코더 (200) 는 캡처된, 사전-캡처된, 또는 컴퓨터 생성된 비디오 데이터를 인코딩한다. 비디오 인코더 (200) 는 픽처들을 수신된 순서 (때때로 "디스플레이 순서" 로서 지칭됨) 로부터 코딩을 위한 코딩 순서로 재배열할 수도 있다. 비디오 인코더 (200) 는 인코딩된 비디오 데이터를 포함하는 비트스트림을 생성할 수도 있다. 그 다음, 소스 디바이스 (102) 는 예를 들어, 목적지 디바이스 (116) 의 입력 인터페이스 (122) 에 의한 수신 및/또는 취출을 위해 인코딩된 비디오 데이터를 출력 인터페이스 (108) 를 통해 컴퓨터 관독가능 매체 (110) 상으로 출력할 수도 있다.

[0027] 소스 디바이스 (102) 의 메모리 (106) 및 목적지 디바이스 (116) 의 메모리 (120) 는 범용 메모리들을 나타낸다. 일부 예들에서, 메모리들 (106, 120) 은 원시 비디오 데이터, 예컨대, 비디오 소스 (104) 로부터의 원시 비디오 및 비디오 디코더 (300) 로부터의 원시, 디코딩된 비디오 데이터를 저장할 수도 있다. 추가적으로 또는 대안적으로, 메모리들 (106, 120) 은, 예컨대, 비디오 인코더 (200) 및 비디오 디코더 (300) 에 의해 각각 실행가능한 소프트웨어 명령들을 저장할 수도 있다. 메모리 (106) 및 메모리 (120) 가 이 예에서는 비디오 인코더 (200) 및 비디오 디코더 (300) 와 별도로 도시되지만, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 또한 기능적으로 유사하거나 또는 동등한 목적들을 위한 내부 메모리들을 포함할 수도 있음을 이해해야 한다. 더욱이, 메모리들 (106, 120) 은, 예컨대, 비디오 인코더 (200) 로부터 출력되고 비디오 디코더 (300) 에 입력되는 인코딩된 비디오 데이터를 저장할 수도 있다. 일부 예들에 있어서, 메모리들 (106, 120) 의 부분들은, 예컨대, 원시의, 디코딩된, 및/또는 인코딩된 비디오 데이터를 저장하기 위해 하나 이상의 비디오 버퍼들로서 할당될 수도 있다.

[0028] 컴퓨터 관독가능 매체 (110) 는 인코딩된 비디오 데이터를 소스 디바이스 (102) 로부터 목적지 디바이스 (116) 로 전송할 수 있는 임의의 타입의 매체 또는 디바이스를 나타낼 수도 있다. 일 예에 있어서, 컴퓨터 관독가능 매체 (110) 는, 소스 디바이스 (102) 로 하여금 인코딩된 비디오 데이터를 직접 목적지 디바이스 (116) 에 실시간으로, 예컨대, 무선 주파수 네트워크 또는 컴퓨터 기반 네트워크를 통해 송신할 수 있게 하기 위한 통신 매체를 나타낸다. 무선 통신 프로토콜과 같은 통신 표준에 따라, 출력 인터페이스 (108) 는 인코딩된 비디오 데이터를 포함하는 송신 신호를 변조할 수도 있고, 입력 인터페이스 (122) 는 수신된 송신 신호를 복조할 수도 있다. 통신 매체는 임의의 무선 또는 유선 통신 매체, 이를 테면 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들을 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 (102) 로부터 목적지 디바이스 (116) 로의 통신을 가능하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0029] 일부 예들에서, 소스 디바이스 (102) 는 출력 인터페이스 (108) 로부터 저장 디바이스 (112) 로 인코딩된 데이터를 출력할 수도 있다. 유사하게, 목적지 디바이스 (116) 는 입력 인터페이스 (122) 를 통해 저장 디바이스 (112) 로부터의 인코딩된 데이터에 액세스할 수도 있다. 저장 디바이스 (112) 는 하드 드라이브, 블루레이 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 로컬로 액세스 데이터 저장 매체들 중 임의의 것을 포함할 수도 있다.

[0030] 일부 예들에 있어서, 소스 디바이스 (102) 는, 소스 디바이스 (102) 에 의해 생성된 인코딩된 비디오 데이터를 저장할 수도 있는 파일 서버 (114) 또는 다른 중간 저장 디바이스로 인코딩된 비디오 데이터를 출력할 수도 있다. 목적지 디바이스 (116) 는 스트리밍 또는 다운로드를 통해 파일 서버 (114) 로부터의 저장된 비디오 데이터에 액세스할 수도 있다.

[0031] 파일 서버 (114) 는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (116) 에 송신할 수 있는 임의의 타입의 서버 디바이스일 수도 있다. 파일 서버 (114) 는 (예컨대, 웹 사이트에 대한) 웹 서버, (파일 전송 프로토콜 (FTP) 또는 FLUTE (File Delivery over Unidirectional Transport) 프로토콜과 같은) 파일 전송 프로토콜 서비스를 제공하도록 구성된 서버, 콘텐츠 전달 네트워크 (CDN) 디바이스, 하이퍼텍스트 전송 프로토콜 (HTTP) 서버, 멀티미디어 브로드캐스트 멀티캐스트 서비스 (MBMS) 또는 강화된 MBMS (eMBMS) 서버, 및/또는 네트워크 어태치형 스토리지 (NAS) 디바이스를 나타낼 수도 있다. 파일 서버 (114) 는, 추가적으로 또는 대안적으로, DASH (Dynamic Adaptive Streaming over HTTP), HTTP 라이브 스트리밍 (HLS), 실시간 스트리밍 프로토콜 (RTSP), HTTP 동적 스트리밍 등과 같은 하나 이상의 HTTP 스트리밍 프로토콜들을 구현할 수도 있다.

- [0032] 목적지 디바이스 (116) 는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해 파일 서버 (114) 로부터 인코딩된 비디오 데이터에 액세스할 수도 있다. 이것은 파일 서버 (114) 상에 저장된 인코딩된 비디오 데이터에 액세스하기에 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, 디지털 가입자 라인 (DSL), 케이블 모뎀 등), 또는 양자의 조합을 포함할 수도 있다. 입력 인터페이스 (122) 는 파일 서버 (114) 로부터 미디어 데이터를 추출하거나 수신하기 위해 상기 논의된 다양한 프로토콜들 중 임의의 하나 이상의 프로토콜들, 또는 미디어 데이터를 추출하기 위한 다른 그러한 프로토콜들에 따라 동작하도록 구성될 수도 있다.
- [0033] 출력 인터페이스 (108) 및 입력 인터페이스 (122) 는 무선 송신기들/수신기들, 모뎀들, 유선 네트워킹 컴포넌트들 (예컨대, 이더넷 카드들), 다양한 IEEE 802.11 표준들 중 임의의 것에 따라 동작하는 무선 통신 컴포넌트들, 또는 다른 물리적 컴포넌트들을 나타낼 수도 있다. 출력 인터페이스 (108) 및 입력 인터페이스 (122) 가 무선 컴포넌트들을 포함하는 예들에 있어서, 출력 인터페이스 (108) 및 입력 인터페이스 (122) 는 4G, 4G-LTE (롱텀 에볼루션), LTE 어드밴스드, 5G 등과 같은 셀룰러 통신 표준에 따라, 인코딩된 비디오 데이터와 같은 데이터를 전송하도록 구성될 수도 있다. 출력 인터페이스 (108) 가 무선 송신기를 포함하는 일부 예들에 있어서, 출력 인터페이스 (108) 및 입력 인터페이스 (122) 는 IEEE 802.11 사양, IEEE 802.15 사양 (예컨대, ZigBee™), Bluetooth™ 표준 등과 같은 다른 무선 표준들에 따라, 인코딩된 비디오 데이터와 같은 데이터를 전송하도록 구성될 수도 있다. 일부 예들에서, 소스 디바이스 (102) 및/또는 목적지 디바이스 (116) 는 개별의 시스템-온-칩 (system-on-a-chip; SoC) 디바이스들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (102) 는 비디오 인코더 (200) 및/또는 출력 인터페이스 (108) 에 기인한 기능을 수행하기 위한 SoC 디바이스를 포함할 수도 있고, 목적지 디바이스 (116) 는 비디오 디코더 (300) 및/또는 입력 인터페이스 (122) 에 기인한 기능을 수행하기 위한 SoC 디바이스를 포함할 수도 있다.
- [0034] 본 개시의 기법들은 오버-디-에어 (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 인터넷 스트리밍 비디오 송신들, 예컨대 DASH (dynamic adaptive streaming over HTTP), 데이터 저장 매체 상으로 인코딩되는 디지털 비디오, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 어플리케이션들과 같은 다양한 멀티미디어 어플리케이션들 중 임의의 것을 지원하여 비디오 코딩에 적용될 수도 있다.
- [0035] 목적지 디바이스 (116) 의 입력 인터페이스 (122) 는 컴퓨터 판독가능 매체 (110) (예컨대, 통신 매체, 저장 디바이스 (112), 파일 서버 (114) 등) 로부터 인코딩된 비디오 비트스트림을 수신한다. 인코딩된 비디오 비트스트림은, 비디오 블록들 또는 다른 코딩된 유닛들 (예컨대, 슬라이스들, 픽처들, 픽처들의 그룹들, 시퀀스들 등) 의 프로세싱 및/또는 특성들을 기술하는 값들을 갖는 신택스 엘리먼트들과 같은, 비디오 디코더 (300) 에 의해 또한 사용되는 비디오 인코더 (200) 에 의해 정의된 시그널링 정보를 포함할 수도 있다. 디스플레이 디바이스 (118) 는 디코딩된 비디오 데이터의 디코딩된 픽처들을 사용자에게 디스플레이한다. 디스플레이 디바이스 (118) 는 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 것을 나타낼 수도 있다.
- [0036] 도 1 에 도시되지는 않았지만, 일부 예들에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 각각 오디오 인코더 및/또는 오디오 디코더와 통합될 수도 있고, 공통 데이터 스트림에서 오디오 및 비디오 양자를 포함하는 멀티플렉싱된 스트림들을 핸들링하기 위해, 적절한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및/또는 소프트웨어를 포함할 수도 있다. 적용가능한 경우, MUX-DEMUX 유닛들은 ITU H.223 다중화기 프로토폴, 또는 사용자 데이터그램 프로토콜 (UDP) 과 같은 다른 프로토콜에 따라 수도 있다.
- [0037] 비디오 인코더 (200) 및 비디오 디코더 (300) 각각은 다양한 적합한 인코더 및/또는 디코더 회로부, 이를 테면 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들 중 임의의 것으로서 구현될 수도 있다. 기법들이 부분적으로 소프트웨어에서 구현되는 경우, 디바이스는 적합한 비일시적 컴퓨터 판독가능 매체에 소프트웨어에 대한 명령들을 저장하고, 본 개시의 기법들을 수행하기 위해 하나 이상의 프로세서들을 사용하는 하드웨어에서 그 명령들을 실행할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 의 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 하나는 개별 디바이스에 있어서 결합된 인코더/디코더 (CODEC) 의 부분으로서 통합될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 를 포함하는 디바이스는 집적 회로, 마이크로 프로세서, 및/또는 셀룰러 전화기와 같은 무선 통신 디바이스를 포함할 수도 있다.

- [0038] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 고 효율 비디오 코딩 (HEVC) 으로서도 또한 지칭되는 ITU-T H.265 와 같은 비디오 코딩 표준 또는 그에 대한 확장들, 예컨대 멀티-뷰 및/또는 스케일러블 비디오 코딩 확장들에 따라 동작할 수도 있다. 대안적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 다기능 비디오 코딩 (VVC) 으로서 또한 지칭되는 ITU-T H.266 또는 공동 탐구 테스트 모델 (JEM) 과 같은, 다른 독점 또는 산업 표준들에 따라 동작할 수도 있다. VVC 표준의 최근 드래프트는 Bross 등의 "Versatile Video Coding (Draft 7)," Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 16th Meeting: Geneva, CH, 1-11 October 2019, JVET-P2001-v14 (이하, "VVC 드래프트 7" 이라고 함) 에 기술된다. 하지만, 본 개시의 기법들은 임의의 특정 코딩 표준으로 한정되지 않는다.
- [0039] 일반적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 픽처들의 블록 기반 코딩을 수행할 수도 있다. 용어 "블록" 은 일반적으로 프로세싱될 (예컨대, 인코딩될, 디코딩될, 또는 다르게는 인코딩 및/또는 디코딩 프로세스에서 사용될) 데이터를 포함하는 구조를 지칭한다. 예를 들어, 블록은 루미넌스 및/또는 크로미넌스 데이터의 샘플들의 2 차원 매트릭스를 포함할 수도 있다. 일반적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 YUV (예컨대, Y, Cb, Cr) 포맷으로 표현된 비디오 데이터를 코딩할 수도 있다. 즉, 픽처의 샘플들에 대한 적색, 녹색, 및 청색 (RGB) 데이터를 코딩하기 보다는, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 및 크로미넌스 컴포넌트들을 코딩할 수도 있으며, 여기서, 크로미넌스 컴포넌트들은 적색 색조 및 청색 색조 크로미넌스 컴포넌트들 양자 모두를 포함할 수도 있다. 일부 예들에 있어서, 비디오 인코더 (200) 는 인코딩 이전에 수신된 RGB 포맷팅된 데이터를 YUV 표현으로 변환하고, 비디오 디코더 (300) 는 YUV 표현을 RGB 포맷으로 변환한다. 대안적으로, 프리- 및 포스트-프로세싱 유닛들 (도시되지 않음) 이 이들 변환들을 수행할 수도 있다.
- [0040] 본 개시는 일반적으로 픽처의 데이터를 인코딩 또는 디코딩하는 프로세스를 포함하기 위한 픽처들의 코딩 (예컨대, 인코딩 및 디코딩) 을 참조할 수도 있다. 유사하게, 본 개시는, 블록들에 대한 데이터를 인코딩하거나 또는 디코딩하는 프로세스, 예컨대, 예측 및/또는 잔차 코딩을 포함하도록 픽처의 블록들의 코딩을 참조할 수도 있다. 인코딩된 비디오 비트스트림은 일반적으로 코딩 결정들 (예컨대, 코딩 모드들) 및 픽처들의 블록들의 파티셔닝을 나타내는 신택스 엘리먼트들에 대한 일련의 값들을 포함한다. 따라서, 픽처 또는 블록을 코딩하는 것에 대한 참조들은 일반적으로 픽처 또는 블록을 형성하는 신택스 엘리먼트들에 대한 코딩 값들로서 이해되어야 한다.
- [0041] HEVC 는 코딩 유닛들 (CU들), 예측 유닛들 (PU들), 및 변환 유닛들 (TU들) 을 포함하는 다양한 블록들을 정의한다. HEVC 에 따르면, (비디오 인코더 (200) 와 같은) 비디오 코더는 쿼드트리 구조에 따라 코딩 트리 유닛 (CTU) 을 CU들로 파티셔닝한다. 즉, 비디오 코더는 CTU들 및 CU들을 4 개의 동일한 비오버랩하는 정사각형들로 파티셔닝하고, 쿼드트리의 각각의 노드는 0 개 또는 4 개의 자식 노드들 중 어느 하나를 갖는다. 자식 노드들이 없는 노드들은 "리프 노드들" 로서 지칭될 수도 있고, 그러한 리프 노드들의 CU들은 하나 이상의 PU들 및/또는 하나 이상의 TU들을 포함할 수도 있다. 비디오 코더는 PU들 및 TU들을 추가로 파티셔닝할 수도 있다. 예를 들어, HEVC 에서, 잔차 쿼드트리 (RQT) 는 TU들의 파티셔닝을 나타낸다. HEVC 에서, PU들은 인터-예측 데이터를 나타내는 한편, TU들은 잔차 데이터를 나타낸다. 인트라-예측되는 CU들은 인트라-모드 표시와 같은 인트라-예측 정보를 포함한다.
- [0042] 다른 예로서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 VVC 에 따라 동작하도록 구성될 수도 있다. VVC 에 따르면, 비디오 코더 (이를 테면 비디오 인코더 (200)) 는 픽처를 복수의 코딩 트리 유닛들 (CTU들) 로 파티셔닝한다. 비디오 인코더 (200) 는 쿼드트리 바이너리 트리 (QTBT) 구조 또는 멀티-타입 트리 (MTT) 구조와 같은 트리 구조에 따라 CTU 를 파티셔닝할 수도 있다. QTBT 구조는 HEVC 의 CU들, PU들, 및 TU들 간의 분리와 같은 다중의 파티션 타입들의 개념들을 제거한다. QTBT 구조는 2 개의 레벨들: 즉, 쿼드트리 파티셔닝에 따라 파티셔닝된 제 1 레벨, 및 바이너리 트리 파티셔닝에 따라 파티셔닝된 제 2 레벨을 포함한다. QTBT 구조의 루트 노드는 CTU 에 대응한다. 바이너리 트리들의 리프 노드들은 코딩 유닛들 (CU들) 에 대응한다.
- [0043] MTT 파티셔닝 구조에서, 블록들은 쿼드트리 (QT) 파티션, 바이너리 트리 (BT) 파티션, 및 하나 이상의 타입들의 트리플 트리 (TT) (터너리 (ternary) 트리 (TT) 로 또한 지칭됨) 파티션들을 사용하여 파티셔닝될 수도 있다. 트리플 또는 터너리 트리 파티션은, 블록이 3개의 서브블록들로 스플릿되는 파티션이다. 일부 예들에 있어서, 트리플 또는 터너리 트리 파티션은 중심을 통해 오리지널 블록을 분할하지 않고 블록을 3개의 서브블록들로 분할한다. MTT 에서의 파티셔닝 타입들 (예컨대, QT, BT, 및 TT) 은 대칭적이거나 비대칭적일 수도

있다.

