



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0019596
(43) 공개일자 2018년02월26일

- | | |
|---|---|
| <p>(51) 국제특허분류(Int. Cl.)
H04N 19/105 (2014.01) H04N 19/503 (2014.01)
H04N 19/573 (2014.01) H04N 19/593 (2014.01)
H04N 19/70 (2014.01)</p> <p>(52) CPC특허분류
H04N 19/105 (2015.01)
H04N 19/503 (2015.01)</p> <p>(21) 출원번호 10-2017-7036777</p> <p>(22) 출원일자(국제) 2016년06월21일
심사청구일자 없음</p> <p>(85) 번역문제출일자 2017년12월20일</p> <p>(86) 국제출원번호 PCT/US2016/038498</p> <p>(87) 국제공개번호 WO 2016/209804
국제공개일자 2016년12월29일</p> <p>(30) 우선권주장
62/183,676 2015년06월23일 미국(US)
15/187,694 2016년06월20일 미국(US)</p> | <p>(71) 출원인
헬컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775</p> <p>(72) 발명자
세레진 바딤
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
라파카 크리쉬나칸트
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(뒷면에 계속)</p> <p>(74) 대리인
특허법인코리아나</p> |
|---|---|

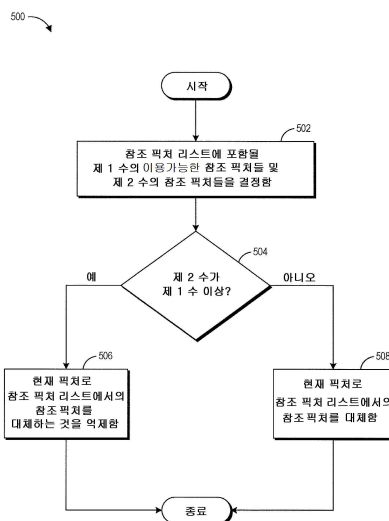
전체 청구항 수 : 총 30 항

(54) 발명의 명칭 **인트라 블록 카페 모드에서의 참조 픽처 리스트 구축**

(57) 요약

참조 픽처 리스트를 구축하도록 구성된 장치는 메모리 및 메모리와 통신하는 하나 이상의 프로세서들을 포함한다. 메모리는 하나 이상의 참조 픽처들을 저장하도록 구성된다. 하나 이상의 프로세서들은, (i) 인터 모드 또는 인트라 블록 카페 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하고, 그리고 제 2 수가 제 1 수 이상이라고 결정하는 것에 응답하여, 예측될 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하도록 구성된다. 하나 이상의 프로세서들은 참조 픽처 리스트에 기초하여 현재 픽처를 인코딩하거나 디코딩할 수도 있다.

대표도 - 도5



(52) CPC특허분류

H04N 19/573 (2015.01)

H04N 19/593 (2015.01)

H04N 19/70 (2015.01)

(72) 발명자

왕 예-쿠이

미국 92121-1714 캘리포니아주 샌디에고 모어하우스
스 드라이브 5775

카르체비츠 마르타

미국 92121-1714 캘리포니아주 샌디에고 모어하우스
스 드라이브 5775

명세서

청구범위

청구항 1

참조 픽처 리스트를 구축하기 위한 방법으로서,

(i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 상기 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하는 단계; 및

상기 제 2 수가 상기 제 1 수 이상이라고 결정하는 것에 응답하여, 예측될 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하는 단계를 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 2

제 1 항에 있어서,

상기 제 2 수가 상기 제 1 수 이상이 아니라고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 3

제 1 항에 있어서,

(i) 인터 모드에서 상기 현재 픽처에서의 블록을 예측하는데 사용하기 위해 이용가능한 참조 픽처들의 총 수와, (ii) 상기 현재 픽처가 상기 현재 픽처에서의 블록을 예측하기 위해 사용되도록 허용될지 여부에 기초한 값과의 총합으로서 상기 제 1 수를 결정하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 4

제 1 항에 있어서,

(i) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 선행하는 상기 참조 픽처들의 총 수와, (ii) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 후속하는 상기 참조 픽처들의 총 수와, (iii) 상기 현재 픽처를 예측하는데 이용가능한 상기 참조 픽처들의 총 수와, (iv) 상기 현재 픽처가 상기 현재 픽처에서의 블록을 예측하는데 이용되도록 허용되는지 여부에 기초한 값과의 총합으로서 상기 제 1 수를 결정하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 5

제 1 항에 있어서,

상기 참조 픽처 리스트가 상기 참조 픽처 리스트의 최종 포지션 이외의 포지션에서 상기 현재 픽처를 포함한다고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트의 상기 최종 포지션에서의 참조 픽처를 대체하는 것을 억제하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 6

제 1 항에 있어서,

참조 픽처 시퀀스의 적어도 일부가 상기 참조 픽처 리스트 내에서 두 번 이상 반복되도록 상기 참조 픽처 리스트가 상기 현재 픽처를 포함하는 상기 참조 픽처 시퀀스를 포함한다고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 7

제 1 항에 있어서,

상기 제 1 수는 고효율 비디오 코딩 (HEVC) 표준에서 변수 NumPicTotalCurr 의 값에 의해 표시되는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 8

제 1 항에 있어서,

상기 제 2 수는 고효율 비디오 코딩 (HEVC) 표준에서 변수 num_ref_idx_l0_active_minus1 의 값과 1 과의 총합에 의해 표시되는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 9

제 1 항에 있어서,

상기 제 1 수는 고효율 비디오 코딩 (HEVC) 표준에서 변수 NumRpsCurrTempList0 의 값에 의해 표시되는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 10

제 1 항에 있어서,

상기 참조 픽처 리스트를 사용하여 상기 현재 픽처에서의 현재 블록에 대해 예측 블록을 결정하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 11

제 10 항에 있어서,

상기 예측 블록에 기초하여 상기 현재 블록을 인코딩하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 12

제 10 항에 있어서,

상기 예측 블록에 기초하여 상기 현재 블록을 디코딩하는 단계를 더 포함하는, 참조 픽처 리스트를 구축하기 위한 방법.

청구항 13

참조 픽처 리스트를 구축하도록 구성된 장치로서,

하나 이상의 참조 픽처들을 저장하도록 구성된 메모리; 및

상기 메모리와 통신하는 하나 이상의 프로세서들을 포함하고,

상기 하나 이상의 프로세서들은,

(i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 상기 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하고; 그리고

상기 제 2 수가 상기 제 1 수 이상이라는 결정에 응답하여, 예측될 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하도록 구성되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 14

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 상기 제 2 수가 상기 제 1 수 이상이 아니라고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하도록 구성되는, 참조 픽처 리스트를 구축하

도록 구성된 장치.

청구항 15

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 또한, (i) 인터 모드에서 상기 현재 픽처에서의 블록을 예측하는데 사용하기 위해 이용가능한 참조 픽처들의 총 수와, (ii) 상기 현재 픽처가 상기 현재 픽처에서의 블록을 예측하기 위해 사용되도록 허용될지 여부에 기초한 값과의 총합으로서 상기 제 1 수를 결정하도록 구성되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 16

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 또한, (i) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 선행하는 단기 참조 픽처들의 총 수와, (ii) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 후속하는 단기 참조 픽처들의 총 수와, (iii) 상기 현재 픽처를 예측하는데 이용가능한 장기 참조 픽처들의 총 수와, (iv) 상기 현재 픽처가 상기 현재 픽처에서의 블록을 예측하는데 사용되도록 허용되는지 여부에 기초한 값과의 총합으로서 상기 제 1 수를 결정하도록 구성되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 17

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 상기 참조 픽처 리스트가 상기 참조 픽처 리스트의 최종 포지션 이외의 포지션에서 상기 현재 픽처를 포함한다고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트의 상기 최종 포지션에서의 참조 픽처를 대체하는 것을 억제하도록 구성되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 18

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 참조 픽처 시퀀스의 적어도 일부가 상기 참조 픽처 리스트 내에서 두번 이상 반복되도록 상기 참조 픽처 리스트가 상기 현재 픽처를 포함하는 상기 참조 픽처 시퀀스를 포함한다고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하도록 구성되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 19