- [0044] 일부 예들에 있어서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 및 크로미넌스 컴포넌트들의 각각을 나타내기 위해 단일의 QTBT 또는 MTT 구조를 사용할 수도 있는 한편, 다른 예들에 있어서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 컴포넌트에 대한 하나의 QTBT/MTT 구조 및 양자 모두의 크로미넌스 컴포넌트들에 대한 다른 QTBT/MTT 구조 (또는 개별의 크로미넌스 컴포넌트들에 대한 2 개의 QTBT/MTT 구조들) 와 같은 2 이상의 QTBT 또는 MTT 구조들을 사용할 수도 있다.
- [0045] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 HEVC 마다의 쿼드트리 파티셔닝, QTBT 파티셔닝, MTT 파티셔닝, 또는 다른 파티셔닝 구조들을 사용하도록 구성될 수도 있다. 설명의 목적들을 위해, 본 개시의 기법들의 설명은 QTBT 파티셔닝에 관하여 제시된다. 하지만, 본 개시의 기법들은 또한, 쿼드트리 파티셔닝, 또는 다른 타입들의 파티셔닝도 물론 사용하도록 구성된 비디오 코더들에 적용될 수도 있음을 이해해야 한다.
- [0046] 일부 예들에 있어서, CTU 는 루마 샘플들의 코딩 트리 블록 (CTB), 3개의 샘플 어레이들을 갖는 픽처의 크로마 샘플들의 2개의 대응하는 CTB들, 또는 샘플들을 코딩하는데 사용된 3개의 별도의 컬러 평면들 및 신택스 구조들을 사용하여 코딩되는 픽처 또는 모노크롬 픽처의 샘플들의 CTB 를 포함한다. CTB는 컴포넌트의 CTB 들로의 분할이 파티셔닝이 되도록 N의 일부 값에 대한 샘플들의 NxN 블록일 수 있다. 컴포넌트는, 4:2:0, 4:2:2, 또는 4:4:4 컬러 포맷으로 픽처를 구성하는 3개의 어레이들 (루마 및 2개의 크로마) 중 하나로부터의 단일 샘플 또는 어레이, 또는 모노크롬 포맷으로 픽처를 구성하는 어레이의 단일 샘플 또는 어레이이다. 일부 예들에 있어서, 코딩 블록은, CTB 의 코딩 블록들로의 분할이 파티셔닝이 되도록 M 및 N 의 일부 값들에 대한 샘플들의 MxN 블록이다.
- [0047] 블록들 (예컨대, CTU들 또는 CU들) 은 픽처에서 다양한 방식으로 그룹핑될 수도 있다. 일 예로서, 브릭 (brick) 은 픽처에서의 특정 타일 내의 CTU 행들의 직사각형 영역을 지칭할 수도 있다. 타일은 픽처에서의 특정 타일 열 및 특정 타일 행 내의 CTU들의 직사각형 영역일 수도 있다. 타일 열은, 픽처의 높이와 동일한 높이 및 (예컨대, 픽처 파라미터 세트에서와 같이) 신택스 엘리먼트들에 의해 명시된 폭을 갖는 CTU들의 직사각형 영역을 지칭한다. 타일 행은, (예컨대, 픽처 파라미터 세트에서와 같이) 신택스 엘리먼트들에 의해 명시된 높이 및 픽처의 폭과 동일한 폭을 갖는 CTU들의 직사각형 영역을 지칭한다.
- [0048] 일부 예들에 있어서, 타일은 다중의 브릭들로 파티셔닝될 수도 있으며, 그 각각은 타일 내의 하나 이상의 CTU 로우들을 포함할 수도 있다. 다중의 브릭들로 파티셔닝되지 않은 타일이 또한, 브릭으로서 지칭될 수도 있다. 그러나, 타일의 진정한 서브세트인 브릭은 타일로서 지칭되지 않을 수도 있다.
- [0049] 픽처에서의 브릭들은 또한 슬라이스로 배열될 수도 있다. 슬라이스는, 단일의 네트워크 추상화 계층 (NAL) 유닛에 배타적으로 포함될 수도 있는 픽처의 정수 개의 브릭들일 수도 있다. 일부 예들에서, 슬라이스는 다수의 완전한 타일들 또는 하나의 타일의 완전한 브릭들의 연속 시퀀스만을 포함한다.
- [0050] 본 개시는 수직 및 수평 차원들의 관점에서 블록 (예를 들어, CU 또는 다른 비디오 블록) 의 샘플 차원들을 지칭하기 위해 상호교환가능하게 "NxN" 및 "N 바이 N", 예를 들어, 16x16 샘플들 또는 16 바이 16 샘플들을 사용할 수도 있다. 일반적으로, 16x16 CU 는, 수직 방향에서 16개 샘플들 (y = 16) 그리고 수평 방향에서 16개 샘플들 (x = 16) 을 가질 것이다. 마찬가지로, N×N CU 는 일반적으로 수직 방향에서 N 샘플들 및 수평 방향에서 N 샘플들을 가지며, 여기서, N 은 음이 아닌 정수 값을 나타낸다. CU 에서의 샘플들은 행들 및 열들로 배열될 수도 있다. 더욱이, CU들은 수직 방향에서와 동일한 수의 샘플들을 수평 방향에서 반드시 가질 필요는 없다. 예를 들면, CU들은 NxM 샘플들을 포함할 수도 있고, 여기서 M 은 N 과 반드시 동일한 것은 아니다.
- [0051] 비디오 인코더 (200) 는 예측 및/또는 잔차 정보를 나타내는 CU들에 대한 비디오 데이터, 및 다른 정보를 인코딩한다. 예측 정보는, CU 에 대한 예측 블록을 형성하기 위하여 CU 가 어떻게 예측될지를 표시한다. 잔차 정보는 일반적으로, 인코딩 이전의 CU 의 샘플들과 예측 블록 사이의 샘플 별 차이들을 나타낸다.
- [0052] CU 를 예측하기 위해, 비디오 인코더 (200) 는 일반적으로 인터 예측 또는 인트라 예측을 통해 CU 에 대한 예측 블록을 형성할 수도 있다. 인터-예측은 일반적으로 이전에 코딩된 픽처의 데이터로부터 CU 를 예측하는 것을 지칭하는 반면, 인트라-예측은 일반적으로 동일한 픽처의 이전에 코딩된 데이터로부터 CU 를 예측하는 것을 지칭한다. 인터-예측을 수행하기 위해, 비디오 인코더 (200) 는 하나 이상의 모션 벡터들을 사용하여 예측 블록을 생성할 수도 있다. 비디오 인코더 (200) 는 일반적으로, 예컨대, CU 와 레퍼런스 블록 사이의 차이

들의 관점에서, CU 에 근접하게 매칭하는 레퍼런스 블록을 식별하기 위해 모션 탐색을 수행할 수도 있다. 비디오 인코더 (200) 는 절대 차이의 합 (SAD), 제곱 차이들의 합 (SSD), 평균 절대 차이 (MAD), 평균 제곱 차이들 (MSD), 또는 레퍼런스 블록이 현재 CU 에 근접하게 매칭하는지 여부를 결정하기 위한 다른 그러한 차이 계산들을 사용하여 차이 메트릭을 계산할 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 단방향 예측 또는 양방향 예측을 사용하여 현재 CU 를 예측할 수도 있다.

[0053] VVC 의 일부 예들은 또한, 인터 예측 모드로 고려될 수도 있는 아핀 모션 보상 모드를 제공한다. 아핀 모션 보상 모드에서, 비디오 인코더 (200) 는 줌 인 또는 아웃, 회전, 원근 모션, 또는 다른 불규칙한 모션 타입들과 같은 비-병진 모션을 나타내는 2 개 이상의 모션 벡터들을 결정할 수도 있다.

[0054] 인트라-예측을 수행하기 위해, 비디오 인코더 (200) 는 예측 블록을 생성하기 위해 인트라-예측 모드를 선택할 수도 있다. VVC 의 일부 예들은 다양한 방향성 모드들 뿐만 아니라 평면 모드 및 DC 모드를 포함하여 67 개의 인트라-예측 모드들을 제공한다. 일반적으로, 비디오 인코더 (200) 는 현재 블록의 샘플들을 예측하기 위한 현재 블록 (예를 들어, CU 의 블록) 에 대해 이웃하는 샘플들을 기술하는 인트라-예측 모드를 선택한다. 이러한 샘플들은 일반적으로, 비디오 인코더 (200) 가 래스터 스캔 순서로 (좌측에서 우측으로, 상단에서 하단으로) CTU들 및 CU들을 코딩한다고 가정하여, 현재 블록과 동일한 픽처에서 현재 블록의 상측, 상측 및 좌측으로, 또는 좌측으로 있을 수도 있다.

[0055] 비디오 인코더 (200) 는 현재 블록에 대한 예측 모드를 나타내는 데이터를 인코딩한다. 예를 들어, 인터-예측 모드의 경우, 비디오 인코더 (200) 는 다양한 이용가능한 인터-예측 모드들 중 어느 것이 사용되는지를 나타내는 데이터 뿐만 아니라, 대응하는 모드를 위한 모션 정보를 인코딩할 수도 있다. 단방향 또는 양방향 인터-예측을 위해, 예를 들어, 비디오 인코더 (200) 는 어드밴스드 모션 벡터 예측 (AMVP) 또는 병합 모드를 사용하여 모션 벡터들을 인코딩할 수도 있다. 비디오 인코더 (200) 는 유사한 모드들을 사용하여 아핀 모션 보상 모드에 대한 모션 벡터들을 인코딩할 수도 있다.

[0056] 블록의 인트라-예측 또는 인터-예측과 같은 예측에 후속하여, 비디오 인코더 (200) 는 블록에 대한 잔차 데이터를 계산할 수도 있다. 잔차 블록과 같은 잔차 데이터는 대응하는 예측 모드를 사용하여 형성된, 블록과 블록에 대한 예측 블록 사이의 샘플 별 차이들을 나타낸다. 비디오 인코더 (200) 는 샘플 도메인 대신에 변환 도메인에서 변환된 데이터를 생성하기 위해, 잔차 블록에 하나 이상의 변환들을 적용할 수도 있다. 예를 들어, 비디오 인코더 (200) 는 이산 코사인 변환 (DCT), 정수 변환, 웨이블릿 변환, 또는 개념적으로 유사한 변환을 잔차 비디오 데이터에 적용할 수도 있다. 부가적으로, 비디오 인코더 (200) 는 모드 의존적 비-분리가능 이차 변환 (MDNSST), 신호 의존적 변환, Karhunen-Loeve 변환 (KLT) 등과 같은 제 1 변환에 후속하여 이차 변환을 적용할 수도 있다. 비디오 인코더 (200) 는 하나 이상의 변환들의 적용에 후속하여 변환 계수들을 생성한다.

[0057] 상기 언급된 바와 같이, 변환 계수들을 생성하기 위한 임의의 변환들에 후속하여, 비디오 인코더 (200) 는 변환 계수들의 양자화를 수행할 수도 있다. 일반적으로 양자화는 변환 계수들이 그 변환 계수들을 나타내는데 사용되는 데이터의 양을 가능하게는 감소시키도록 양자화되어 추가 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스를 수행함으로써, 비디오 인코더 (200) 는 변환 계수들의 일부 또는 전부와 연관된 비트 심도를 감소시킬 수도 있다. 예를 들어, 비디오 인코더 (200) 는 양자화 동안 n -비트 값을 m -비트 값으로 라운딩 다운할 수도 있고, 여기서 n 은 m 보다 크다. 일부 예들에서, 양자화를 수행하기 위해, 비디오 인코더 (200) 는 양자화될 값의 비트단위 우측-시프트를 수행할 수도 있다.

[0058] 양자화에 후속하여, 비디오 인코더 (200) 는 변환 계수들을 스캔하여, 양자화된 변환 계수들을 포함한 2 차원 매트릭스로부터 1 차원 벡터를 생성할 수도 있다. 스캔은 더 높은 에너지 (및 따라서 더 낮은 주파수) 변환 계수들을 벡터의 전방에 배치하고 그리고 더 낮은 에너지 (및 따라서 더 높은 주파수) 변환 계수들을 벡터의 후방에 배치하도록 설계될 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 양자화된 변환 계수들을 스캔하기 위해 미리정의된 스캔 순서를 활용하여 직렬화된 벡터를 생성하고, 그 다음, 벡터의 양자화된 변환 계수들을 엔트로피 인코딩할 수도 있다. 다른 예들에서, 비디오 인코더 (200) 는 적응 스캔을 수행할 수도 있다.

1차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캐닝한 후, 비디오 인코더 (200) 는 예를 들어, 컨텍스트 적응 이진 산술 코딩 (CABAC) 에 따라 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (200) 는 또한, 비디오 데이터를 디코딩하는데 있어서 비디오 디코더 (300) 에 의한 사용을 위해 인코딩된 비디오 데이터와 연관된 메타데이터를 기술하는 신택스 엘리먼트들에 대한 값들을 엔트로피 인코딩할 수도 있다.

- [0059] CABAC 을 수행하기 위해, 비디오 인코더 (200) 는 송신될 심볼에 컨텍스트 모델 내의 컨텍스트를 할당할 수도 있다. 컨텍스트는 예를 들어, 심볼의 이웃하는 값들이 제로 값인지 여부와 관련될 수도 있다. 확률 결정은 심볼에 할당된 컨텍스트에 기초할 수도 있다.
- [0060] 비디오 인코더 (200) 는 선택스 데이터, 이를 테면 블록 기반 선택스 데이터, 픽처 기반 선택스 데이터, 및 시퀀스 기반 선택스 데이터를, 비디오 디코더 (300) 에, 예컨대, 픽처 헤더, 블록 헤더, 슬라이스 헤더, 또는 다른 선택스 데이터, 이를 테면 시퀀스 파라미터 세트 (SPS), 픽처 파라미터 세트 (PPS), 또는 비디오 파라미터 세트 (VPS) 에서 추가로 생성할 수도 있다. 마찬가지로, 비디오 디코더 (300) 는 대응하는 비디오 데이터를 디코딩하는 방법을 결정하기 위해 그러한 선택스 데이터를 디코딩할 수도 있다.
- [0061] 이러한 방식으로, 비디오 인코더 (200) 는 인코딩된 비디오 데이터, 예컨대, 픽처의 블록들 (예컨대, CU들) 로의 파티셔닝을 기술하는 선택스 엘리먼트들 및 블록들에 대한 예측 및/또는 잔차 정보를 포함하는 비트스트림을 생성할 수도 있다. 궁극적으로, 비디오 디코더 (300) 는 비트스트림을 수신하고, 인코딩된 비디오 데이터를 디코딩할 수도 있다.
- [0062] 일반적으로, 비디오 디코더 (300) 는 비트스트림의 인코딩된 비디오 데이터를 디코딩하기 위해 비디오 인코더 (200) 에 의해 수행되는 것과 가역적인 프로세스를 수행한다. 예를 들어, 비디오 디코더 (300) 는 비디오 인코더 (200) 의 CABAC 인코딩 프로세스와 실질적으로 유사하지만 가역적인 방식으로 CABAC 을 사용하여 비트스트림의 선택스 엘리먼트들에 대한 값들을 디코딩할 수도 있다. 선택스 엘리먼트들은 픽처의 CTU들로의 파티셔닝, 및 QTBT 구조와 같은 대응하는 파티션 구조에 따른 각각의 CTU 의 파티셔닝을 위한 파티셔닝 정보를 정의하여, CTU 의 CU들을 정의할 수도 있다. 선택스 엘리먼트들은 비디오 데이터의 블록들 (예컨대, CU들) 에 대한 예측 및 잔차 정보를 추가로 정의할 수도 있다.
- [0063] 잔차 정보는, 예를 들어, 양자화된 변환 계수들에 의해 표현될 수도 있다. 비디오 디코더 (300) 는 블록에 대한 잔차 블록을 재생하기 위해 블록의 양자화된 변환 계수들을 역양자화 및 역변환할 수도 있다. 비디오 디코더 (300) 는 시그널링된 예측 모드 (인트라- 또는 인터-예측) 및 관련된 예측 정보 (예컨대, 인터-예측을 위한 모션 정보) 를 사용하여 블록에 대한 예측 블록을 형성한다. 그 다음, 비디오 디코더 (300) 는 예측 블록과 잔차 블록을 (샘플 별 기준으로) 결합하여 원래 블록을 재생할 수도 있다. 비디오 디코더 (300) 는 블록의 경계들을 따라 시각적 아티팩트들을 감소시키기 위해 디블록킹 프로세스를 수행하는 것과 같은 추가적인 프로세싱을 수행할 수도 있다.
- [0064] 본 개시는 일반적으로 선택스 엘리먼트들과 같은 소정의 정보를 "시그널링" 하는 것을 참조할 수도 있다. 용어 "시그널링" 은 일반적으로, 인코딩된 비디오 데이터를 디코딩하는데 사용되는 선택스 엘리먼트들에 대한 값들 및/또는 다른 데이터의 통신을 지칭할 수도 있다. 즉, 비디오 인코더 (200) 는 비트스트림에서 선택스 엘리먼트들에 대한 값들을 시그널링할 수도 있다. 일반적으로, 시그널링은 비트스트림에서 값을 생성하는 것을 지칭한다. 상기 언급된 바와 같이, 소스 디바이스 (102) 는 목적지 디바이스 (116) 에 의한 추후 취출을 위해 저장 디바이스 (112) 에 선택스 엘리먼트들을 저장할 때 발생할 수도 있는 바와 같이, 비실시간으로 또는 실질적으로 실시간으로 비트스트림을 목적지 디바이스 (116) 로 전송할 수도 있다.
- [0065] 도 2a 및 도 2b 는 예시적인 쿼드트리 바이너리 트리 (QTBT) 구조 (130) 및 대응하는 코딩 트리 유닛 (CTU) (132) 을 예시하는 개념 다이어그램들이다. 실선들은 쿼드트리 스플리팅을 나타내고, 점선들은 이진 트리 스플리팅을 나타낸다. 이진 트리의 각각의 스플리팅된 (즉, 비-리프) 노드에서, 어느 스플리팅 타입 (즉, 수평 또는 수직) 이 사용되는지를 표시하기 위해 하나의 플래그가 시그널링되고, 이 예에서 0 은 수평 스플리팅을 표시하고 1 은 수직 스플리팅을 표시한다. 쿼드트리 스플리팅에 대해, 쿼드트리 노드들은 블록을 동일한 사이즈를 갖는 4 개의 서브-블록들로 수평으로 및 수직으로 스플리팅하기 때문에 스플리팅 타입을 표시할 필요가 없다. 이에 따라, QTBT 구조 (130) 의 영역 트리 레벨 (즉, 실선들) 에 대한 (스플리팅 정보와 같은) 선택스 엘리먼트들 및 QTBT 구조 (130) 의 예측 트리 레벨 (즉, 점선들) 에 대한 (스플리팅 정보와 같은) 선택스 엘리먼트들을, 비디오 인코더 (200) 가 인코딩할 수도 있고 비디오 디코더 (300) 가 디코딩할 수도 있다. QTBT 구조 (130) 의 종단 리프 노드들에 의해 표현된 CU들에 대한, 예측 및 변환 데이터와 같은 비디오 데이터를, 비디오 인코더 (200) 가 인코딩할 수도 있고 비디오 디코더 (300) 가 디코딩할 수도 있다.
- [0066] 일반적으로, 도 2b 의 CTU (132) 는 제 1 및 제 2 레벨들에서 QTBT 구조 (130) 의 노드들에 대응하는 블록들의 사이즈들을 정의하는 파라미터들과 연관될 수도 있다. 이들 파라미터들은 CTU 사이즈 (샘플들에서 CTU (132) 의 사이즈를 나타냄), 최소 쿼드트리 사이즈 (MinQTSIZE, 최소 허용된 쿼드트리 리프 노드 사이즈를 나타냄), 최대 바이너리 트리 사이즈 (MaxBTSIZE, 최대 허용된 바이너리 트리 루트 노드 사이즈를 나타냄), 최대 바

이너리 트리 깊이 (MaxBTDDepth, 최대 허용된 바이너리 트리 깊이를 나타냄), 및 최소 바이너리 트리 사이즈 (MinBTSize, 최소 허용된 바이너리 트리 리프 노드 사이즈를 나타냄) 를 포함할 수도 있다.