제 13 항에 있어서,

상기 제 1 수는 고효율 비디오 코딩 (HEVC) 표준에서 변수 NumPicTotalCurr 의 값에 의해 표시되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 20

제 13 항에 있어서,

상기 제 2 수는 고효율 비디오 코딩 (HEVC) 표준에서 변수 num_ref_idx_l0_active_minus1 의 값과 1 과의 총합에 의해 표시되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 21

제 13 항에 있어서,

상기 제 1 수는 고효율 비디오 코딩 (HEVC) 표준에서 변수 NumRpsCurrTempList0 의 값에 의해 표시되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 22

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 상기 참조 픽처 리스트를 사용하여 상기 현재 픽처에서의 현재 블록에 대해 예측 블록을 결정하도록 구성되는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 23

제 22 항에 있어서,

상기 장치는, 상기 예측 블록에 기초하여 상기 현재 블록을 인코딩하도록 구성된 비디오 인코더 및 상기 현재 픽처를 캡처하도록 구성된 카메라를 포함하는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 24

제 22 항에 있어서,

상기 장치는, 상기 예측 블록에 기초하여 상기 현재 블록을 디코딩하도록 구성된 비디오 디코더 및 상기 현재 픽처를 디스플레이하도록 구성된 디스플레이를 포함하는, 참조 픽처 리스트를 구축하도록 구성된 장치.

청구항 25

코드를 포함하는 비밀시적인 물리적 컴퓨터 스토리지로서,

상기 코드는, 실행될 때 장치로 하여금,

(i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하게 하고; 그리고

상기 제 2 수가 상기 제 1 수 이상이라는 결정에 응답하여, 예측될 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하게 하는, 비밀시적인 물리적 컴퓨터 스토리지.

청구항 26

제 25 항에 있어서,

상기 코드는 또한, 상기 장치로 하여금, 상기 제 2 수가 상기 제 1 수 이상이 아니라고 결정하는 것에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하게 하는, 비밀시적인 물리적 컴퓨터 스토리지.

청구항 27

제 25 항에 있어서,

상기 코드는 또한, 상기 장치로 하여금, (i) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 선행하는 단기 참조 픽처들의 총 수와, (ii) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 후속하는 단기 참조 픽처들의 총 수와, (iii) 상기 현재 픽처를 예측하는데 이용가능한 장기 참조 픽처들의 총 수와, (iv) 상기 현재 픽처가 상기 현재 픽처에서의 블록을 예측하는데 사용되도록 허용되는지 여부에 기초한 값과의 총합으로서 상기 제 1 수를 결정하게 하는, 비밀시적인 물리적 컴퓨터 스토리지.

청구항 28

참조 픽처 리스트를 구축하도록 구성된 비디오 코딩 디바이스로서,

하나 이상의 참조 픽처들을 저장하는 수단;

(i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 상기 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하는 수단; 및

상기 제 2 수가 상기 제 1 수 이상이라는 결정에 응답하여, 예측될 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하는 수단을 포함하는, 참조 픽처 리스트를 구축하도록 구성된 비디오 코딩 디바이스.

청구항 29

제 28 항에 있어서,

상기 제 2 수가 상기 제 1 수 이상이 아니라는 결정에 응답하여, 상기 현재 픽처로 상기 참조 픽처 리스트에서의 참조 픽처를 대체하는 수단을 더 포함하는, 참조 픽처 리스트를 구축하도록 구성된 비디오 코딩 디바이스.

청구항 30

제 28 항에 있어서,

(i) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 선행하는 단기 참조 픽처들의 총 수와, (ii) 상기 현재 픽처를 예측하는데 이용가능하고 출력 순서에 있어서 상기 현재 픽처를 후속하는 단기 참조 픽처들의 총 수와, (iii) 상기 현재 픽처를 예측하는데 이용가능한 장기 참조 픽처들의 총 수와, (iv) 상기 현재 픽처가 상기 현재 픽처에서의 블록을 예측하는데 사용되도록 허용되는지 여부에 기초한 값과의 총합으로서 상기 제 1 수를 결정하게 하기 위한 수단을 더 포함하는, 참조 픽처 리스트를 구축하도록 구성된 비디오 코딩 디바이스.