[0067] CTU 에 대응하는 QTBT 구조의 루트 노드는 QTBT 구조의 제 1 레벨에서 4 개의 자식 노드들을 가질 수도 있으며, 이들의 각각은 쿼드트리 파티셔닝에 따라 파티셔닝될 수도 있다. 즉, 제 1 레벨의 노드들은 리프 노드들 (자식 노드들을 갖지 않음) 이거나 또는 4 개의 자식 노드들을 갖는다. QTBT 구조 (130) 의 예는 그러한 노드들을, 브랜치들에 대한 실선들을 갖는 자식 노드들 및 부모 노드를 포함하는 것으로서 나타낸다. 제 1 레벨의 노드들이 최대 허용된 이진 트리 루트 노드 사이즈 (MaxBTSize) 보다 크지 않으면, 그 노드들은 개별의 이진 트리들에 의해 추가로 파티셔닝될 수 있다. 하나의 노드의 이진 트리 스플리팅은, 스플리팅으로부터 발생하는 노드들이 최소 허용된 이진 트리 리프 노드 사이즈 (MinBTSize) 또는 최대 허용된 이진 트리 심도 (MaxBTDDepth) 에 도달할 때까지 반복될 수 있다. QTBT 구조 (130) 의 예는 그러한 노드들을, 브랜치들에 대한 점선들을 갖는 것으로서 나타낸다. 바이너리 트리 리프 노드는, 어떠한 추가의 파티셔닝도 없이, 예측 (예컨대, 인트라-픽처 또는 인터-픽처 예측) 및 변환을 위해 사용되는 코딩 유닛 (CU) 으로서 지칭된다. 상기 논의된 바와 같이, CU들은 또한, "비디오 블록들" 또는 "블록들" 로서 지칭될 수도 있다.

[0068] QTBT 파티셔닝 구조의 일 예에 있어서, CTU 사이즈는 128x128 (루마 샘플들 및 2 개의 대응하는 64x64 크로마 샘플들) 로서 설정되고, MinQTSize 는 16x16 으로서 설정되고, MaxBTSize 는 64x64 로서 설정되고, (폭 및 높이 양자 모두에 대한) MinBTSize 는 4 로서 설정되고, 그리고 MaxBTDDepth 는 4 로서 설정된다. 쿼드트리 파티셔닝은 쿼드-트리 리프 노드들을 생성하기 위해 먼저 CTU 에 적용된다. 쿼드트리 리프 노드들은 16x16 (즉, MinQTSize) 으로부터 128x128 (즉, CTU 사이즈) 까지의 사이즈를 가질 수도 있다. 쿼드트리 리프 노드가 128x128 이면, 사이즈가 MaxBTSize (즉, 이 예에서 64x64) 를 초과하기 때문에, 리프 쿼드트리 노드는 이진 트리에 의해 추가로 스플리팅되지 않을 것이다. 그렇지 않으면, 쿼드트리 리프 노드는 이진 트리에 의해 추가로 파티셔닝될 것이다. 따라서, 쿼드트리 리프 노드는 또한 이진 트리에 대한 루트 노드이고, 이진 트리 심도를 0 으로서 갖는다. 이진 트리 심도가 MaxBTDDepth (이 예에서는 4) 에 도달할 때, 추가의 스플리팅이 허용되지 않는다. MinBTSize (이 예에서는 4) 와 동일한 폭을 갖는 이진 트리 노드는, 그 이진 트리 노드에 대해 추가의 수직 스플리팅 (즉, 폭의 분할) 이 허용되지 않음을 암시한다. 유사하게, MinBTSize 와 동일한 높이를 갖는 이진 트리 노드는, 그 이진 트리 노드에 대해 추가의 수평 스플리팅 (즉, 높이의 분할) 이 허용되지 않음을 암시한다. 상기 언급된 바와 같이, 이진 트리의 리프 노드들은 CU들로서 지칭되고, 추가의 파티셔닝 없이 예측 및 변환에 따라 추가로 프로세싱된다.

[0069] VVC 드래프트 7 은 DPB 가 상이한 계층들에 걸쳐 공통적이거나 공유되는 다중 계층 코딩을 지원한다. DPB 연산은 서브-DPB들의 관점에서 표현되며, 여기서 서브-DPB는 VVC 드래프트 7 의 섹션 C.3.1 에서 다음과 같이 정의된다.

[0070] DPB 는 개념적으로 서브-DPB들로 구성되고, 각각의 서브-DPB 는 하나의 계층의 디코딩된 픽처들의 저장을 위한 픽처 저장 버퍼들을 포함한다. 픽처 저장 버퍼들 각각은 "참조를 위해 사용됨" 으로 마킹되거나 장래의 출력을 위해 유지되는 디코딩된 픽처를 포함한다.

[0071] 픽처 출력 프로세스는 각 계층 대해 독립적으로 호출된다. VVC 드래프트 7 의 섹션 C.5.2.1 은 이 프로세스를 다음과 같이 설명한다:

[0072] 조항 C.5.2.2 에 명시된 바와 같은 현재 픽처의 디코딩 전에 DPB 로부터의 픽처들의 출력 및 제거를 위한 프로세스가 호출되고, 이어서 조항 C.3.4 에 명시된 바와 같은 현재 디코딩된 픽처 마킹 및 저장을 위한 프로세스의 호출이 후속되고, 마지막으로 조항 C.5.2.3 에 명시된 바와 같은 추가적인 범핑을 위한 프로세스의 호출이 후속된다. "범핑" 프로세스는 조항 C.5.2.4 에 명시되고 조항 C.5.2.2 및 C.5.2.3 에 명시된 바와 같이 호출된다.

[0073] 이들 프로세스는 OLS 의 가장 낮은 계층으로부터 시작하여, OLS 의 계층들의 nuh_layer_id 값들의 증가하는 순서로, 각각의 계층에 대해 독립적으로 적용된다. 이들 프로세스들이 특정 계층에 대해 적용될 때, 특정 계층에 대한 서브-DPB만이 영향을 받는다.

[0074] DPB 파라미터들은 하기에 재생되는 dpb_parameters() 신택스 구조에서 시그널링된다

dpb_parameters(dpbSizeOnlyFlag, maxSubLayersMinus1, subLayerInfoFlag)	디스크립터
{	
for(i = (subLayerInfoFlag ? 0 : maxSubLayersMinus1);	
i <= maxSubLayersMinus1; i++) {	
max_dec_pic_buffering_minus1[i]	ue(v)
if(!dpbSizeOnlyFlag) {	
max_num_reorder_pics[i]	ue(v)
max_latency_increase_plus1[i]	ue(v)
}	
}	
}	

[0075]

[0076] VVC 드래프트 7 은 신택스 엘리먼트 "max_dec_pic_buffering_minus1[i]" 를 다음과 같이 정의한다 - max_dec_pic_buffering_minus1[i] 플러스 1 은, CVS 의 각각의 CLVS 에 대해, Htid 가 i 와 동일할 때 픽처 저장 버퍼들의 유닛들로 DPB 의 최대 요구된 사이즈를 특정한다. max_dec_pic_buffering_minus1[i] 의 값은 0 내지 MaxDpbSize - 1 을 포함하는 범위에 있을 것이며, 여기서 MaxDpbSize 는 조항 A.4.2 에 명시된 바와 같다. i 가 0 보다 클 때, max_dec_pic_buffering_minus1[i] 는 max_dec_pic_buffering_minus1[i - 1] 보다 크거나 동일할 것이다. max_dec_pic_buffering_minus1[i] 가 0 내지 maxSubLayersMinus1 - 1 을 포함하는 범위 내의 i 에 대해 존재하지 않을 때, subLayerInfoFlag 가 0 과 동일하기 때문에, max_dec_pic_buffering_minus1[maxSubLayersMinus1] 와 동일한 것으로 추론된다.

[0077] VVC 드래프트 7 은 신택스 엘리먼트 "max_num_reorder_pics[i]" 를 다음과 같이 정의한다 - max_num_reorder_pics[i] 는 CVS 의 각각의 CLVS 에 대해, 디코딩 순서에서 CLVS 에서의 임의의 픽처를 선행하고 Htid 가 i 와 동일할 때 출력 순서에서 그 픽처를 후속할 수 있는 CLVS 의 픽처들의 최대 허용된 수를 특정한다. max_num_reorder_pics[i] 의 값은 0 내지 max_dec_pic_buffering_minus1[i] 을 포함하는 범위에 있을 것이다. i 가 0 보다 클 때, max_num_reorder_pics[i] 는 max_num_reorder_pics[i - 1] 보다 크거나 동일할 것이다. max_num_reorder_pics[i] 가 0 내지 maxSubLayersMinus1 - 1 을 포함하는 범위 내의 i 에 대해 존재하지 않을 때, subLayerInfoFlag 가 0 과 동일하기 때문에, max_num_reorder_pics[maxSubLayersMinus1] 와 동일한 것으로 추론된다.

[0078] VVC 드래프트 7 은 신택스 엘리먼트 "max_latency_increase_plus1[i]" 를 다음과 같이 정의한다 - 0 과 동일하지 않은 max_latency_increase_plus1[i] 이 MaxLatencyPictures[i] 의 값을 컴퓨팅하기 위해 사용되며, 이는 CVS 의 각각의 CLVS 에 대해, 출력 순서에서 CLVS 에서의 임의의 픽처를 선행하고 Htid 가 i 와 동일할 때 디코딩 순서에서 그 픽처를 후속할 수 있는 CLVS 에서의 픽처들의 최대 허용된 수를 특정한다.

[0079] max_latency_increase_plus1[i] 가 0 과 동일하지 않을 때, MaxLatencyPictures[i] 의 값은 다음과 같이 특정된다:

[0080]
$$\text{MaxLatencyPictures}[i] = \text{max_num_reorder_pics}[i] + \text{max_latency_increase_plus1}[i] - 1$$
 (7-73)

[0081] max_latency_increase_plus1[i] 가 0 과 동일할 때, 대응하는 제한이 표현되지 않는다.

[0082] max_latency_increase_plus1[i] 의 값은 0 내지 $2^{32} - 2$ 을 포함하는 범위에 있을 것이다. max_latency_increase_plus1[i] 가 0 내지 maxSubLayersMinus1 - 1 을 포함하는 범위 내의 i 에 대해 존재하지 않을 때, subLayerInfoFlag 가 0 과 동일하기 때문에, max_latency_increase_plus1[maxSubLayersMinus1] 와 동일한 것으로 추론된다.

[0083] VVC 드래프트 7 의 DPB 구현은 몇몇 문제들을 가질 수도 있다. 일 예로서, VVC 드래프트 7 에서, DPB 동작은 서브-DPB들 및 서브-DPB 충만도를 사용하여 표현된다. 그러나, VVC 드래프트 7 은 서브-DPB 충만도를 정의하지 않고, 서브-DPB 충만도가 dpb_parameters() 픽처 구조에서 특정된 DPB 충만도에 어떻게 관련되는지를 특정하지 않는다.

[0084] 또한, 서브-DPB 충만도를 어떻게 체크해야 하는지와 같이, 서브-DPB 에 대한 동작은 특정되지 않는다. 특정 서브-DPB 를 최대 DPB 사이즈 (MaxDpbSize) 와 비교하는 것은 다수의 서브-DPB들이 존재할 수 있기 때문에 충분

하지 않을 수도 있고, 이들 서브-DPB들은 특정된 DPB 리소스를 초과할 수도 있다.

- [0085] VVC 드래프트 7 의 섹션 C.5.2.1 에서, DPB 로부터의 픽처들의 출력 및 제거는 특정 계층에 대한 서브-DPB만이 영향을 받는 각각의 계층에 대해 호출된다. 그러나, DPB 사이즈는 서브-DPB 에 대해 정의되지 않으므로, 예를 들어, "DPB 내의 픽처들의 수가 $\text{max_dec_pic_buffering_minus1}[\text{Htid}] + 1$ 보다 크거나 동일하다" 는 조건이 서브-DPB 에 대해 어떻게 체크되어야 하는지가 불명확하다.
- [0086] 본 개시물은 전술한 문제들을 해결할 수도 있는 기법들을 기술하였다. 본 개시물에서 제공된 해결책들은 독립적으로 또는 임의의 조합으로 사용될 수 있다.
- [0087] 여기에 제공된 설명의 부분들은 $\text{max_dec_pic_buffering_minus1}$ 신택스 엘리먼트에 의해 특정되는 DPB 사이즈를 사용하는 예에 관하여 설명된다. 그러나, 본 개시물의 기법들은 또한 $\text{max_num_reorder_pics}$ 및 $\text{max_latency_increase_plus1}$ 과 같은 $\text{dpb_parameters}()$ 신택스 구조에서 시그널링된 다른 파라미터와 함께 사용될 수도 있다. 모든 이들 파라미터들은 서브-DPB 에 대해서가 아니라 전체 DPB 에 대해 시그널링되고, 따라서 VVC 드래프트 7 에서 현재 이용되는 서브-DPB 조건 체크들은 시그널링된 $\text{dpb_parameters}()$ 신택스 엘리먼트들을 이용하여 전체 DPB 에 대해 수행될 수도 있다.
- [0088] DPB 에 대해 설명된 개시된 기법들은 서브-DPB 가 사용되는 다른 동작들에서 적용가능할 수도 있다. 예를 들어, 코딩된 픽처 버퍼 (CPB) 또는 가상 레퍼런스 디코더 (HRD) 프로세싱을 위해.
- [0089] 다수의 계층들에 대해 공유된 디코더 픽처 버퍼를 이용하는 것의 일부로서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 다음의 기법들을 수행하도록 구성될 수도 있다.
- [0090] 서브-DPB들이 사용될 때, 전체 DPB 층만도는 VVC 드래프트 7 에서 초기화되지 않는다. 본 개시물의 일부 예들에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는, 비디오 시퀀스의 제 1 픽처가 파싱될 때, DPB 층만도를 0 으로 초기화하도록 구성될 수도 있다. 일 예에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 가장 낮은 nuh_layer_id 를 갖는 CLVSS 픽처 0 의 제 1 슬라이스가 파싱될 때 DPB 층만도를 0 으로 초기화하도록 구성될 수도 있다.
- [0091] 현재 계층 픽처에 DPB 픽처 마킹 프로세스가 적용된다. 그러나, DPB 에서의 다른 nuh_layer_id 픽처들의 상태는 VVC 드래프트 7 에서 특정되지 않는다. 본 개시물의 일부 예들에서, 픽처 상태 또는 스테이터스는 "단기 참조를 위해 사용됨", "장기 참조를 위해 사용됨", "인터-계층 예측을 위해 사용됨" 등과 같은 픽처 마킹을 지칭하고, 대안적으로 또는 추가적으로, 픽처 "출력을 위해 사용됨", 픽처 "출력을 위해 필요하지 않음" 등과 같은 픽처 출력 상태를 포함할 수도 있다.
- [0092]
 - 일부 예들에 대해, 다른 계층 픽처들의 상태를 변화시키지 않고 유지하는 것이 바람직할 수도 있으며, 즉, 그 상태는 이전의 액세스 유닛 디코딩 후에 DPB 에서 유지된다. 즉, 그 다른 nuh_layer_id 의 이전 픽처의 상태가 유지된다. 이 경우, 다른 계층 픽처가 레퍼런스 픽처인 경우, "범핑" 프로세스에 의해 제거되지 않을 수도 있다.
- [0093]
 - 일부 예들에서, 모든 픽처들은 현재 액세스 유닛의 제 1 픽처가 디코딩될 때 참조를 위해 사용되는 것 (예를 들어, 단기 참조를 위해 사용되는 것, 장기 참조를 위해 사용되는 것) 으로서 마킹될 수 있다.
- [0094]
 - 일부 예들에서, "범핑" 프로세스 (C.5.2.4) 에서, 픽처가 현재 픽처 계층 id 와 동일한 계층 id 를 갖는 경우에만 픽처가 제거된다 (비워진다). 이 경우, 다른 계층들의 픽처는 출력되지만, 이들이 참조되지 않을 때에도 DPB 로부터 제거되지 않을 수도 있고, 그러한 픽처들의 제거는 그 계층의 픽처들이 디코딩될 때 발생할 것이다.
- [0095] 일 예에서, MaxDpbSize 는 서브-DPB에 대해 정의되지 않고 출력 계층 세트 (OLS) 에서의 각각의 계층에 대해 정의되기 때문에, DPB 층만도는 서브-DPB 층만도의 합으로서 정의될 수도 있다. 예를 들어, MaxDpbSize 는 프로파일/레벨/티어에 의해 픽처 저장 버퍼들의 최대 수로서 특정될 수도 있다. 추가적으로, 서브-DPB 사이즈가 정의되거나 시그널링되면, 서브-DPB 사이즈들의 합은 MaxDpbSize 를 초과하지 않을 것이다. 일 예에서, 이러한 제약은 다음과 같이 표현될 수도 있다:
- [0096] AU 에 포함된 모든 계층들에 대한 $\text{max_dec_pic_buffering_minus1}[i]$ 의 합이 0 내지 $\text{MaxDpbSize} - 1$ 를 포함하는 범위에 있는 것이 비트스트림 적합성의 요건이다.

- [0097] 일부 예들에서, 제약은 다음과 같이 표현될 수도 있다:
- [0098] OLS 에 포함된 모든 계층들에 대한 $\text{max_dec_pic_buffering_minus1}[i]$ 의 합이 0 내지 $\text{MaxDpbSize} - 1$ 를 포함하는 범위에 있는 것이 비트스트림 적합성의 요건이다.
- [0099] 출력 및 제거 픽처 프로세스들에서, 각각의 계층에 대한 모든 동작들을 독립적으로 수행하는 대신에, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 프로세스가 계층마다 호출되더라도 계층마다 일부 동작들 및 계층들에 걸쳐 전체 DPB 에 대한 일부 동작들을 수행하도록 구성될 수도 있다. 예를 들어, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 적어도 제 1 계층 및 제 2 계층을 포함하는 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 DPB 를 유지하도록 구성될 수도 있다. 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 섹션 C.5.2.2 에 대해 아래에서 설명되는 것과 같은 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하도록 구성될 수도 있다. CPB 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 섹션들 C.5.2.3 및 C.5.2.4 에 대해 아래에서 설명되는 바와 같이 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하도록 구성될 수도 있다.
- [0100] 일 예에서, 픽처 출력 및 제거 및 추가적인 범핑 프로세스들이 계층마다 호출되고, 여기서 코딩된 계층 비디오 시퀀스 시작 (CLVSS) 픽처들 (인트라 랜덤 액세스 포인트 (IRAP) 픽처들) 에 대해, 서브-DPB 는 비워지는 반면, "범핑" 프로세스 C.5.2.4 는 계층들에 걸쳐 전체 DPB 에 대해 동작하고 있는데, 이는 AU 가 불안전할 (모든 계층 픽처들이 존재하는 것은 아님) 경우일 수 있지만, 현재 AU 에 존재하는 계층들의 픽처들 전에 다른 계층들의 픽처들이 출력될 필요가 있을 수도 있기 때문이다.
- [0101] 서브-DPB 가 계층마다 비워질 때, 즉 특정 계층 ID 를 갖는 픽처들이 비워질 때, 픽처 "범핑" 프로세스는 모든 계층들의 픽처들을 포함하는 전체 DPB 에 대해 호출된다.
- [0102] 유사한 방식으로, DPB 동작의 프로세스들 (예를 들어, 섹션 C.3) 은 계층에 대해 또는 계층마다 호출되지만, 픽처들이 출력되거나 비워질 때, 그러한 프로세스들은 DPB 에서의 모든 픽처들에 대해, 예를 들어 계층들에 걸쳐 호출된다.
- [0103] 일 예에서, 이 기술은 VVC 드래프트 7 로부터의 관련 섹션들로부터 "이들 프로세스들이 특정 계층에 대해 적용될 때, 오직 특정 계층에 대한 서브-DPB 만이 영향을 받는다" 는 조건을 제거함으로써 구현될 수도 있다.
- [0104] 일부 예들에서, 픽처 제거는 각각의 계층 내에서만 적용될 수도 있는 반면, 픽처 출력은 상이한 계층들에 걸쳐 적용될 수도 있다.
- [0105] 전술한 기법들을 수행하는 것의 일부로서, 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하고 - 여기서 DPB 는 복수의 계층들 중의 계층에 대한 레퍼런스 픽처들을 저장하기 위한 서브-DPB 를 포함함 -; 비디오 데이터의 픽처를 디코딩하고; 그리고 디코딩된 픽처의 카피를 레퍼런스 픽처로서 DPB 에 저장하도록 구성될 수도 있다.
- [0106] 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 또한 서브-DPB 의 충만도를 결정하고 및/또는 DPB 의 충만도를 결정하도록 구성될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 또한, 비디오 시퀀스의 제 1 픽처가 파싱되는 것에 응답하여 서브-DPB 의 충만도를 0 으로 초기화하고 및/또는 가장 낮은 nuh_layer_id 를 갖는 CLVSS 픽처 0 의 제 1 슬라이스가 파싱되는 것에 응답하여 서브-DPB 의 충만도를 0 으로 초기화하도록 구성될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 또한 서브-DPB 에 저장된 픽처들을 상태로 마킹하도록 구성될 수도 있다.
- [0107] VVC 드래프트 7 을 사용하는 일 예에서, 본 개시의 기법들은 픽처 출력 및 제거 설명에 대한 다음의 수정들로 구현될 수도 있다. 태그 "<ADD>" 는 본 개시물에 의해 제안된 추가들의 시작을 나타내고, 태그 "</ADD>" 는 그 추가들의 끝을 나타낸다. 태그 "" 는 본 개시물에 의해 제안된 삭제들 (즉, 텍스트 제거) 의 시작을 나타내고, 태그 "" 는 그 삭제들의 끝을 나타낸다.
- [0108] 다음의 설명에서, AU 는 상이한 계층들에 속하고 DPB 로부터의 출력을 위해 동일한 시간과 연관된 코딩된 픽처들을 포함하는 PU들의 세트를 나타낸다. 디코딩 유닛 (DU) 은 $\text{DecodingUnitHrdFlag}$ 가 0 과 동일한 경우 AU 를 지칭하거나, 그렇지 않으면 AU 의 서브세트를 지칭하며, AU 내의 하나 이상의 VCL NAL 유닛들 및 연관된 비-VCL NAL 유닛들로 구성된다.
- [0109] **8.3.3 레퍼런스 픽처 마킹을 위한 디코딩 프로세스**