발명의 설명

기술 분야

[0001] 이 개시물은 비디오 코딩 및 압축 분야에 관한 것이고, 특히 스크린 콘텐츠 코딩에 관한 것이다.

배경 기술

[0002] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 휴대정보 단말기 (PDA) 들, 랩탑 또는 데스크탑 컴퓨터들, 디지털 카메라들, 디지털 리코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 원격 화상회의 디바이스들 등을 포함하는, 매우 다양한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2 (Moving Picture Experts Group-2), MPEG-4, ITU-T (International Telegraph Union-Telecommunication Standardization Sector) H.263, ITU-T H.264/MPEG-4, 파트 10, 어드밴스드 비디오 코딩 (AVC), 고효율 비디오 코딩 (HEVC) 표준에 의해 정의된 표준들, 및 이러한 표준들의 확장판들에 기재된 것과 같은 비디오 압축 기법들을 구현한다. 비디오 디바이스들은 그러한 비디오 코딩 기법들을 구현하는 것에 의해 디지털 비디오 정보를 보다 효율적으로 송신, 수신, 인코딩, 디코딩 및/또는 저장할 수도 있다.

[0003] 고속 인터넷 액세스의 보급으로, 신흥 비디오 어플리케이션들, 예컨대 원격 데스크탑 공유, 가상 데스크탑 인프라구조, 및 무선 디스플레이는 스크린 콘텐츠의 높은 압축 효율을 필요로 한다. 하지만, 부가 인트라 및 인터 비디오 코딩 툴들은 자연적 콘텐츠에 대해 주로 설계되었다. 스크린 콘텐츠는 자연적 콘텐츠 (예를 들어, 예리한 에지들 및 적거나 없는 잡음) 와 비교하여 상당히 상이한 특성들을 가지며, 이는 그러한 전형적인 코딩 툴들을 덜 충분하게 한다.

발명의 내용

과제의 해결 수단

[0004] 비디오 코딩에서, 이전에 디코딩된 픽처들은 다른 픽처들을 코딩하기 위한 예측기들로서 사용될 수도 있다. 그러한 이전에 디코딩된 픽처들은 참조 픽처 리스트에 추가되고 참조 인덱스들을 사용하여 참조 픽처 내에서 참조될 수도 있다. 그러한 참조 픽처 리스트는 인터 모드에서 픽처, 슬라이스 또는 블록을 코딩하기 위해 사용될 수도 있고, 여기서 시간적으로 이웃하는 픽처들은 참조 픽처에 추가되고 현재 픽처/슬라이스/블록에 대해 예측기들로서 사용될 수도 있다. 일부 경우들에서, 그러한 참조 픽처 리스트는 또한 인트라 블록 카피 모드에서 픽처/슬라이스/블록을 코딩하기 위해 사용될 수도 있고, 여기서 픽처 내에서 이전에 재구성된 블록들은 동일한 픽처 내에서 다른 블록을 코딩하는데 사용될 수 있다. 그러한 경우, 현재 픽처는 참조 픽처 리스트에

부가되고 참조 인덱스를 사용하여 참조될 수 있다.