- [0110] 이 프로세스는, 조항 8.3.2 에서 특정된 바와 같이 슬라이스 헤더의 디코딩 및 슬라이스에 대한 레퍼런스 픽처 리스트 구성을 위한 디코딩 프로세스 후에, 그러나 슬라이스 데이터의 디코딩 전에, 픽처마다 한 번씩 호출된다. 이 프로세스는 DPB 의 하나 이상의 레퍼런스 픽처들이 "참조를 위해 사용되지 않음" 또는 "장기 참조를 위해 사용됨" 으로 마킹되게 할 수도 있다.
- [0111] DPB 에서의 디코딩된 픽처는 "참조를 위해 사용되지 않음", "단기 참조를 위해 사용됨" 또는 "장기 참조를 위해 사용됨" 으로서 마킹될 수 있지만, 디코딩 프로세스의 동작 동안 임의의 주어진 순간에 이들 3 개 중 단지 하나만이 마킹된다. 이들 마킹들 중 하나를 픽처에 할당하는 것은 적용가능하다면 이들 마킹들 중 다른 하나를 암시적으로 제거한다. 픽처가 "참조를 위해 사용됨" 으로 마킹되는 것으로 지칭될 때, 이는 "단기 참조를 위해 사용됨" 또는 "장기 참조를 위해 사용됨" 으로 마킹되는 픽처를 총체적으로 지칭한다 (둘 모두는 아님).
- [0112] STRP들 및 ILRP들은 그들의 nuh_layer_id 및 PicOrderCntVal 값들에 의해 식별된다. LTRP들은 그들의 nuh_layer_id 값들 및 그들의 PicOrderCntVal 값들의 $\text{Log}_2(\text{MaxLtPicOrderCntLsb})$ LSB들에 의해 식별된다.
- [0113] 현재 픽처가 CLVSS 픽처이면, 현재 픽처와 동일한 nuh_layer_id 를 갖는 DPB (임의의 경우) 에서의 현재 모든 레퍼런스 픽처들은 "참조를 위해 사용되지 않음" 으로 마킹된다.
- [0114] 그렇지 않으면, 다음이 적용된다:
- [0115] - RefPicList[0] 또는 RefPicList[1] 의 각 LTRP 엔트리에 대해, 픽처가 현재 픽처와 동일한 nuh_layer_id 를 갖는 STRP 일 때, 그 픽처는 "장기 참조를 위해 사용됨" 으로 마킹된다.
- [0116] - RefPicList[0] 또는 RefPicList[1] 의 임의의 엔트리에 의해 참조되지 않는 DPB 의 현재 픽처와 동일한 nuh_layer_id 를 갖는 각각의 레퍼런스 픽처는 "참조를 위해 사용되지 않음" 으로 마킹된다.
- [0117] - RefPicList[0] 또는 RefPicList[1] 의 각 ILRP 엔트리에 대해, 픽처는 "장기 참조를 위해 사용됨" 으로 마킹된다.
- [0118] <ADD>현재 픽처 nuh_layer_id 와 상이한 DPB 의 픽처들은 이전 픽처 마킹에서 DPB 상태를 유지한다.</ADD>
- [0119] 일부 예들에서, 현재 픽처의 레퍼런스 픽처 마킹이 현재 픽처의 레퍼런스 픽처 마킹과 동일하지 않은 nuh_layer_id 를 갖는 픽처들의 레퍼런스 픽처 마킹에 영향을 미치지 않는다고 특정될 수도 있다.
- [0120] **C.5.2 출력 순서 DPB 의 동작**
- [0121] **C.5.2.1 일반**
- [0122] 이 조항의 사양은 조항 C.1 에 명시된 바와 같이 선택된 DPB 파라미터들의 각 세트에 독립적으로 적용된다.
- [0123] DPB 는 개념적으로 서브-DPB들로 구성되고, 각각의 서브-DPB 는 하나의 계층의 디코딩된 픽처들의 저장을 위한 픽처 저장 버퍼들을 포함한다. 픽처 저장 버퍼들 각각은 "참조를 위해 사용됨" 으로 마킹되거나 장래의 출력을 위해 유지되는 디코딩된 픽처를 포함한다.
- [0124] 조항 C.5.2.2 에 명시된 바와 같은 현재 픽처의 디코딩 전에 DPB 로부터의 픽처들의 출력 및 제거를 위한 프로세스가 호출되고, 이어서 조항 C.3.4 에 명시된 바와 같은 현재 디코딩된 픽처 마킹 및 저장을 위한 프로세스의 호출이 후속되고, 마지막으로 조항 C.5.2.3 에 명시된 바와 같은 추가적인 범핑을 위한 프로세스의 호출이 후속된다. "범핑" 프로세스는 조항 C.5.2.4 에 명시되고 조항 C.5.2.2 및 C.5.2.3 에 명시된 바와 같이 호출된다.
- [0125] 이러한 프로세스들 <ADD>조항 C.5.2.2 "Output and removal of pictures from the DPB" 및 조항 C.5.2.3 "Additional bumping" </ADD> 은 OLS 에서 가장 낮은 레이어부터 시작하여, OLS 에서 계층들의 nuh_layer_id 값이 증가하는 순서로 각 계층에 대해 독립적으로 적용된다. 이들 프로세스들이 특정 계층에 대해 적용될 때, 특정 계층에 대한 서브-DPB만이 영향을 받는다. <ADD>조항 C.5.2.4 은 DPB 에서 임의의 계층의 모든 픽처들에 적용된다.</ADD>
- [0126] 참고 - 출력 타이밍 DPB 의 동작에서와 동일한 출력 순서 DPB 의 동작에서, 동일한 액세스 유닛에서 1 과 동일한 PicOutputFlag 를 갖는 디코딩된 픽처들은 또한 디코딩된 픽처들의 nuh_layer_id 값들의 오름차순으로 연속적으로 출력된다.
- [0127] 픽처 n 및 현재 픽처를 nuh_layer_id 의 특정 값에 대한 액세스 유닛 n 의 코딩된 픽처 또는 디코딩된 픽처라고

하며, 여기서 n 은 음이 아닌 정수이다.

- [0128] **C.5.2.2 DPB 에서 픽처들의 출력 및 제거**
- [0129] 현재 픽처의 디코딩 전에 (그러나 현재 픽처의 제 1 슬라이스의 슬라이스 헤더를 파싱한 후에) DPB로부터의 픽처들의 출력 및 제거는, 현재 픽처를 포함하는 AU 의 제 1 DU 가 CPB 로부터 제거될 때 순시적으로 발생하고, 다음과 같이 진행된다:
- [0130] - 조항 8.3.2 에 명시된 레퍼런스 픽처 리스트 구성을 위한 디코딩 프로세스 및 조항 8.3.3 에 명시된 레퍼런스 픽처 마킹을 위한 디코딩 프로세스가 호출된다.
- [0131] - 현재 픽처가 픽처 0 이 아닌 CLVSS 픽처인 경우, 다음 순서의 단계들이 적용된다:
- [0132] 1. 변수 NoOutputOfPriorPicsFlag 는 다음과 같이 테스트 중인 디코더에 대해 도출된다:
- [0133] - 현재 AU 의 임의의 픽처에 대해 도출된 pic_width_max_in_luma_samples, pic_height_max_in_luma_samples, chroma_format_idc, separate_color_plane_flag, bit_depth_minus8, 또는 max_dec_pic_buffering_minus1[Htid] 의 값이 각각 동일한 CLVS 에서의 선행하는 픽처에 대한 pic_width_max_in_luma_samples, pic_height_max_in_luma_samples, chroma_format_idc, separate_color_plane_flag, bit_depth_minus8, 또는 max_dec_pic_buffering_minus1[Htid] 의 값과 상이하다면, no_output_ofPriorPics_flag 의 값에 관계없이, NoOutputOfPriorPicsFlag 는 테스트 하의 디코더에 의해 1 로 설정될 수도 있다 (그러나 그러지 않을 것이다).
- [0134] 참고 - 이러한 조건들 하에서 NoOutputOfPriorPicsFlag equal 를 no_output_of_prior_pics_flag 와 동일하게 설정하는 것이 선호되지만, 테스트 중인 디코더는 이 경우 NoOutputOfPriorPicsFlag 를 1 로 설정하도록 허용된다.
- [0135] - 그렇지 않으면, NoOutputOfPriorPicsFlag 는 no_output_of_prior_pics_flag 와 동일하게 설정된다.
- [0136] 2. 테스트 중인 디코더에 대해 도출된 NoOutputOfPriorPicsFlag 의 값은 다음과 같이 HRD 에 적용된다:
- [0137] - NoOutputOfPriorPicsFlag 가 1 이면, DPB<ADD>sub-DPB</ADD> 내의 모든 픽처 저장 버퍼가 이들이 포함하는 픽처들의 출력 없이 비워지고 DPB 충만도는 0 과 동일하게 설정된다 <ADD>비워진 각 픽처 저장 버퍼에 대해 1 씩 감분된다</ADD>.
- [0138] [서브-DPB 충만도를 사용하는 대안적인 언어에서, NoOutputOfPriorPicsFlag 가 1 과 동일하면, DPB<ADD>서브-DPB</ADD> 내의 모든 픽처 저장 버퍼들은 그들이 포함하는 픽처들의 출력 없이 비워지고, DPB<ADD>서브-DPB</ADD> 충만도는 0 으로 설정된다.]
- [0139] - 그렇지 않으면 (NoOutputOfPriorPicsFlag 는 0 과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 사용되지 않음" 으로 마킹된 픽처를 포함하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워지고 DPB<ADD>서브-DPB</ADD> 내의 모든 비어있지 않은 픽처 저장 버퍼들은 조항 C.5.2.4 에 명시된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지며, DPB 충만도는 0 과 동일하게 설정된다<ADD>비워진 각 픽처 저장 버퍼에 대해 1 씩 감분된다</ADD>.
- [0140] [서브-DPB 충만도를 사용하는 대안적인 언어에서, 그렇지 않으면 (NoOutputOfPriorPicsFlag 는 0 과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 사용되지 않음" 으로 마킹된 픽처를 포함하는 모든 픽처 저장 버퍼가 (출력 없이) 비워지고, DPB<ADD>서브-DPB</ADD>내의 모든 비어있지 않은 픽처 저장 버퍼들은 조항 C.5.2.4 에 명시된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고 DPB<ADD>서브-DPB</ADD> 충만도는 0과 동일하게 설정된다.]
- [0141] - [다른 대안적인 언어에서, 그렇지 않으면 (NoOutputOfPriorPicsFlag 는 0 과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 사용되지 않음" 으로 마킹된 픽처를 포함하는 모든 픽처 저장 버퍼가 (출력 없이) 비워지고, <ADD>모든</ADD> DPB<ADD>서브-DPB</ADD>내의 모든 비어있지 않은 픽처 저장 버퍼들은 조항 C.5.2.4 에 명시된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고 <ADD>모든 서브-DPB들의 </ADD>DPB<ADD>서브-DPB</ADD> 충만도는 0과 동일하게 설정된다.]
- [0142] 그렇지 않으면 (현재 픽처가 CLVSS 픽처가 아님 <ADD>또는 CLVSS 픽처가 픽처 0 임</ADD>), "출력을 위해 필요하지 않음" 및 "참조를 위해 사용되지 않음" 으로 마킹된 픽처를 포함하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워진다. 비워진 각각의 픽처 저장 버퍼에 대해, DPB 충만도는 1 씩 감분된다. 다음 조건들 중

하나 이상이 참 (true) 인 경우, 다음 조건 중 어느 것도 참이 아닐 때까지, 비워진 각각의 추가 픽처 저장 버퍼에 대해 DPB 충만도를 1 씩 추가로 감분하면서 조항 C.5.2.4 에 명시된 "범핑" 프로세스가 반복적으로 호출된다:

- [0143] - "출력을 위해 필요함" 로 마킹된 DPB 내의 픽처들의 수는 $\text{max_num_reorder_pics}[\text{Htid}]$ 보다 크다.
- [0144] - $\text{max_latency_increase_plus1}[\text{Htid}]$ 는 0 과 동일하지 않고, 연관된 변수 PicLatencyCount 가 $\text{MaxLatencyPictures}[\text{Htid}]$ 보다 크거나 동일한, "출력을 위해 필요함" 으로 마킹된 적어도 하나의 픽처가 DPB 내에 존재한다.
- [0145] - DPB 내의 픽처들의 수는 $\text{max_dec_pic_buffering_minus1}[\text{Htid}] + 1$ 보다 크거나 동일하다.

[0146] **C.5.2.3 추가 범핑**

[0147] 이 조항에 명시된 프로세스는 현재 픽처를 포함하는 AU n 의 최종 DU 가 CPB 로부터 제거될 때 순간적으로 발생한다.

[0148] 현재 픽처가 1 과 동일한 PictureOutputFlag 를 가질 때, "출력을 위해 필요함" 으로 마킹되고 출력 순서에서 현재 픽처를 뒤따르는 DPB 내의 각각의 픽처에 대해, 연관된 변수 PicLatencyCount 는 $\text{PicLatencyCount} + 1$ 과 동일하게 설정된다.

[0149] 다음이 적용된다:

[0150] - 현재 디코딩된 픽처가 1 과 동일한 PictureOutputFlag 를 가지면, "출력을 위해 필요함" 으로 마킹되고 연관된 변수 PicLatencyCount 가 0 으로 설정된다.

[0151] - 그렇지 않으면 (현재 디코딩된 픽처가 0 과 동일한 PictureOutputFlag 를 가짐), "출력을 위해 필요하지 않음"으로 마킹된다.

[0152] 다음 조건들 중 하나 이상이 참이면, 다음 조건들 중 어느것도 참이 아닐때까지 조항 C.5.2.4 에 명시된 "범핑" 프로세스가 반복적으로 호출된다.

[0153] - "출력을 위해 필요함" 로 마킹된 DPB 내의 픽처들의 수는 $\text{max_num_reorder_pics}[\text{Htid}]$ 보다 크다.

[0154] - $\text{max_latency_increase_plus1}[\text{Htid}]$ 는 0 과 동일하지 않고, 연관된 변수 PicLatencyCount 가 $\text{MaxLatencyPictures}[\text{Htid}]$ 보다 크거나 동일한, "출력을 위해 필요함" 으로 마킹된 적어도 하나의 픽처가 DPB 내에 존재한다.

[0155] **C.5.2.4 "범핑" 프로세스**

[0156] "범핑" 프로세스는 다음의 순서화된 단계들을 포함한다:

[0157] 1. 먼저 출력되기 위한 픽처 또는 픽처들이 "출력을 위해 필요함" 으로 마킹된 DPB 내의 모든 픽처들의 PicOrderCntVal 의 최소 값을 갖는 픽처로서 선택된다.

[0158] 2. 이러한 픽처들 각각은, 오름차순 nuh_layer_id 순서로, 픽처에 대한 적합성 크롭핑 윈도우를 사용하여 크롭되고, 크롭된 픽처가 출력되고, 픽처는 "출력을 위해 필요하지 않음" 으로 마킹된다.

[0159] 3. "참조를 위해 사용되지 않음" 으로 마킹된 픽처를 포함하고, 크롭된 픽처들 중 하나 및 출력적이었던 각 픽처 저장 버퍼는 비워지고, 연결된 서브-DPB 의 충만도는 1 씩 감분된다.

[0160] 참고 - 동일한 CVS 에 속하고 "범핑 프로세스" 에 의해 출력되는 임의의 2 개의 픽처들 picA 및 picB 에 대해, picA 가 picB 보다 먼저 출력되는 경우, picA 의 PicOrderCntVal 의 값은 picB 의 PicOrderCntVal 의 값보다 작다<ADD>또는 동일하다</ADD>.

[0161] 도 3 은 본 개시물의 기법들을 수행할 수도 있는 예시적인 비디오 인코더 (200) 를 예시하는 블록 다이어그램이다. 도 3 은 설명의 목적들로 제공되며, 본 개시에서 대략적으로 예시화되고 설명된 바와 같은 기술들의 한 정으로서 고려되지 않아야 한다. 설명의 목적으로, 본 개시는 HEVC 비디오 코딩 표준 및 개발 중인 H.266 비디오 코딩 표준과 같은 비디오 코딩 표준들의 콘텍스트에서 비디오 인코더 (200) 를 설명한다. 그러나, 본 개시의 기법들은 이들 비디오 코딩 표준들에 제한되지 않으며, 일반적으로 비디오 인코딩 및 디코딩에 적용 가능하다.

- [0162] 도 3의 예에서, 비디오 인코더 (200)는 비디오 데이터 메모리 (230), 모드 선택 유닛 (202), 잔차 생성 유닛 (204), 변환 프로세싱 유닛 (206), 양자화 유닛 (208), 역양자화 유닛 (210), 역변환 프로세싱 유닛 (212), 재구성 유닛 (214), 필터 유닛 (216), DPB (218), 및 엔트로피 인코딩 유닛 (220)을 포함한다. 비디오 데이터 메모리 (230), 모드 선택 유닛 (202), 잔차 생성 유닛 (204), 변환 프로세싱 유닛 (206), 양자화 유닛 (208), 역양자화 유닛 (210), 역변환 프로세싱 유닛 (212), 복원 유닛 (214), 필터 유닛 (216), DPB (218), 및 엔트로피 인코딩 유닛 (220) 중 임의의 것 또는 전부는 하나 이상의 프로세서들에서 또는 프로세싱 회로부에서 구현될 수도 있다. 예를 들어, 비디오 인코더 (200)의 유닛들은 하드웨어 회로부의 부분으로서, 또는 프로세서, ASIC, 또는 FPGA의 부분으로서 하나 이상의 회로들 또는 로직 엘리먼트들로서 구현될 수도 있다. 더욱이, 비디오 인코더 (200)는 이들 및 다른 기능들을 수행하기 위해 추가 또는 대안의 프로세서들 또는 프로세싱 회로부를 포함할 수도 있다.
- [0163] 비디오 데이터 메모리 (230)는 비디오 인코더 (200)의 컴포넌트들에 의해 인코딩될 비디오 데이터를 저장할 수도 있다. 비디오 인코더 (200)는 예를 들어, 비디오 소스 (104) (도 1)로부터 비디오 데이터 메모리 (230)에 저장된 비디오 데이터를 수신할 수도 있다. DPB (218)는 비디오 인코더 (200)에 의한 후속 비디오 데이터의 예측에 사용하기 위해 레퍼런스 비디오 데이터를 저장하는 레퍼런스 픽처 메모리로서 작용할 수도 있다. 비디오 데이터 메모리 (230) 및 DPB (218)는 동기식 동적 랜덤 액세스 메모리 (SDRAM)를 포함한 동적 랜덤 액세스 메모리 (DRAM), 자기저항성 RAM (MRAM), 저항성 RAM (RRAM), 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들 중 임의의 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (230) 및 DPB (218)는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에 있어서, 비디오 데이터 메모리 (230)는, 예시된 바와 같은 비디오 인코더 (200)의 다른 컴포넌트들과 온-칩형이거나 또는 그들 컴포넌트들에 대하여 오프-칩형일 수도 있다.
- [0164] DPB (218)는 공유되는 다중-계층 DPB (219)를 포함할 수도 있다. 본 개시물의 다른 곳에서 더 상세히 설명되는 바와 같이, 비디오 인코더 (200)는 복수의 계층들에 대한 레퍼런스 픽처들을 공유된 다중-계층 DPB (219)에 저장하도록 구성될 수도 있다. 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 비디오 인코더 (200)는 공유된 다중-계층 DPB (219)에 대해 픽처 출력 및 제거 프로세스를 수행하여 제 1 계층에 속하는 일부 디코딩된 픽처들만을 공유된 다중-계층 DPB (219)로부터 제거할 수도 있다. CPB로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 비디오 인코더 (200)는 공유된 다중-계층 DPB (219)의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행할 수도 있다.
- [0165] 본 개시에서, 비디오 데이터 메모리 (230)에 대한 참조는, 이와 같이 구체적으로 설명되지 않으면 비디오 인코더 (200) 내부의 메모리, 또는 이와 같이 구체적으로 설명되지 않으면 비디오 인코더 (200) 외부의 메모리로 한정되는 것으로 해석되지 않아야 한다. 오히려, 비디오 데이터 메모리 (230)에 대한 참조는, 비디오 인코더 (200)가 인코딩을 위해 수신하는 비디오 데이터 (예컨대, 인코딩될 현재 블록에 대한 비디오 데이터)를 저장하는 레퍼런스 메모리로서 이해되어야 한다. 도 1의 메모리 (106)는 또한 비디오 인코더 (200)의 다양한 유닛들로부터의 출력들의 일시적 저장을 제공할 수도 있다.
- [0166] 도 3의 다양한 유닛들은 비디오 인코더 (200)에 의해 수행되는 동작들의 이해를 돕기 위해 예시된다. 그 유닛들은 고정 기능 회로들, 프로그래밍가능 회로들, 또는 이들의 조합으로서 구현될 수도 있다. 고정 기능 회로들은 특정 기능성을 제공하는 회로들을 지칭하고, 수행될 수 있는 동작들에 대해 미리설정된다. 프로그래밍가능 회로들은 다양한 태스크들을 수행하도록 프로그래밍될 수 있는 회로들을 지칭하고, 수행될 수 있는 동작들에서 유연한 기능성을 제공한다. 예를 들어, 프로그래밍가능 회로들은, 프로그래밍가능 회로들로 하여금 소프트웨어 또는 펌웨어의 명령들에 의해 정의된 방식으로 동작하게 하는 소프트웨어 또는 펌웨어를 실행할 수도 있다. 고정 기능 회로들은 (예컨대, 파라미터들을 수신하거나 또는 파라미터들을 출력하기 위해) 소프트웨어 명령들을 실행할 수도 있지만, 고정 기능 회로들이 수행하는 동작들의 타입들은 일반적으로 불변이다. 일부 예들에 있어서, 유닛들 중 하나 이상은 별개의 회로 블록들 (고정 기능 또는 프로그래밍가능)일 수도 있고, 일부 예들에 있어서, 유닛들 중 하나 이상은 집적 회로들일 수도 있다.
- [0167] 비디오 인코더 (200)는 프로그래밍가능 회로들로부터 형성된, 산술 로직 유닛들 (ALU들), 기본 함수 유닛들 (EFU들), 디지털 회로들, 아날로그 회로들, 및/또는 프로그래밍가능 코어들을 포함할 수도 있다. 비디오 인코더 (200)의 동작들이 프로그래밍가능 회로들에 의해 실행된 소프트웨어를 사용하여 수행되는 예들에 있어서, 메모리 (106) (도 1)는 비디오 인코더 (200)가 수신하고 실행하는 소프트웨어의 명령들 (예컨대, 오브젝트 코드)을 저장할 수도 있거나, 또는 비디오 인코더 (200)내의 다른 메모리 (도시 안됨)가 그러한 명령들을 저장

할 수도 있다.

- [0168] 비디오 데이터 메모리 (230) 는 수신된 비디오 데이터를 저장하도록 구성된다. 비디오 인코더 (200) 는 비디오 데이터 메모리 (230) 로부터 비디오 데이터의 픽처를 추출하고, 비디오 데이터를 잔차 생성 유닛 (204) 및 모드 선택 유닛 (202) 에 제공할 수도 있다. 비디오 데이터 메모리 (230) 에서의 비디오 데이터는 인코딩될 원시 비디오 데이터일 수도 있다.
- [0169] 모드 선택 유닛 (202) 은 모션 추정 유닛 (222), 모션 보상 유닛 (224), 및 인트라-예측 유닛 (226) 을 포함한다. 모드 선택 유닛 (202) 은 다른 예측 모드들에 따라 비디오 예측을 수행하기 위해 추가적인 기능 유닛들을 포함할 수도 있다. 예를로서, 모드 선택 유닛 (202) 은 팔레트 유닛, 인트라-블록 카피 유닛 (모션 추정 유닛 (222) 및/또는 모션 보상 유닛 (224) 의 부분일 수도 있음), 아핀 유닛, 선형 모델 (LM) 유닛 등을 포함할 수도 있다.
- [0170] 모드 선택 유닛 (202) 은 일반적으로 인코딩 파라미터들의 조합들 및 그러한 조합들에 대한 결과적인 레이트-왜곡 값들을 테스트하기 위해 다중 인코딩 패스들을 조정한다. 인코딩 파라미터들은 CTU들의 CU들로의 파티셔닝, CU들에 대한 예측 모드들, CU들의 잔차 데이터에 대한 변환 타입들, CU들의 잔차 데이터에 대한 양자화 파라미터들 등을 포함할 수도 있다. 모드 선택 유닛 (202) 은 궁극적으로 다른 테스트된 조합들보다 더 양호한 레이트-왜곡 값들을 갖는 인코딩 파라미터들의 조합을 선택할 수도 있다.
- [0171] 비디오 인코더 (200) 는 비디오 데이터 메모리 (230) 로부터 추출된 화상을 일련의 CTU들로 파티셔닝하고, 슬라이스 내에 하나 이상의 CTU들을 캡슐화할 수도 있다. 모드 선택 유닛 (202) 은 위에 설명된 HEVC 의 쿼드트리 구조 또는 QTBT 구조와 같은, 트리 구조에 따라 화상의 CTU 를 파티셔닝할 수도 있다. 위에 설명된 바와 같이, 비디오 인코더 (200) 는 트리 구조에 따라 CTU 를 파티셔닝하는 것으로부터 하나 이상의 CU들을 형성할 수도 있다. 그러한 CU 는 일반적으로 "비디오 블록" 또는 "블록" 으로도 또한 지칭될 수도 있다.
- [0172] 일반적으로, 모드 선택 유닛 (202) 은 또한 그의 컴포넌트들 (예컨대, 모션 추정 유닛 (222), 모션 보상 유닛 (224), 및 인트라-예측 유닛 (226)) 을 제어하여 현재 블록 (예컨대, 현재 CU, 또는 HEVC 에서, PU 및 TU 의 중첩 부분) 에 대한 예측 블록을 생성한다. 현재 블록의 인터 예측을 위해, 모션 추정 유닛 (222) 은 하나 이상의 레퍼런스 픽처들 (DPB (218) 에 저장된 하나 이상의 이전에 코딩된 픽처들) 에서 하나 이상의 밀접하게 매칭하는 레퍼런스 블록들을 식별하기 위해 모션 탐색을 수행할 수도 있다. 특히, 모션 추정 유닛 (222) 은, 예를 들어, 절대 차이의 합 (SAD), 제곱 차이들의 합 (SSD), 평균 절대 차이 (MAD), 평균 제곱 차이들 (MSD) 등에 따라, 잠재적 레퍼런스 블록이 현재 블록에 얼마나 유사한지를 나타내는 값을 계산할 수도 있다. 모션 추정 유닛 (222) 은 일반적으로 고려되는 레퍼런스 블록과 현재 블록 사이의 샘플 별 차이들을 사용하여 이들 계산들을 수행할 수도 있다. 모션 추정 유닛 (222) 은 현재 블록에 가장 근접하게 매칭하는 레퍼런스 블록을 표시하는, 이러한 계산들로부터 야기되는 최저 값을 갖는 레퍼런스 블록을 식별할 수도 있다.
- [0173] 모션 추정 유닛 (222) 은, 현재 픽처에서의 현재 블록의 위치에 대한 레퍼런스 픽처들에서의 레퍼런스 블록들의 위치들을 정의하는 하나 이상의 모션 벡터들 (MV들) 을 형성할 수도 있다. 그 다음, 모션 추정 유닛 (222) 은 모션 벡터들을 모션 보상 유닛 (224) 에 제공할 수도 있다. 예를 들어, 단방향 인터-예측에 대해, 모션 추정 유닛 (222) 은 단일 모션 벡터를 제공할 수도 있는 반면, 양방향 인터-예측에 대해, 모션 추정 유닛 (222) 은 2개의 모션 벡터들을 제공할 수도 있다. 그 다음, 모션 보상 유닛 (224) 은 모션 벡터들을 사용하여 예측 블록을 생성할 수도 있다. 예를 들어, 모션 보상 유닛 (224) 은 모션 벡터를 사용하여 레퍼런스 블록의 데이터를 추출할 수도 있다. 다른 예로서, 모션 벡터가 분수 샘플 정밀도를 갖는다면, 모션 보상 유닛 (224) 은 하나 이상의 보간 필터들에 따라 예측 블록에 대한 값들을 보간할 수도 있다. 더욱이, 양방향 인터-예측에 대해, 모션 보상 유닛 (224) 은 개별의 모션 벡터들에 의해 식별된 2 개의 레퍼런스 블록들에 대한 데이터를 추출하고, 예컨대, 샘플 별 평균화 또는 가중 평균화를 통해 추출된 데이터를 결합할 수도 있다.
- [0174] 다른 예로서, 인트라-예측 또는 인트라-예측 코딩에 대해, 인트라-예측 유닛 (226) 은 현재 블록에 이웃하는 샘플들로부터 예측 블록을 생성할 수도 있다. 예를 들어, 방향성 모드들에 대해, 인트라-예측 유닛 (226) 은 일반적으로 이웃하는 샘플들의 값들을 수학적으로 결합하고 현재 블록에 걸쳐 정의된 방향에서 이들 계산된 값들을 과플레이팅하여 예측 블록을 생성할 수도 있다. 다른 예로서, DC 모드에 대해, 인트라-예측 유닛 (226) 은 현재 블록에 대한 이웃하는 샘플들의 평균을 계산하고 예측 블록을 생성하여 예측 블록의 각각의 샘플에 대해 이러한 결과적인 평균을 포함할 수도 있다.

- [0175] 모드 선택 유닛 (202) 은 예측 블록을 잔차 생성 유닛 (204) 에 제공한다. 잔차 생성 유닛 (204) 은 비디오 데이터 메모리 (230) 로부터의 현재 블록의 원시, 인코딩되지 않은 버전 및 모드 선택 유닛 (202) 으로부터의 예측 블록을 수신한다. 잔차 생성 유닛 (204) 은 현재 블록과 예측 블록 사이의 샘플 별 차이들을 계산한다. 결과의 샘플 별 차이들은 현재 블록에 대한 잔차 블록을 정의한다. 일부 예들에서, 잔차 생성 유닛 (204) 은 또한 잔차 차동 펄스 코드 변조 (RDPCM) 를 사용하여 잔차 블록을 생성하기 위해 잔차 블록에서의 샘플 값들 사이의 차이들을 결정할 수도 있다. 일부 예들에서, 잔차 생성 유닛 (204) 은 이진 감산을 수행하는 하나 이상의 감산 회로들을 사용하여 형성될 수도 있다.
- [0176] 모드 선택 유닛 (202) 이 CU들을 PU들로 파티셔닝하는 예들에서, 각각의 PU 는 루마 예측 유닛 및 대응하는 크로마 예측 유닛들과 연관될 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 다양한 사이즈를 갖는 PU들을 지원할 수도 있다. 상기 나타낸 바와 같이, CU 의 사이즈는 CU 의 루마 코딩 블록의 사이즈를 지칭할 수도 있고 PU 의 사이즈는 PU 의 루마 예측 유닛의 사이즈를 지칭할 수도 있다. 특정 CU 의 사이즈가 $2N_x \times 2N_y$ 임을 가정하면, 비디오 인코더 (200) 는 인트라 예측을 위해 $2N_x \times 2N_y$ 또는 $N_x \times N_y$ 의 PU 사이즈들을 지원하고, 인터 예측을 위해 $2N_x \times 2N_y$, $2N_x \times N_y$, $N_x \times 2N_y$, $N_x \times N_y$ 등의 대칭적인 PU 사이즈들을 지원할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 또한, 인터 예측을 위해 $2N_x \times N_y$, $2N_x \times nD$, $nL_x \times 2N_y$, 및 $nR_x \times 2N_y$ 의 PU 크기들에 대한 비대칭적인 파티셔닝을 지원할 수도 있다.
- [0177] 모드 선택 유닛 (202) 이 CU 를 PU들로 추가로 파티셔닝하지 않는 예들에 있어서, 각각의 CU 는 루마 코딩 블록 및 대응하는 크로마 코딩 블록들과 연관될 수도 있다. 상기에서와 같이, CU 의 사이즈는 CU 의 루마 코딩 블록의 사이즈를 지칭할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 $2N_x \times 2N_y$, $2N_x \times N_y$, 또는 $N_x \times 2N_y$ 의 CU 사이즈들을 지원할 수도 있다.
- [0178] 몇몇 예들로서, 인트라-블록 카피 모드 코딩, 아핀 모드 코딩 및 선형 모델 (LM) 모드 코딩과 같은 다른 비디오 코딩 기법들에 대해, 모드 선택 유닛 (202) 은, 코딩 기법들과 연관된 개별의 유닛들을 통해, 인코딩되는 현재 블록에 대한 예측 블록을 생성한다. 팔레트 모드 코딩과 같은 일부 예들에서, 모드 선택 유닛 (202) 은 예측 블록을 생성하지 않을 수도 있고, 대신 선택된 팔레트에 기초하여 블록을 재구성하는 방식을 표시하는 구문 요소들을 생성할 수도 있다. 그러한 모드들에서, 모드 선택 유닛 (202) 은 이들 선택스 엘리먼트들을 인코딩될 엔트로피 인코딩 유닛 (220) 에 제공할 수도 있다.
- [0179] 상기 설명된 바와 같이, 잔차 생성 유닛 (204) 은 현재 블록 및 대응하는 예측 블록에 대한 비디오 데이터를 수신한다. 잔차 생성 유닛 (204) 은 그 후 현재 블록을 위한 잔차 블록을 생성한다. 잔차 블록을 생성하기 위해, 잔차 생성 유닛 (204) 은 예측 블록과 현재 블록 사이의 샘플 별 차이들을 계산한다.
- [0180] 변환 프로세싱 유닛 (206) 은 잔차 블록에 하나 이상의 변환들을 적용하여 변환 계수들의 블록 (본원에서는 "변환 계수 블록" 으로 지칭됨) 을 생성한다. 변환 프로세싱 유닛 (206) 은 다양한 변환들을 잔차 블록에 적용하여 변환 계수 블록을 형성할 수도 있다. 예를 들어, 변환 프로세싱 유닛 (206) 은 이산 코사인 변환 (DCT), 방향성 변환, Karhunen-Loeve 변환 (KLT), 또는 개념적으로 유사한 변환을 잔차 블록에 적용할 수도 있다. 일부 예들에 있어서, 변환 프로세싱 유닛 (206) 은 잔차 블록에 대한 다중의 변환들, 예컨대, 1 차 변환 및 2 차 변환, 예컨대 회전 변환을 수행할 수도 있다. 일부 예들에 있어서, 변환 프로세싱 유닛 (206) 은 잔차 블록에 변환들을 적용하지 않는다.
- [0181] 양자화 유닛 (208) 은 양자화된 변환 계수 블록을 생성하기 위해 변환 계수 블록에서의 변환 계수들을 양자화할 수도 있다. 양자화 유닛 (208) 은 현재 블록과 연관된 양자화 파라미터 (QP) 값에 따라 변환 계수 블록의 변환 계수들을 양자화할 수도 있다. 비디오 인코더 (200) 는 (예컨대, 모드 선택 유닛 (202) 을 통해) CU 와 연관된 QP 값을 조정함으로써 현재 블록과 연관된 변환 계수 블록들에 적용되는 양자화도를 조정할 수도 있다. 양자화는 정보의 손실을 도입할 수도 있으며, 따라서, 양자화된 변환 계수들은 변환 프로세싱 유닛 (206) 에 의해 생성된 원래 변환 계수들보다 더 낮은 정밀도를 가질 수도 있다.
- [0182] 역양자화 유닛 (210) 및 역변환 프로세싱 유닛 (212) 은 각각 양자화된 변환 계수 블록에 역양자화 및 역변환들을 적용하여, 변환 계수 블록으로부터 잔차 블록을 복원할 수도 있다. 복원 유닛 (214) 은 모드 선택 유닛 (202) 에 의해 생성된 예측 블록 및 복원된 잔차 블록에 기초하여 (잠재적으로 어느 정도의 왜곡을 가짐에도 불구하고) 현재 블록에 대응하는 복원된 블록을 생성할 수도 있다. 예를 들어, 복원 유닛 (214) 은 복원된 잔차 블록의 샘플들을, 모드 선택 유닛 (202) 에 의해 생성된 예측 블록으로부터의 대응하는 샘플들에 가산하여 복원된 블록을 생성할 수도 있다.