- [0005] 일부 구현들에서, 인트라 블록 카피 모드가 사용을 위해 인에이블될 때, 현재 픽처가 항상 참조 픽처 리스트에 부가될 수도 있다. 예를 들어, 코더 (예를 들어, 인코더 또는 디코더) 는, 참조 픽처 리스트 구축 프로세서의 종료 시, 현재 픽처로 참조 픽처 리스트에서의 참조 픽처들 중 하나를 대체할 수도 있어서, 현재 픽처가 인터 모드 또는 인트라 블록 카피 모드에서 블록을 코딩하기 위해 사용될 수 있다. 하지만, 일부 경우들에서, 현재 픽처를 참조 픽처 리스트에 자동으로 부가하는 것은, 인트라 블록 카피 모드가 사용을 위해 인에이블될 때에도, 소정의 단점들을 초래할 수도 있다.
- [0006] 따라서, 인터 모드 및 인트라 블록 카피 모드를 위한 개선된 참조 픽처 리스트 구축 방법이 요망된다.
- [0007] 이 개시물의 시스템들, 방법들 및 디바이스들은 각각 몇몇 혁신적인 양태들을 가지며, 이들 중 단 하나만이 본 명세서에 개시된 바람직한 속성들을 담당하지 않는다.
- [0008] 일 양태에서, 참조 픽처 리스트를 구축하기 위한 방법은, (i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하는 단계, 및 제 2 수가 제 1 수 이상이라고 결정하는 것에 응답하여, 예측될 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하는 단계를 포함한다.
- [0009] 다른 양태에서, 참조 픽처 리스트를 구축하기 위한 장치는 메모리 및 메모리와 통신하는 하나 이상의 프로세서들을 포함한다. 메모리는 하나 이상의 참조 픽처들을 저장하도록 구성된다. 하나 이상의 프로세서들은, (i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하고, 그리고 제 2 수가 제 1 수 이상이라고 결정하는 것에 응답하여, 예측될 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하도록 구성된다.
- [0010] 또 다른 양태에서, 비일시적인 물리적 컴퓨터 스토리지는 코드를 포함하고, 코드는, 실행될 때 장치로 하여금, (i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하게 하고, 그리고 제 2 수가 제 1 수 이상이라고 결정하는 것에 응답하여, 예측될 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하게 한다.
- [0011] 또 다른 양태에서, 참조 픽처 리스트를 구축하도록 구성된 비디오 코딩 디바이스는, 하나 이상의 참조 픽처들을 저장하는 수단, (i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정하는 수단, 및 제 2 수가 제 1 수 이상이라고 결정하는 것에 응답하여, 예측될 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제하는 수단을 포함한다.

도면의 간단한 설명

- [0012] 도 1a 는 이 개시물에 기재된 양태들에 따른 기법들을 활용할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템을 도시하는 블록 다이어그램이다.
- 도 1b 는 이 개시물에 기재된 양태들에 따른 기법들을 수행할 수도 있는 또 다른 예의 비디오 인코딩 및 디코딩 시스템을 도시하는 블록 다이어그램이다.
- 도 2 는 이 개시물에 기재된 양태들에 따른 기법들을 구현할 수도 있는 비디오 인코더의 일 예를 도시하는 블록 다이어그램이다.
- 도 3 은 이 개시물에 기재된 양태들에 따른 기법들을 구현할 수도 있는 비디오 디코더의 일 예를 도시하는 블록 다이어그램이다.
- 도 4 는 이 개시물에 기재된 양태들에 따른 일 예의 참조 픽처 리스트 구축 프로세스이다.
- 도 5 는 본 개시물의 양태들에 따른 참조 픽처 리스트를 구축하기 위한 방법을 도시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

- [0013] 하기의 기재에서, 소정의 실시형태들에 관련된 H.264/어드밴스드 비디오 코딩 (AVC) 기법들이 개시되고, HEVC

표준 및 관련된 기법들이 또한 논의된다. 소정의 실시형태들은 본 명세서에서 HEVC 및/또는 H.264 표준들의 콘텍스트에서 기재되지만, 당업자는 본 명세서에 개시된 시스템들 및 방법들은 임의의 적절한 비디오 코딩 표준에 적용가능함을 이해할 것이다. 예를 들어, 본 명세서에 개시된 실시형태들은 다음의 표준들 중 하나 이상에 적용가능할 수도 있다: 범위 확장을 포함한, 국제 텔레통신 연합 (ITU) 텔레통신 표준화 섹터 (ITU-T) H.261, 표준화를 위한 국제 기구/국제 전기기술 위원회 (ISO/IEC) MPEG 1 비주얼, ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG 4 비주얼, ITU-T H.264 (ISO/IEC MPEG-4 AVC 로서 또한 알려짐).

[0014] HEVC 는 일반적으로 많은 양태들에서 이전 비디오 코딩 표준들의 프레임워크를 따른다. HEVC 에서의 예측 유닛은 소정의 이전 비디오 코딩 표준들에서의 예측 유닛들 (예를 들어, 매크로블록들) 과 상이하다. 사실상, 매크로블록의 개념은 소정의 이전 비디오 코딩 표준들에서 이해되는 바와 같이 HEVC 에 존재하지 않는다. 매크로블록은 다른 가능한 이익들 중 높은 유연성을 제공할 수도 있는, 쿼드트리 스킵에 기초하여 계위적 구조에 의해 대체된다. 예를 들어, HEVC 스킵 내에서, 3 가지 타입의 블록들, 코딩 유닛 (CU), 예측 유닛 (PU), 및 변환 유닛 (TU) 이 정의된다. CU 는 영역 스플릿팅의 기본 유닛을 지칭할 수도 있다. CU 는 매크로블록의 개념과 유사한 것으로 고려될 수도 있지만, HEVC 는 CU들의 최대 사이즈를 한정하지 않으며 4 개의 동일한 사이즈 CU들로의 회귀적 스플릿팅을 허용하여 콘텐츠 적응성을 개선할 수도 있다. PU 는 인터/인트라 예측의 기본 유닛으로 고려될 수도 있으며, 단일 PU 는 다중의 임의적 형상 파티션들을 포함하여 불규칙적인 이미지 패티들을 효율적으로 코딩할 수도 있다. TU 는 변환의 기본 유닛으로 고려될 수도 있다. TU 는 PU 로부터 독립적으로 정의될 수도 있다; 하지만, TU 의 사이즈는 TU 가 속하는 CU 의 사이즈로 제한될 수도 있다. 3 개의 상이한 개념들로의 블록 구조의 이러한 분리는 각각의 유닛이 유닛의 개별 역할에 따라 최적화되도록 할 수도 있으며, 이는 개선된 코딩 효율을 야기할 수도 있다.

[0015] 비디오 코딩 표준들

[0016] 디지털 이미지, 예컨대 비디오 이미지, TV 이미지, 스틸 이미지 또는 비디오 레코더 또는 컴퓨터에 의해 생성된 이미지는, 수평 및 수직 라인들로 배열된 픽셀들 및 샘플들을 포함할 수도 있다. 단일 이미지에서의 픽셀들의 수는 통상적으로 몇만이다. 각각의 픽셀은 통상적으로 루미넌스 및 크로미넌스 정보를 포함한다. 압축 없이, 이미지 인코더로부터 이미지 디코더로 전달될 정보의 순전한 양은 비현실적인 실시간 이미지 송신을 렌더링하게 된다. 송신될 정보의 양을 감소시키기 위해, 많은 상이한 압축 방법들, 예컨대 JPEG, MPEG 및 H.263 표준들이 개발되고 있다.

[0017] 비디오 코딩 표준들은 스케일러블 비디오 코딩 (SVC) 및 멀티뷰 비디오 코딩 (MVC) 확장들을 포함한, ITU-T H.261, ISO/IEC MPEG-1 비주얼, ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG-4 비주얼 및 ITU-T H.264 (ISO/IEC MPEG-4 AVC 로서 또한 알려짐) 을 포함한다.

[0018] 부가적으로, 비디오 코딩 표준, 즉 HEVC 는 ITU-T 비디오 코딩 전문가 그룹 (VCEG) 및 ISO/IEC MPEG 의 비디오 코딩에 대한 공동 협력 팀 (JCT-VC) 에 의해 개발되고 있다. HEVC 사양은 “Recommendation ITU-T H.265” 의 명칭 하에서 ITU 웹사이트 상에서 입수가 가능하다. HEVC 스크린 콘텐츠 코딩의 초안은 문서 번호 JCTVC-W0105-v3 에서의 JCT-VC 문서 관리 시스템 상에서 입수가 가능하다.