- [0183] 필터 유닛 (216) 은 복원된 블록들에 대해 하나 이상의 필터 동작을 수행할 수도 있다. 예를 들어, 필터 유닛 (216) 은 CU들의 예지들을 따라 블록키니스 아티팩트들 (blockiness artifacts) 을 감소시키기 위해 디블록킹 동작들을 수행할 수도 있다. 필터 유닛 (216) 의 동작들은 일부 예들에서 스킵될 수도 있다.
- [0184] 비디오 인코더 (200) 는 DPB (218) 에 재구성된 블록들을 저장한다. 예를 들어, 필터 유닛 (216) 의 동작들이 수행되지 않은 예들에서, 재구성 유닛 (214) 은 재구성된 블록들을 DPB (218) 에 저장할 수도 있다. 필터 유닛 (216) 의 동작들이 수행되는 예들에서, 필터 유닛 (216) 은 필터링된 복원된 블록들을 DPB (218) 에 저장할 수도 있다. 모션 추정 유닛 (222) 및 모션 보상 유닛 (224) 은 복원된 (및 잠재적으로 필터링된) 블록들로부터 형성된 DPB (218) 로부터 레퍼런스 픽처를 추출하여, 후속하여 인코딩된 픽처들의 블록들을 인터-예측할 수도 있다. 또한, 인트라-예측 유닛 (226) 은 현재 픽처에서의 다른 블록들을 인트라-예측하기 위해 현재 픽처의 DPB (218) 에서의 복원된 블록들을 사용할 수도 있다.
- [0185] 일반적으로, 엔트로피 인코딩 유닛 (220) 은 비디오 인코더 (200) 의 다른 기능 컴포넌트들로부터 수신된 선택스 엘리먼트들을 엔트로피 인코딩할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (220) 은 양자화 유닛 (208) 으로부터의 양자화된 변환 계수 블록들을 엔트로피 인코딩할 수도 있다. 다른 예로서, 엔트로피 인코딩 유닛 (220) 은 모드 선택 유닛 (202) 으로부터 예측 선택스 엘리먼트들 (예를 들어, 인트라-예측에 대한 인트라-모드 정보 또는 인터-예측에 대한 모션 정보) 을 엔트로피 인코딩할 수도 있다. 엔트로피 인코딩 유닛 (220) 은 엔트로피-인코딩된 데이터를 생성하기 위해, 비디오 데이터의 다른 예인, 선택스 엘리먼트들에 대해 하나 이상의 엔트로피 인코딩 동작들을 수행할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (220) 은 컨텍스트 적응적 가변 길이 코딩 (CAVLC) 동작, CABAC 동작, V2V (variable-to-variable) 길이 코딩 동작, 선택스 기반 컨텍스트 적응적 이진 산술 코딩 (SBAC) 동작, 확률 인터벌 파티셔닝 엔트로피 (PIPE) 코딩 동작, 지수-골롬 인코딩 동작, 또는 다른 타입의 엔트로피 인코딩 동작을 데이터에 대해 수행할 수도 있다. 일부 예들에서, 엔트로피 인코딩 유닛 (220) 은 선택스 엘리먼트들이 엔트로피 인코딩되지 않는 바이패스 모드에서 동작할 수도 있다.
- [0186] 비디오 인코더 (200) 는 픽처 또는 슬라이스의 블록들을 복원하는데 필요한 엔트로피 인코딩된 선택스 엘리먼트들을 포함하는 비트스트림을 출력할 수도 있다. 특히, 엔트로피 인코딩 유닛 (220) 은 비트스트림을 출력할 수도 있다.
- [0187] 상기 설명된 동작들은 블록에 관하여 설명된다. 그러한 설명은 루마 코딩 블록 및/또는 크로마 코딩 블록들에 대한 동작들인 것으로서 이해되어야 한다. 상기 설명된 바와 같이, 일부 예들에서, 루마 코딩 블록 및 크로마 코딩 블록들은 CU 의 루마 및 크로마 컴포넌트들이다. 일부 예들에 있어서, 루마 코딩 블록 및 크로마 코딩 블록들은 PU 의 루마 및 크로마 컴포넌트들이다.
- [0188] 일부 예들에 있어서, 루마 코딩 블록에 대해 수행된 동작들은 크로마 코딩 블록들에 대해 반복될 필요가 없다. 일 예로서, 루마 코딩 블록에 대한 모션 벡터 (MV) 및 레퍼런스 픽처를 식별하기 위한 동작들이, 크로마 블록들에 대한 MV 및 레퍼런스 픽처를 식별하기 위해 반복될 필요는 없다. 오히려, 루마 코딩 블록에 대한 MV 는 크로마 블록들에 대한 MV 를 결정하도록 스케일링될 수도 있으며, 레퍼런스 픽처는 동일할 수도 있다. 다른 예로서, 인트라-예측 프로세스는 루마 코딩 블록 및 크로마 코딩 블록들에 대해 동일할 수도 있다.
- [0189] 비디오 인코더 (200) 는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로에서 구현되고 이하 청구범위 섹션에 설명된 기법들을 포함하는 본 개시물의 기법들을 수행하도록 구성된 하나 이상의 프로세싱 유닛들을 포함하는, 비디오 데이터를 인코딩하도록 구성된 디바이스의 예를 나타낸다.
- [0190] 도 4 는 본 개시물의 기법들을 수행할 수도 있는 예시적인 비디오 디코더 (300) 를 예시하는 블록 다이어그램이다. 도 4 는 설명의 목적들로 제공되며, 본 개시에 대략적으로 예시화되고 설명된 바와 같은 기술들에 대한 한정이 아니다. 설명의 목적으로, 본 개시는 JEM, VVC 및 HEVC 의 기법들에 따른 비디오 디코더 (300) 를 기재한다. 그러나, 본 개시의 기법들은 다른 비디오 코딩 표준들로 구성되는 비디오 코딩 디바이스들에 의해 수행될 수도 있다.
- [0191] 도 4 의 예에서, 비디오 디코더 (300) 는 CPB 메모리 (320), 엔트로피 디코딩 유닛 (302), 예측 프로세싱 유닛 (304), 역양자화 유닛 (306), 역변환 프로세싱 유닛 (308), 재구성 유닛 (310), 필터 유닛 (312), 및 DPB (314) 를 포함한다. CPB 메모리 (320), 엔트로피 디코딩 유닛 (302), 예측 프로세싱 유닛 (304), 역양자화 유닛 (306), 역변환 프로세싱 유닛 (308), 복원 유닛 (310), 필터 유닛 (312), 및 DPB (314) 중 임의의 것 또는 전부는 하나 이상의 프로세서들에서 또는 프로세싱 회로부에서 구현될 수도 있다. 예를 들어, 비디오 디

코더 (300) 의 유닛들은 하드웨어 회로부의 일부로서 또는 FPGA 의 프로세서, ASIC 의 일부로서 하나 이상의 회로들 또는 로직 엘리먼트들로서 구현될 수도 있다. 더욱이, 비디오 디코더 (300) 는 이들 및 다른 기능들을 수행하기 위해 부가 또는 대안의 프로세서들 또는 프로세싱 회로부를 포함할 수도 있다.

[0192] 예측 프로세싱 유닛 (304) 은 모션 보상 유닛 (316) 및 인트라-예측 유닛 (318) 을 포함한다. 예측 프로세싱 유닛 (304) 은 다른 예측 모드들에 따라 예측을 수행하기 위해 추가적인 유닛들을 포함할 수도 있다. 예를 들어, 예측 프로세싱 유닛 (304) 은 팔레트 유닛, 인트라-블록 카피 유닛 (모션 보상 유닛 (316) 의 부분을 형성할 수도 있음), 아핀 유닛, 선형 모델 (LM) 유닛 등을 포함할 수도 있다. 다른 예들에서, 비디오 디코더 (300) 는 더 많거나, 더 적거나, 또는 상이한 기능 컴포넌트들을 포함할 수도 있다.

[0193] CPB 메모리 (320) 는 비디오 디코더 (300) 의 컴포넌트들에 의해 디코딩될 인코딩된 비디오 비트스트림과 같은 비디오 데이터를 저장할 수도 있다. CPB 메모리 (320) 에 저장된 비디오 데이터는, 예를 들어 컴퓨터 판독 가능 매체 (110) (도 1) 로부터 획득될 수도 있다. CPB 메모리 (320) 는 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터 (예를 들어, 신텍스 엘리먼트들) 를 저장하는 CPB 를 포함할 수도 있다. 또한, CPB 메모리 (320) 는 비디오 디코더 (300) 의 다양한 유닛들로부터의 출력들을 나타내는 일시적 데이터와 같은, 코딩된 픽처의 신텍스 엘리먼트들 이외의 비디오 데이터를 저장할 수도 있다. DPB (314) 는 일반적으로, 인코딩된 비디오 비트스트림의 후속 데이터 또는 픽처들을 디코딩할 때, 레퍼런스 비디오 데이터로서 비디오 디코더 (300) 가 출력 및/또는 사용할 수도 있는 디코딩된 픽처들을 저장한다. CPB 메모리 (320) 및 DPB (314) 는 SDRAM 을 포함한 DRAM, MRAM, RRAM, 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들 중 임의의 것에 의해 형성될 수도 있다. CPB 메모리 (320) 및 DPB (314) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, CPB 메모리 (320) 는 비디오 디코더 (300) 의 다른 컴포넌트들과 온-칩이거나 그 컴포넌트들에 대하여 오프-칩일 수도 있다.

[0194] 추가적으로 또는 대안적으로, 일부 예들에서, 비디오 디코더 (300) 는 메모리 (120) (도 1) 로부터 코딩된 비디오 데이터를 취출할 수도 있다. 즉, 메모리 (120) 는 CPB 메모리 (320) 로 상기 논의된 바와 같이 데이터를 저장할 수도 있다. 마찬가지로, 메모리 (120) 는, 비디오 디코더 (300) 의 기능성의 일부 또는 전부가 비디오 디코더 (300) 의 프로세싱 회로부에 의해 실행되도록 소프트웨어에서 구현될 때, 비디오 디코더 (300) 에 의해 실행될 명령들을 저장할 수도 있다.

[0195] 도 4 에 도시된 다양한 유닛들은 비디오 디코더 (300) 에 의해 수행되는 동작들의 이해를 돕기 위해 도시된다. 상기 유닛들은 고정 기능 회로, 프로그램가능 회로, 또는 이들의 조합으로 구현될 수도 있다. 도 3 과 유사하게, 고정 기능 회로들은 특정 기능성을 제공하는 회로들을 지칭하고, 수행될 수 있는 동작들에 대해 사전 설정된다. 프로그램 가능한 회로들은 다양한 작업을 수행하도록 프로그래밍될 수 있는 회로들을 지칭하고, 수행될 수 있는 동작들에 유연한 기능을 제공한다. 예를 들어, 프로그래밍가능 회로들은, 프로그래밍가능 회로들이 소프트웨어 또는 펌웨어의 명령들에 의해 정의된 방식으로 동작하게 하는 소프트웨어 또는 펌웨어를 실행할 수도 있다. 고정 기능 회로들은 (예를 들어, 파라미터들을 수신하거나 또는 파라미터들을 출력하기 위해) 소프트웨어 명령들을 실행할 수도 있지만, 고정 기능 회로들이 수행하는 동작들의 타입들은 일반적으로 불변이다. 일부 예들에 있어서, 유닛들 중 하나 이상은 별개의 회로 블록들 (고정 기능 또는 프로그래밍가능) 일 수도 있고, 일부 예들에 있어서, 유닛들 중 하나 이상은 집적 회로들일 수도 있다.

[0196] 비디오 디코더 (300) 는 프로그래밍가능 회로들로부터 형성된, ALU들, EFU들, 디지털 회로들, 아날로그 회로들, 및/또는 프로그래밍가능 코어들을 포함할 수도 있다. 비디오 디코더 (300) 의 동작들이 프로그래밍가능 회로들 상에서 실행하는 소프트웨어에 의해 수행되는 예들에 있어서, 온-칩 또는 오프-칩 메모리는, 비디오 디코더 (300) 가 수신 및 실행하는 소프트웨어의 명령들 (예컨대, 오브젝트 코드) 을 저장할 수도 있다.

[0197] 엔트로피 디코딩 유닛 (302) 은 인코딩된 비디오 데이터를 CPB 로부터 수신하고, 비디오 데이터를 엔트로피 디코딩하여 신텍스 엘리먼트들을 재생할 수도 있다. 예측 프로세싱 유닛 (304), 역양자화 유닛 (306), 역변환 프로세싱 유닛 (308), 복원 유닛 (310), 및 필터 유닛 (312) 은 비트스트림으로부터 추출된 신텍스 엘리먼트들에 기초하여 디코딩된 비디오 데이터를 생성할 수도 있다.

[0198] 일반적으로, 비디오 디코더 (300) 는 블록 별 기반으로 픽처를 복원한다. 비디오 디코더 (300) 는 개별적으로 각각의 블록에 대해 복원 동작을 수행할 수도 있다 (여기서 현재 복원되는, 즉 디코딩되는 블록은 "현재 블록" 으로서 지칭될 수도 있음).

[0199] 엔트로피 디코딩 유닛 (302) 은, 양자화 파라미터 (QP) 및/또는 변환 모드 표시(들)와 같은 변환 정보 뿐만 아

나라, 양자화된 변환 계수 블록의 양자화된 변환 계수들을 정의하는 신택스 엘리먼트들을 엔트로피 디코딩할 수도 있다. 역양자화 유닛 (306)은 양자화된 변환 계수 블록과 연관된 QP 를 사용하여, 양자화도 그리고 마찬가지로, 역양자화 유닛 (306)이 적용할 역양자화도를 결정할 수도 있다. 역양자화 유닛 (306)은, 예를 들어, 양자화된 변환 계수들을 역양자화하기 위해 비트단위 좌측-시프트 동작을 수행할 수도 있다. 역양자화 유닛 (306)은 이에 의해 변환 계수들을 포함하는 변환 계수 블록을 형성할 수도 있다.

[0200] 역양자화 유닛 (306)이 변환 계수 블록을 형성한 후, 역변환 프로세싱 유닛 (308)은 현재 블록과 연관된 잔차 블록을 생성하기 위해 변환 계수 블록에 하나 이상의 역변환들을 적용할 수도 있다. 예를 들어, 역변환 프로세싱 유닛 (308)은 역 DCT, 역 정수 변환, 역 Karhunen-Loeve 변환 (KLT), 역 회전 변환, 역 방향성 변환, 또는 다른 역변환을 변환 계수 블록에 적용할 수도 있다.

[0201] 또한, 예측 프로세싱 유닛 (304)은 엔트로피 디코딩 유닛 (302)에 의해 엔트로피 디코딩된 예측 정보 신택스 엘리먼트들에 따라 예측 블록을 생성한다. 예를 들어, 예측 정보 신택스 엘리먼트들이 현재 블록이 인터-예측됨을 표시하면, 모션 보상 유닛 (316)은 예측 블록을 생성할 수도 있다. 이 경우, 예측 정보 신택스 엘리먼트들은 레퍼런스 블록을 추출할 DPB (314)에서의 레퍼런스 픽처 뿐만 아니라 현재 픽처에서의 현재 블록의 로케이션에 대한 레퍼런스 픽처에서의 레퍼런스 블록의 로케이션을 식별하는 모션 벡터를 표시할 수도 있다. 모션 보상 유닛 (316)은 일반적으로 모션 보상 유닛 (224) (도 3)에 대하여 설명된 것과 실질적으로 유사한 방식으로 인터-예측 프로세스를 수행할 수도 있다.

[0202] 다른 예로서, 예측 정보 신택스 엘리먼트들이 현재 블록이 인트라-예측됨을 표시하면, 인트라-예측 유닛 (318)은 예측 정보 신택스 엘리먼트들에 의해 표시된 인트라-예측 모드에 따라 예측 블록을 생성할 수도 있다. 인트라-예측 유닛 (318)은 일반적으로 인트라-예측 유닛 (226) (도 3)에 대하여 설명된 것과 실질적으로 유사한 방식으로 인트라-예측 프로세스를 수행할 수도 있다. 인트라-예측 유닛 (318)은 DPB (314)로부터 현재 블록에 대한 이웃하는 샘플들의 데이터를 추출할 수도 있다.

[0203] 복원 유닛 (310)은 예측 블록 및 잔차 블록을 사용하여 현재 블록을 복원할 수도 있다. 예를 들어, 복원 유닛 (310)은 잔차 픽셀 블록의 샘플들을 예측 블록의 대응하는 샘플들에 가산하여 현재 블록을 복원할 수도 있다.

[0204] 필터 유닛 (312)은 복원된 블록들에 대해 하나 이상의 필터 동작들을 수행할 수도 있다. 예를 들어, 필터 유닛 (312)은 복원된 블록들의 에지들을 따라 블록화 아티팩트들을 감소시키기 위해 디블록킹 동작들을 수행할 수도 있다. 필터 유닛 (312)의 동작들이 모든 예들에서 반드시 수행되는 것은 아니다.

[0205] 비디오 디코더 (300)는 복원된 블록들을 DPB (314)에 저장할 수도 있다. 예를 들어, 필터 유닛 (312)의 동작들이 수행되는 않은 예들에 있어서, 복원 유닛 (310)이, 복원된 블록들을 DPB (314)에 저장할 수도 있다. 필터 유닛 (312)의 동작들이 수행되는 예들에서, 필터 유닛 (312)은 필터링된 복원된 블록들을 DPB (314)에 저장할 수도 있다. 상기 논의된 바와 같이, DPB (314)는 예측 프로세싱 유닛 (304)에, 후속 모션 보상을 위한 이전에 디코딩된 픽처들 및 인트라-예측을 위한 현재 픽처의 샘플들과 같은 레퍼런스 정보를 제공할 수도 있다. 더욱이, 비디오 디코더 (300)는 도 1의 디스플레이 디바이스 (118)와 같은 디스플레이 디바이스 상으로의 후속 프리젠테이션을 위해 DPB (314)로부터 디코딩된 픽처들 (예컨대, 디코딩된 비디오)을 출력할 수도 있다.

[0206] DPB (314)는 공유되는 다중-계층 DPB (315)를 포함할 수도 있다. 본 개시물의 다른 곳에서 더 상세히 설명되는 바와 같이, 비디오 디코더 (300)는 복수의 계층들에 대한 레퍼런스 픽처들을 공유된 다중-계층 DPB (315)에 저장하도록 구성될 수도 있다. 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 비디오 디코더 (300)는 공유된 다중-계층 DPB (315)에 대해 픽처 출력 및 제거 프로세스를 수행하여 제 1 계층에 속하는 일부 디코딩된 픽처들만을 공유된 다중-계층 DPB (315)로부터 제거할 수도 있다. CPB 메모리 (320)로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 비디오 디코더 (300)는 공유된 다중-계층 DPB (315)의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행할 수도 있다. 즉, 비디오 디코더 (300)는 특정 계층에 제한되지 않고, 대신에, 현재 픽처의 계층 이외의 계층들로부터의 픽처들을 범핑할 수도 있는 픽처 범핑 프로세스를 수행할 수도 있다.