[0019] 초기 고려사항들

[0020] 이하, 신규 시스템들, 장치들 및 방법들의 다양한 양태들은 첨부 도면들을 참조하여 보다 충분히 기재된다. 하지만, 이 개시물은 많은 상이한 형태들로 구현될 수도 있고, 이 개시물 전체에 걸쳐 제시된 임의의 특정 구조 또는 기능에 제한되는 것으로 해석되지 않아야 한다. 오히려, 이들 양태들은 이 개시물이 철저하고 완전해지도록 제공되고, 당업자에게 개시물의 범위를 충분히 전달할 것이다. 본 명세서에 기재된 기법들에 기초하여, 당업자는, 본 개시물의 임의의 다른 양태와 관계 없이 구현되는 또는 이 양태와 결합되는, 개시물의 범위가 본 명세서에 개시된 신규 시스템들, 장치들, 및 방법들의 임의의 양태를 커버하도록 의도된다는 것을 이해해야 한다. 예를 들어, 본 명세서에 기술된 임의의 수의 양태들을 사용하여 장치가 구현될 수도 있고 또는 방법이 실시될 수도 있다. 부가적으로, 본 개시물의 범위는 본 명세서에 기술된 본 개시물의 다양한 양태들에 부가하여 또는 이들 양태들 이외에, 다른 구조, 기능 또는 구조 및 기능을 사용하여 실시되는 그러한 장치 또는 방법을 커버하도록 의도된다. 본 명세서에 개시된 임의의 양태는 청구항의 하나 이상의 엘리먼트들에 의해 구현될 수도 있다는 것을 이해해야 한다.

[0021] 특정 양태들이 본 명세서에 기재되지만, 이들 양태들의 많은 변형들 및 치환들이 개시물의 범위 내에 포함된다. 바람직한 양태들의 일부 이익들 및 이점들이 언급되지만, 개시물의 범위는 특정 이익들, 사용들 또는 목적들

에 제한되도록 의도되지 않는다. 오히려, 개시물의 양태들은 상이한 무선 기술들, 시스템 구성들, 네트워크들 및 송신 프로토콜들에 넓게 적용가능하도록 의도되고, 이들 중 일부는 바람직한 양태들의 다음의 설명에서 그리고 도면들에서 예시로서 도시된다. 상세한 설명 및 도면들은 제한하기 보다는 오히려 개시물을 예시하며, 개시물의 범위는 첨부된 청구항들 및 그 등가물들에 의해 정의된다.

[0022] 첨부된 도면들은 예를 도시한다. 첨부된 도면들에서 참조 번호들에 의해 표시된 엘리먼트들은 다음의 기재에서 같은 참조 번호들에 의해 표시된 엘리먼트들에 대응한다. 이 개시물에 있어서, 서수 단어 (예를 들어, "제 1", "제 2", "제 3" 등) 로 시작하는 명칭을 갖는 엘리먼트들은, 그 엘리먼트들이 반드시 특정 순서를 갖는 것을 나타내지 않는다. 오히려, 그러한 서수 단어는 단지 동일하거나 유사한 타입의 상이한 엘리먼트들을 지칭하기 위해 사용된다.

[0023] 비디오 코딩 시스템

[0024] 도 1a 는 이 개시물에 기재된 양태들에 따른 기법들을 활용할 수도 있는 일 예의 비디오 코딩 시스템 (10) 을 도시하는 블록 다이어그램이다. 본 명세서에 기재된 바와 같이, 용어 "비디오 코더" 또는 "코더" 는 일반적으로 비디오 인코더들 및 비디오 디코더들의 양자를 지칭한다. 이 개시물에서, 용어들 "비디오 코딩" 또는 "코딩" 은 일반적으로 비디오 인코딩 및 비디오 디코딩을 지칭할 수도 있다. 비디오 인코더들 및 비디오 디코더들에 부가하여, 본 출원에 기재된 양태들은 트랜스코더들 (예를 들어, 비트스트림을 디코딩하고 또 다른 비트스트림을 재인코딩할 수 있는 디바이스들) 및 중간박스들 (예를 들어, 비트스트림을 수정, 변환, 및/또는 그렇지 않으면 조종할 수 있는 디바이스들) 과 같은 다른 관련된 디바이스들로 확장될 수도 있다.