[0207] 이러한 방식으로, 비디오 디코더 (300)는, 비디오 데이터를 저장하도록 구성된 메모리, 및 회로부에서 구현되고 그리고 이하 청구범위 섹션에서 설명된 기법들을 포함하는 본 개시물의 기법들을 수행하도록 구성된 하나 이상의 프로세싱 유닛들을 포함하는, 비디오 디코딩 디바이스의 일 예를 나타낸다.

- [0208] 도 5 는 현재 블록을 인코딩하기 위한 예시적인 프로세스를 예시한 플로우차트이다. 현재 블록은 현재 CU 를 포함할 수도 있다. 비디오 인코더 (200) (도 1 및 도 3) 와 관련하여 설명되었지만, 다른 디바이스들이 도 5 의 프로세스 유사한 프로세스를 수행하도록 구성될 수도 있음을 이해해야 한다.
- [0209] 이 예에 있어서, 비디오 인코더 (200) 는 초기에 현재 블록을 예측한다 (350). 예를 들어, 비디오 인코더 (200) 는 현재 블록에 대한 예측 블록을 형성할 수도 있다. 비디오 인코더 (200) 는 그 후에, 현재 블록에 대한 잔차 블록을 계산할 수도 있다 (352). 잔차 블록을 계산하기 위해, 비디오 인코더 (200) 는 오리지널 의 인코딩되지 않은 블록과 현재 블록에 대한 예측 블록 사이의 차이를 계산할 수도 있다. 그 다음, 비디오 인코더 (200) 는 잔차 블록의 계수들을 변환 및 양자화할 수도 있다 (354). 다음으로, 비디오 인코더 (200) 는 잔차 블록의 양자화된 변환 계수들을 스캔할 수도 있다 (356). 스캔 동안, 또는 스캔에 후속하여, 비디오 인코더 (200) 는 변환 계수들을 엔트로피 인코딩할 수도 있다 (358). 예를 들어, 비디오 인코더 (200) 는 CAVLC 또는 CABAC 를 사용하여 변환 계수들을 인코딩할 수도 있다. 그 다음, 비디오 인코더 (200) 는 블록의 엔트로피 인코딩된 데이터를 출력할 수도 있다 (360).
- [0210] 도 6 은 비디오 데이터의 현재 블록을 디코딩하기 위한 예시적인 프로세스를 예시하는 플로우차트이다. 현재 블록은 현재 CU 를 포함할 수도 있다. 비디오 디코더 (300) (도 1 및 도 4) 와 관련하여 설명되었지만, 다른 디바이스들이 도 6 의 프로세스 유사한 프로세스를 수행하도록 구성될 수도 있음을 이해해야 한다.
- [0211] 비디오 디코더 (300) 는 현재 블록에 대응하는 잔차 블록의 계수들에 대한 엔트로피 인코딩된 예측 정보 및 엔트로피 인코딩된 데이터와 같은, 현재 블록에 대하여 엔트로피 인코딩된 데이터를 수신할 수도 있다 (370). 비디오 디코더 (300) 는 엔트로피 인코딩된 데이터를 엔트로피 디코딩하여 현재 블록에 대한 예측 정보를 결정하고 잔차 블록의 계수들을 재생할 수도 있다 (372). 비디오 디코더 (300) 는 현재 블록에 대한 예측 블록을 계산하기 위해, 예를 들어 현재 블록에 대한 예측 정보에 의해 표시된 바와 같은 인트라- 또는 인터-예측 모드를 사용하여, 현재 블록을 예측할 수도 있다 (374). 그 다음, 비디오 디코더 (300) 는 양자화된 변환 계수들의 블록을 생성하기 위해, 재생된 계수들을 역스캔할 수도 있다 (376). 그 다음, 비디오 디코더 (300) 는 잔차 블록을 생성하기 위해 변환 계수들을 역양자화 및 역변환할 수도 있다 (378). 비디오 디코더 (300) 는 궁극적으로, 예측 블록과 잔차 블록을 결합함으로써 현재 블록을 디코딩할 수도 있다 (380).
- [0212] 도 7 은 비디오 데이터를 디코딩하기 위한 예시적인 프로세스를 예시하는 플로우차트이다. 도 7 의 기법들은 비디오 디코더 (300) (도 1 및 도 4) 와 같은 비디오 디코더에 의해 또는 비디오 인코더 (200) (도 1 및 도 3) 와 같은 비디오 인코더의 비디오 디코딩 루프에 의해 수행될 수도 있다.
- [0213] 비디오 디코더는 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 DPB 를 유지한다 (400). DPB 는 이전에 디코딩된 픽처들을 저장한다. 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함한다.
- [0214] 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, 비디오 디코더는 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거함으로써 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행한다 (402). 비디오 디코더는, 예를 들어, 제 1 계층의 현재 픽처를 디코딩하기 전에 그리고 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행할 수도 있다. 비디오 디코더는, 예를 들어, CPB 로부터 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행할 수도 있다. CPB 는 디코딩을 위해 인코딩된 다중-계층 비디오 데이터를 저장할 수도 있다.
- [0215] CPB 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, 비디오 디코더는 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행할 수도 있다 (404). DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하기 위해, 비디오 디코더는 DPB 로부터 제 2 계층의 적어도 하나의 픽처를 제거할 수도 있다. DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하기 위해, 비디오 디코더는, 예를 들어, 출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않는 것으로 마킹된 레퍼런스 픽처를 DPB 로부터 제거할 수도 있다.
- [0216] 비디오 디코더는 DPB 에서 레퍼런스 픽처를 식별하고; 레퍼런스 픽처에서 현재 픽처의 현재 블록에 대한 예측 블록을 식별하고; 예측 블록에 기초하여 현재 블록을 디코딩하고; 그리고 현재 픽처의 디코딩된 버전을 출력할 수도 있고, 여기서 현재 픽처의 디코딩된 버전은 현재 블록의 디코딩된 버전을 포함한다.
- [0217] 다음의 조항들은 상기에 설명된 기법들 및 디바이스들의 예들을 나타낸다.
- [0218] 조항 1: 다중-계층 비디오 데이터를 디코딩하기 위한 디바이스는, 다중-계층 비디오 데이터를 저장하도록 구성된 메모리, 및 회로부로 구현된 하나 이상의 프로세서들을 포함하고, 하나 이상의 프로세서들은, 복수의 계층들

에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 것으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는 상기 DPB 를 유지하고; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하도록 구성되며, 여기서 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 하나 이상의 프로세서들은 또한, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하고; 그리고 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하도록 구성된다.

- [0219] 조항 2: 조항 1 에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 하나 이상의 프로세서들은 추가로, 제 1 계층의 현재 픽처를 디코딩하기 전에 그리고 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하도록 구성되는, 디바이스.
- [0220] 조항 3: 조항 1 및 조항 2 중 어느 한 조항에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, CPB 로부터 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하도록 구성되는, 디바이스.
- [0221] 조항 4: 조항 1 내지 조항 3 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않은 것으로 마킹된 레퍼런스 픽처를 상기 DPB 로부터 제거하도록 구성되는, 디바이스.
- [0222] 조항 5: 조항 1 내지 조항 4 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하기 위해, 상기 하나 이상의 프로세서들은 추가로, 상기 DPB 로부터 상기 제 2 계층의 적어도 하나의 픽처를 제거하도록 구성되는, 디바이스.
- [0223] 조항 6: 조항 1 내지 조항 5 중 어느 한 조항에 있어서, 상기 하나 이상의 프로세서들은 추가로, DPB 에서 레퍼런스 픽처를 식별하고; 레퍼런스 픽처에서 현재 픽처의 현재 블록에 대한 예측 블록을 식별하고; 예측 블록에 기초하여 현재 블록을 디코딩하고; 그리고 현재 픽처의 디코딩된 버전을 출력하도록 구성되고, 상기 현재 픽처의 디코딩된 버전은 현재 블록의 디코딩된 버전을 포함하는, 디바이스.
- [0224] 조항 7: 조항 1 내지 조항 6 중 어느 한 조항에 있어서, 상기 DPB 는 이전에 디코딩된 픽처들을 저장하도록 구성되고, 상기 CPB 는 인코딩된 다중-계층 비디오 데이터를 저장하도록 구성되는, 디바이스.
- [0225] 조항 8: 조항 1 내지 조항 7 중 어느 한 조항에 있어서, 상기 하나 이상의 프로세서들은 추가로, 상기 메모리에 상기 DPB 및 상기 CPB 를 유지하도록 구성되는, 디바이스.
- [0226] 조항 9: 조항 1 내지 조항 8 중 어느 한 조항에 있어서, 상기 디바이스는 무선 통신 디바이스를 포함하고, 인코딩된 다중-계층 비디오 데이터를 수신하도록 구성된 수신기를 더 포함하는, 디바이스.
- [0227] 조항 10: 조항 9 에 있어서, 상기 무선 통신 디바이스는 전화 핸드셋을 포함하고, 상기 수신기는, 무선 통신 표준에 따라, 인코딩된 다중-계층 비디오 데이터를 포함하는 신호를 복조하도록 구성되는, 디바이스.
- [0228] 조항 11: 조항 1 내지 조항 8 중 어느 한 조항에 있어서, 상기 디바이스는 무선 통신 디바이스를 포함하고, 인코딩된 다중-계층 비디오 데이터를 송신하도록 구성된 송신기를 더 포함하는, 디바이스.
- [0229] 조항 12: 조항 11 에 있어서, 상기 무선 통신 디바이스는 전화 핸드셋을 포함하고, 상기 송신기는, 무선 통신 표준에 따라, 인코딩된 다중-계층 비디오 데이터를 포함하는 신호를 변조하도록 구성되는, 디바이스.
- [0230] 조항 13: 조항 1 내지 조항 12 중 어느 한 조항에 있어서, 현재 픽처의 디코딩된 버전을 포함하는 디코딩된 다중-계층 비디오 데이터를 디스플레이하도록 구성된 디스플레이를 더 포함하는, 디바이스.
- [0231] 조항 14: 조항 1 내지 조항 13 중 어느 한 조항에 있어서, 상기 디바이스는 카메라, 컴퓨터, 모바일 디바이스, 브로드캐스트 수신기 디바이스, 또는 셋탑 박스 중 하나 이상을 포함하는, 디바이스.
- [0232] 조항 15: 다중-계층 비디오 데이터를 디코딩하는 방법으로서, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 단계로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하는 단계; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계로서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계는, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하는 단계를 포함하는, 상기 픽처 출력 및 제거 프로세스를 수행하는 단계; 및 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 단계를 포함한다.

- [0233] 조항 16: 조항 15 에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계는 제 1 계층의 현재 픽처를 디코딩하기 전에 그리고 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계를 포함하는, 방법.
- [0234] 조항 17: 조항 15 및 조항 16 중 어느 한 조항에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계는 CPB 로부터 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 단계를 포함하는, 방법.
- [0235] 조항 18: 조항 15 내지 조항 17 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 단계는 출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않은 것으로 마킹된 레퍼런스 픽처를 상기 DPB 로부터 제거하는 단계를 포함하는, 방법.
- [0236] 조항 19: 조항 15 내지 조항 18 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 단계는 상기 DPB 로부터 상기 제 2 계층의 적어도 하나의 픽처를 제거하는 단계를 포함하는, 방법.
- [0237] 조항 20: 조항 15 내지 조항 19 중 어느 한 조항에 있어서, DPB 에서 레퍼런스 픽처를 식별하는 단계; 레퍼런스 픽처에서 현재 픽처의 현재 블록에 대한 예측 블록을 식별하는 단계; 예측 블록에 기초하여 현재 블록을 디코딩하는 단계; 및 현재 픽처의 디코딩된 버전을 출력하는 단계를 더 포함하고, 상기 현재 픽처의 디코딩된 버전은 현재 블록의 디코딩된 버전을 포함하는, 방법.
- [0238] 조항 21: 조항 20 에 있어서, 현재 픽처의 디코딩된 버전을 디스플레이하는 단계를 더 포함하는, 방법.
- [0239] 조항 22: 조항 15 내지 조항 21 중 어느 한 조항에 있어서, 상기 DPB 는 이전에 디코딩된 픽처들을 저장하도록 구성되고, 상기 CPB 는 인코딩된 다중-계층 비디오 데이터를 저장하도록 구성되는, 방법.
- [0240] 조항 23: 조항 15 내지 조항 20 또는 조항 22 중 어느 한 조항에 있어서, 상기 디코딩하는 방법은 인코딩 프로세스의 일부로서 수행되는, 방법.
- [0241] 조항 24: 명령들을 저장한 컴퓨터 판독가능 저장 매체로서, 상기 명령들은 하나 이상의 프로세서들에 의해 실행될 때, 하나 이상의 프로세서들로 하여금, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하게 하는 것으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는 상기 DPB 를 유지하게 하고; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하는 것으로서, DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금, 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하게 하는, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하고; 그리고 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에, DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하게 한다.
- [0242] 조항 25: 조항 24 에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금, 제 1 계층의 현재 픽처를 디코딩하기 전에 그리고 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하는, 컴퓨터 판독가능 저장 매체.
- [0243] 조항 26: 조항 24 및 조항 25 중 어느 한 조항에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금, CPB 로부터 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하게 하는, 컴퓨터 판독가능 저장 매체.
- [0244] 조항 27: 조항 24 내지 조항 26 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금, 출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않은 것으로 마킹된 레퍼런스 픽처를 상기 DPB 로부터 제거하게 하는, 컴퓨터 판독가능 저장 매체.
- [0245] 조항 28: 조항 24 내지 조항 27 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금, 상기 DPB 로부터 상기 제 2 계층의 적어도 하나의 픽처를 제거하게 하는, 컴퓨터 판독가능 저장 매체.

- [0246] 조항 29: 조항 24 내지 조항 28 중 어느 한 조항에 있어서, 상기 명령들은 추가로, 상기 하나 이상의 프로세서들로 하여금, DPB 에서 레퍼런스 픽처를 식별하고; 레퍼런스 픽처에서 현재 픽처의 현재 블록에 대한 예측 블록을 식별하고; 예측 블록에 기초하여 현재 블록을 디코딩하고; 그리고 현재 픽처의 디코딩된 버전을 출력하게 하고, 상기 현재 픽처의 디코딩된 버전은 현재 블록의 디코딩된 버전을 포함하는, 컴퓨터 판독가능 저장 매체.
- [0247] 조항 30: 조항 24 내지 조항 29 중 어느 한 조항에 있어서, 상기 DPB 는 이전에 디코딩된 픽처들을 저장하도록 구성되고, 상기 CPB 는 인코딩된 다중-계층 비디오 데이터를 저장하도록 구성되는, 컴퓨터 판독가능 저장 매체.
- [0248] 조항 31: 다중-계층 비디오 데이터를 디코딩하기 위한 장치는, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 수단으로서, 상기 복수의 계층들은 적어도 제 1 계층 및 제 2 계층을 포함하는, 상기 DPB 를 유지하는 수단; 제 1 계층의 액세스 유닛의 현재 픽처를 디코딩하기 전에 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단으로서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 것은 제 1 계층에 속하는 디코딩된 픽처들만을 DPB 로부터 제거하는 것을 포함하는, 상기 픽처 출력 및 제거 프로세스를 수행하는 수단; 및 코딩된 픽처 버퍼 (CPB) 로부터 현재 픽처의 최종 디코딩 유닛을 제거한 후에 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 수단을 포함한다.
- [0249] 조항 32: 조항 31 에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단은 제 1 계층의 현재 픽처를 디코딩하기 전에 그리고 현재 픽처의 슬라이스에 대한 슬라이스 헤더를 파싱한 후에, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단을 포함하는, 장치.
- [0250] 조항 33: 조항 31 및 조항 32 중 어느 한 조항에 있어서, 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단은 CPB 로부터 현재 픽처의 제 1 디코딩 유닛을 제거하는 것에 응답하여 상기 DPB 에 대해 픽처 출력 및 제거 프로세스를 수행하는 수단을 포함하는, 장치.
- [0251] 조항 34: 조항 31 내지 조항 33 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 수단은 출력을 위해 필요하지 않은 것으로 마킹되고 참조를 위해 사용되지 않은 것으로 마킹된 레퍼런스 픽처를 상기 DPB 로부터 제거하는 수단을 포함하는, 장치.
- [0252] 조항 35: 조항 31 내지 조항 34 중 어느 한 조항에 있어서, 상기 DPB 의 모든 계층들에 걸쳐 픽처 범핑 프로세스를 수행하는 수단은 상기 DPB 로부터 상기 제 2 계층의 적어도 하나의 픽처를 제거하는 수단을 포함하는, 장치.
- [0253] 조항 36: 조항 31 내지 조항 35 중 어느 한 조항에 있어서, DPB 에서 레퍼런스 픽처를 식별하는 수단; 레퍼런스 픽처에서 현재 픽처의 현재 블록에 대한 예측 블록을 식별하는 수단; 예측 블록에 기초하여 현재 블록을 디코딩하는 수단; 및 현재 픽처의 디코딩된 버전을 출력하는 수단을 더 포함하고, 상기 현재 픽처의 디코딩된 버전은 현재 블록의 디코딩된 버전을 포함하는, 장치.
- [0254] 조항 37: 조항 31 내지 조항 36 중 어느 한 조항에 있어서, 상기 DPB 는 이전에 디코딩된 픽처들을 저장하도록 구성되고, 상기 CPB 는 인코딩된 다중-계층 비디오 데이터를 저장하도록 구성되는, 장치.
- [0255] 조항 38: 조항 31 내지 조항 37 중 어느 한 조항에 있어서, 상기 다중-계층 비디오 데이터를 디코딩하기 위한 장치는 비디오 인코더의 부분인, 장치.
- [0256] 조항 39: 비디오 데이터를 디코딩하는 방법은, 복수의 계층들에 대한 레퍼런스 픽처들을 저장하기 위한 디코딩된 픽처 버퍼 (DPB) 를 유지하는 단계로서, 상기 DPB 는 복수의 계층들 중의 계층에 대한 레퍼런스 픽처들을 저장하기 위한 서브-DPB 를 포함하는, 상기 DPB 를 유지하는 단계; 비디오 데이터의 픽처를 디코딩하는 단계; 및 디코딩된 픽처의 카피를 레퍼런스 픽처로서 DPB 에 저장하는 단계를 포함한다.
- [0257] 조항 40: 조항 39 에 있어서, 서브-DPB 의 충만도를 결정하는 단계를 더 포함하는, 방법.
- [0258] 조항 41: 조항 39 또는 조항 40 에 있어서, DPB 의 충만도를 결정하는 단계를 더 포함하는, 방법.
- [0259] 조항 42: 조항 39 내지 조항 41 중 어느 한 조항에 있어서, 비디오 시퀀스의 제 1 픽처가 파싱되는 것에 응답하여 서브-DPB 의 충만도를 0 으로 초기화하는 단계를 더 포함하는, 방법.
- [0260] 조항 43: 조항 39 내지 조항 41 중 어느 한 조항에 있어서, 최저 nuh_layer_id 를 갖는 CLVSS 픽처 0 이 파싱되는 것에 응답하여 서브-DPB 의 충만도를 0 으로 초기화하는 단계를 더 포함하는, 방법.
- [0261] 조항 44: 조항 39 내지 조항 43 중 어느 한 조항에 있어서, 서브-DPB 에 저장된 픽처들을 상태로 마킹하는 단계

를 더 포함하는, 방법.

- [0262] 조항 45 : 조항 39 내지 조항 44 중 어느 한 조항에 있어서, 상기 DPB 는 상기 복수의 계층들 중 제 2 계층에 대한 레퍼런스 픽처들을 저장하기 위한 제 2 서브-DPB 를 포함하는, 방법.
- [0263] 조항 46: 조항 39 내지 조항 45 중 어느 한 조항에 있어서, 상기 디코딩하는 방법은 인코딩 프로세스의 부분으로서 수행되는, 방법.
- [0264] 조항 47: 비디오 데이터를 코딩하기 위한 디바이스로서, 디바이스는 조항들 39-46 중 임의의 방법을 수행하기 위한 하나 이상의 수단을 포함한다.
- [0265] 조항 48: 조항 47 의 디바이스에 있어서, 하나 이상의 수단은 회로부에서 구현된 하나 이상의 프로세서들을 포함하는, 디바이스.
- [0266] 조항 49: 조항 46 또는 조항 47 에 있어서, 비디오 데이터를 저장하기 위한 메모리를 더 포함하는, 디바이스.
- [0267] 조항 50: 조항 47 내지 조항 49 중 어느 한 조항에 있어서, 디코딩된 비디오 데이터를 디스플레이하도록 구성된 디스플레이를 더 포함하는, 디바이스.
- [0268] 조항 51: 조항 47 내지 조항 50 중 어느 한 조항에 있어서, 상기 디바이스는 카메라, 컴퓨터, 모바일 디바이스, 브로드캐스트 수신기 디바이스, 또는 셋탑 박스 중 하나 이상을 포함하는, 디바이스.
- [0269] 조항 52: 조항 47 내지 조항 51 중 어느 한 조항에 있어서, 디바이스는 비디오 디코더를 포함하는, 디바이스.
- [0270] 조항 53: 조항 47 내지 조항 52 중 어느 한 조항에 있어서, 디바이스는 비디오 인코더를 포함하는, 디바이스.
- [0271] 조항 54: 명령들을 저장한 컴퓨터 판독가능 저장 매체로서, 상기 명령들은, 실행될 경우, 하나 이상의 프로세서들로 하여금, 조항 39 내지 조항 46 중 어느 한 조항의 방법을 수행하게 한다.
- [0272] 예에 의존하여, 본 명세서에서 설명된 기법들 중 임의의 것의 소정의 액트들 또는 이벤트들은 상이한 시퀀스로 수행될 수 있거나, 추가, 병합, 또는 전부 생략될 수도 있다 (예를 들어, 모든 설명된 액트들 또는 이벤트들이 기법들의 실시를 위해 필요한 것은 아니다) 는 것이 인식되어야 한다. 더욱이, 소정의 예들에 있어서, 액트들 또는 이벤트들은 순차적인 것보다는, 예를 들어, 다중-스레딩된 프로세싱, 인터럽트 프로세싱, 또는 다중의 프로세서들을 통해 동시에 수행될 수도 있다.
- [0273] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합에서 구현될 수도 있다. 소프트웨어에서 구현되면, 그 기능들은 컴퓨터 판독가능 매체 상에 하나 이상의 명령들 또는 코드로서 저장 또는 이를 통해 송신될 수도 있고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체들은 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체들, 또는 예를 들어, 통신 프로토콜에 따라, 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이러한 방식으로, 컴퓨터 판독가능 매체들은 일반적으로 (1) 비일시적인 유형의 컴퓨터 판독가능 저장 매체들 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체들은 본 개시에서 설명된 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 이용가능한 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수도 있다.
- [0274] 제한이 아닌 예로서, 그러한 컴퓨터 판독가능 저장 매체들은 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 스토리지, 자기 디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 명령들 또는 데이터 구조들의 형태로 원하는 프로그램 코드를 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터 판독가능 매체로 적절히 명명된다. 예를 들어, 명령들이 동축 케이블, 광섬유 케이블, 트위스티드 페어, 디지털 가입자 라인 (DSL), 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들을 사용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되면, 동축 케이블, 광섬유 케이블, 트위스티드 페어, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 매체의 정의에 포함된다. 그러나, 컴퓨터 판독가능 저장 매체들 및 데이터 저장 매체들은 접속들, 캐리어파들, 신호들, 또는 다른 일시적 매체들을 포함하지 않고, 대신 비일시적, 유형의 저장 매체들에 관련된다는 것을 이해해야 한다. 본 명세서에서 사용된 바와 같이, 디스크 (disk) 및 디스크 (disc) 는 콤팩트 디스크 (CD), 레이저 디스크, 광학 디스크, 디지털 다용도 디스크 (DVD), 플로피 디스크 및 블루-레이 디스크를 포함하며, 여기

서, 디스크 (disk) 들은 통상적으로 데이터를 자기적으로 재생하지만 디스크 (disc) 들은 레이저들을 이용하여 데이터를 광학적으로 재생한다. 상기의 조합들이 또한, 컴퓨터 판독가능 매체들의 범위 내에 포함되어야 한다.

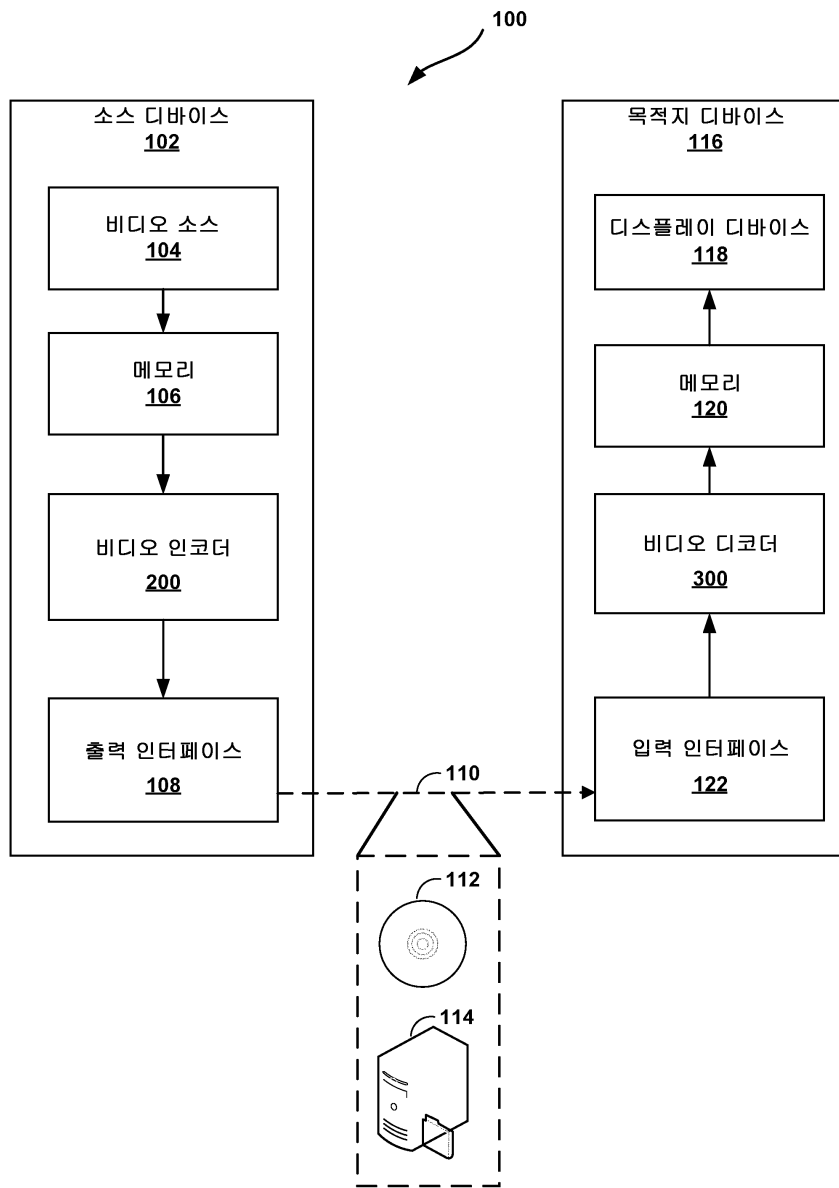
[0275] 명령들은 하나 이상의 프로세서들, 이를 테면 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 또는 다른 등가의 집적 또는 이산 로직 회로부에 의해 실행될 수도 있다. 이에 따라, 본 명세서에서 사용된 바와 같은 용어들 "프로세서" 및 "프로세싱 회로부" 는 전술한 구조들 또는 본 명세서에서 설명된 기법들의 구현에 적합한 임의의 다른 구조 중 임의의 것을 지칭할 수도 있다. 추가로, 일부 양태들에 있어서, 본 명세서에서 설명된 기능은 인코딩 및 디코딩을 위해 구성되거나 또는 결합된 코덱에 통합된 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 그 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수도 있다.

[0276] 본 개시의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예를 들어, 칩 세트) 를 포함한, 광범위한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들, 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하는 것은 아니다. 오히려, 상기 설명된 바와 같이, 다양한 유닛들은 코덱 하드웨어 유닛에서 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 상기 설명된 바와 같은 하나 이상의 프로세서들을 포함하는, 상호운용가능한 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

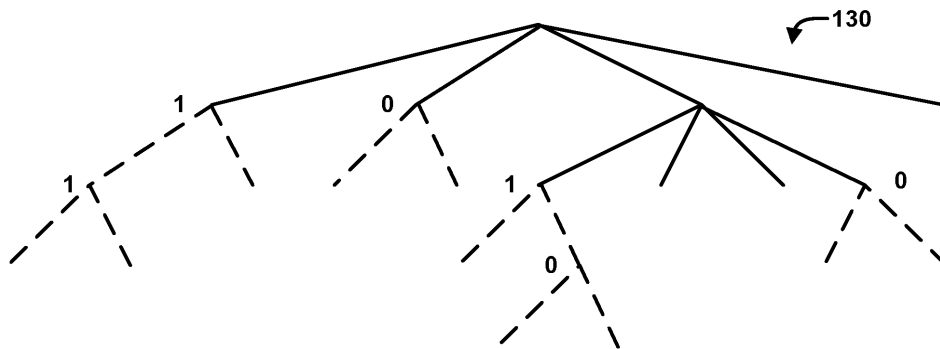
[0277] 다양한 예들이 설명되었다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

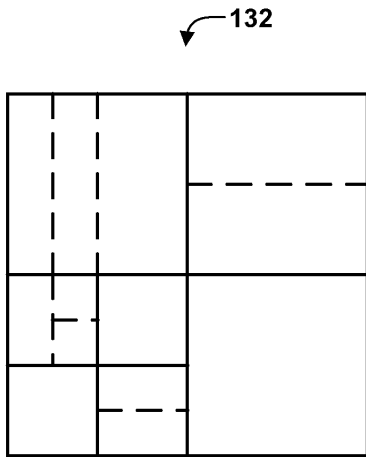
도면1



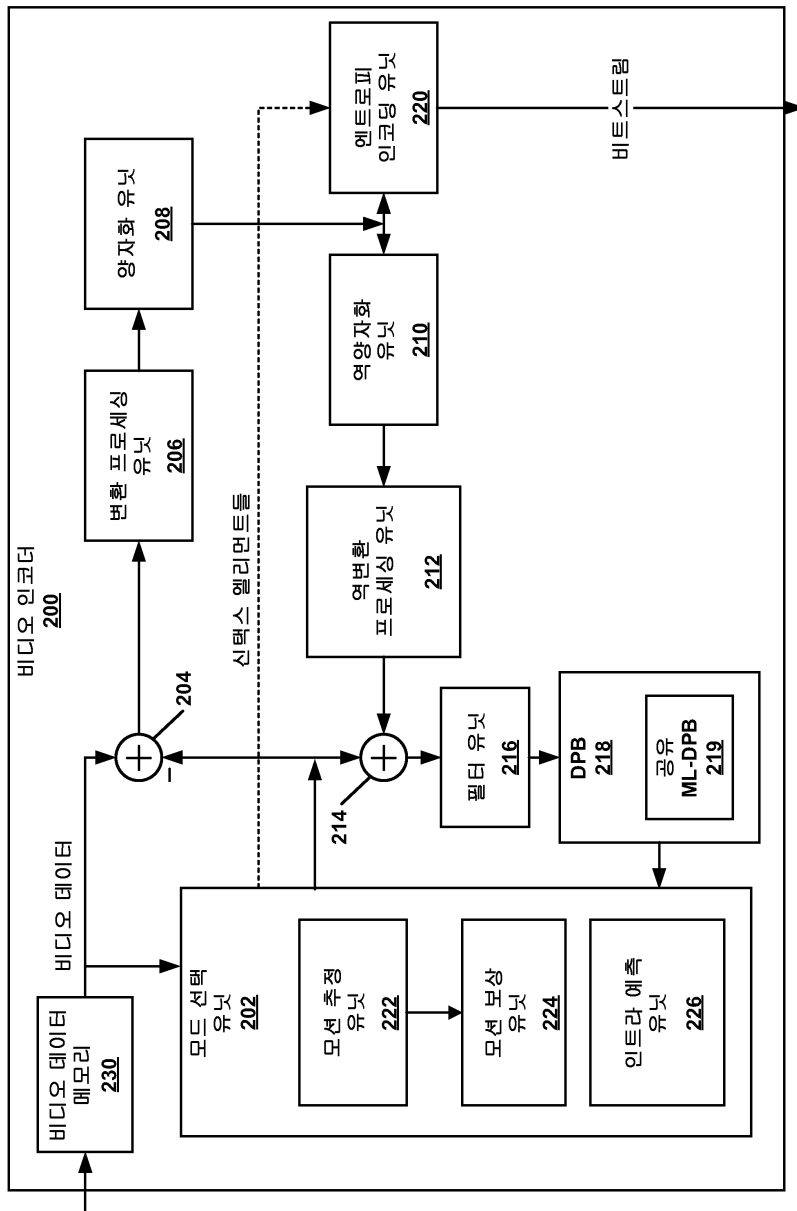
도면2a



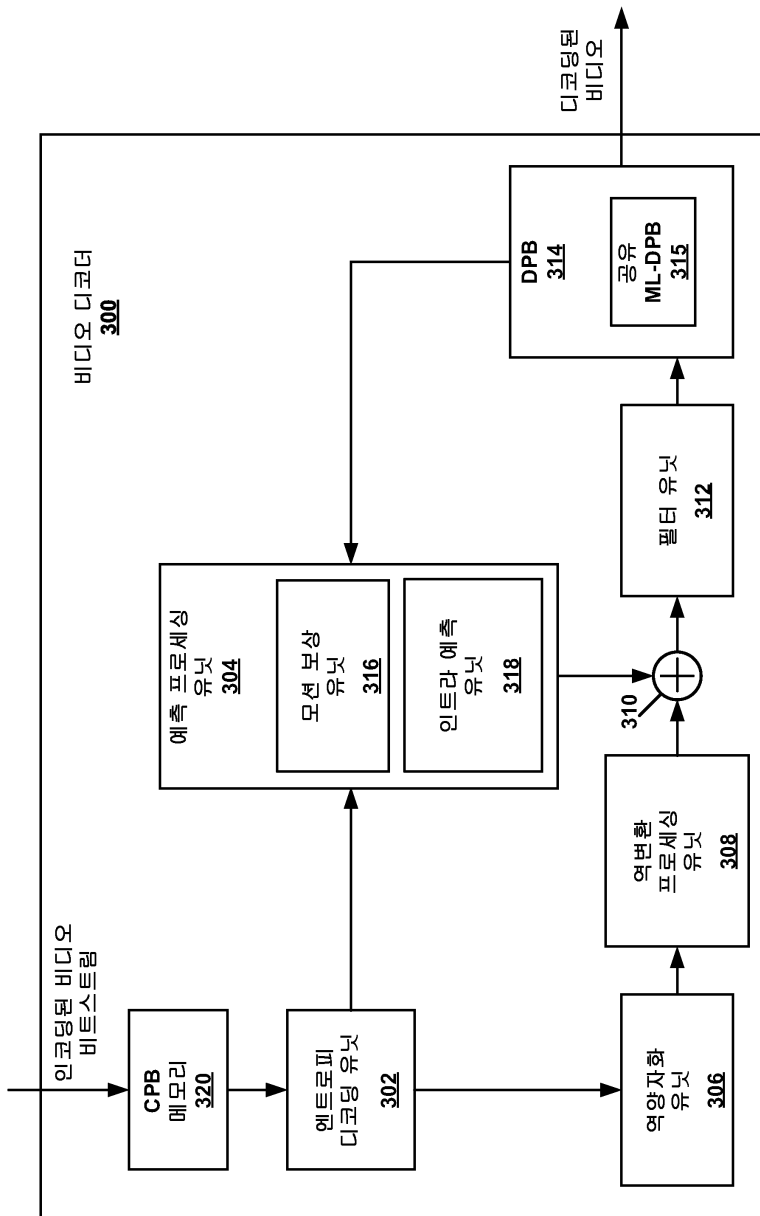
도면2b



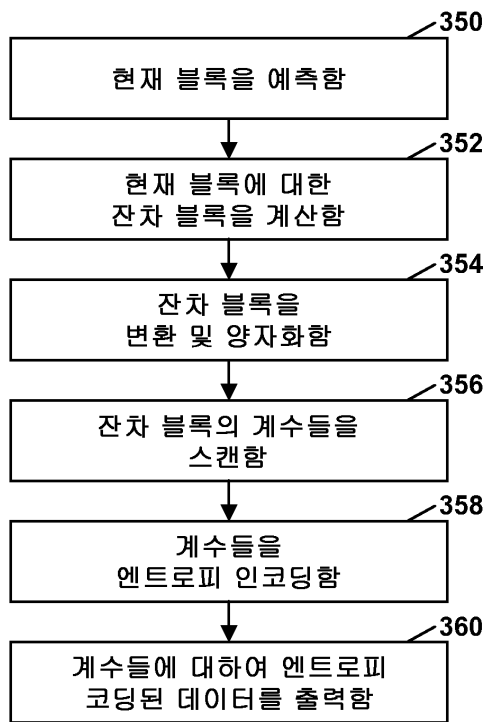
도면3



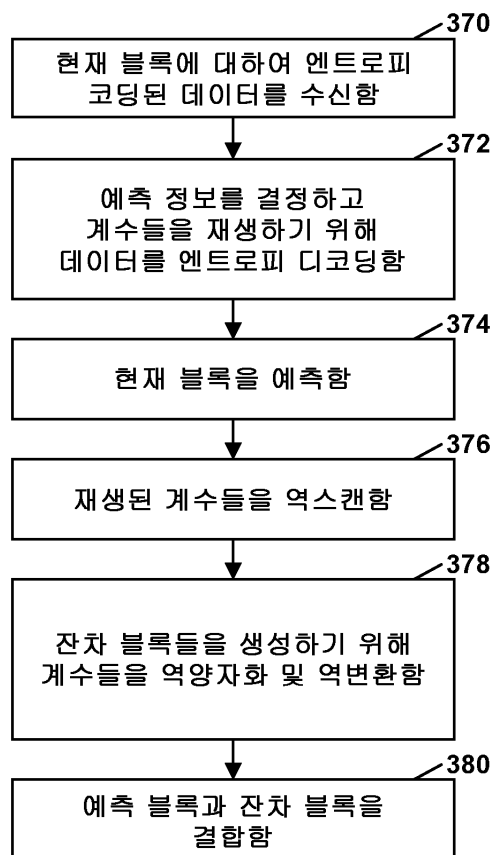
도면4



도면5



도면6



도면7