[0025] 도 1a 에 나타난 바와 같이, 비디오 코딩 시스템 (10) 은 목적지 디바이스 (14) 에 의해 나중에 디코딩될 인코딩된 비디오 데이터를 생성하는 소스 디바이스 (12) 를 포함한다. 도 1a 의 예에서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 별도의 디바이스들을 구성한다. 하지만, 소스 디바이스 (12) 및 목적지 디바이스 (14) 가 도 1b 에 나타난 바와 같이, 동일한 디바이스 상에 또는 그 부분 상에 있을 수도 있음을 유의한다.

[0026] 도 1a 를 다시 참조하면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 각각 데스크탑 컴퓨터들, 노트북 (랩탑) 컴퓨터들, 테블릿 컴퓨터들, 셋톱 박스들, 이른바 "스마트" 폰들과 같은 전화 핸드셋들, 이른바 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함하는 광범위한 범위의 디바이스들 중 임의의 것을 포함할 수 있다. 다양한 실시형태들에서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신을 위해 구비될 수도 있다.

[0027] 목적지 디바이스 (14) 는 디코딩된 인코딩된 비디오 데이터를 링크 (16) 를 통해 수신할 수도 있다. 링크 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 에서 목적지 디바이스 (14) 로 이동시킬 수 있는 매체 또는 디바이스의 임의의 유형을 포함할 수도 있다. 도 1a 의 예에서, 링크 (16) 는 소스 디바이스 (12) 가 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 실시간으로 송신하게 하는 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조될 수도 있고, 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 임의의 무선 또는 유선 통신 매체, 예컨대 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들을 포함할 수도 있다. 통신 매체는 패킷 기반 네트워크, 예컨대 로컬 영역 네트워크 (local area network), 광역 네트워크 (wide-area network), 또는 인터넷과 같은 글로벌 네트워크의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 에서 목적지 디바이스 (14) 로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0028] 대안으로, 인코딩된 데이터는 출력 인터페이스 (22) 에서 저장 디바이스 (31)(옵션으로 존재) 로 출력될 수도 있다. 유사하게, 인코딩된 데이터는 예를 들어, 목적지 디바이스 (14) 의 입력 인터페이스 (28) 에 의해 저장 디바이스 (31) 로부터 액세스될 수도 있다. 저장 디바이스 (31) 는 하드 드라이브, 플래시 메모리, 휘발성 또는 비휘발성 메모리와 같은 임의의 다양한 분산된 또는 국부적으로 액세스된 데이터 저장 매체, 또는 인코딩된 비디오 데이터를 저장하는 임의의 다른 적합한 디지털 저장 매체를 포함할 수도 있다. 추가 예에서, 저장 디바이스 (31) 는 소스 디바이스 (12) 에 의해 생성된 인코딩된 비디오를 유지할 수도 있는 다른 중간 저장 디바이스 또는 파일 서버에 대응할 수도 있다. 목적지 디바이스 (14) 는 스트리밍 또는 다운로드를 통해 저장 디바이스 (31) 로부터의 저장된 비디오 데이터에 액세스할 수도 있다. 파일 서버는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 송신할 수 있는 임의의 유형의 서버일 수도 있다. 예시의 파일 서버들은 (예를 들어, 웹사이트용의) 웹 서버, FTP (File Transfer Protocol) 서버, NAS (network attached storage) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바

이스 (14) 는 인터넷 접속을 포함하는 임의의 표준 데이터 접속을 통해 인코딩된 비디오 데이터에 액세스할 수도 있다. 이것은 파일 서버 상에 저장된 인코딩된 비디오 데이터를 액세스하는데 적합한 무선 채널 (예를 들어, 무선 로컬 영역 네트워크 (WLAN) 접속), 유선 접속 (예를 들어, 디지털 가입자 라인 (DSL), 케이블 모뎀 등), 또는 이들의 조합을 포함할 수도 있다. 저장 디바이스 (31) 로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이들의 조합일 수도 있다.

[0029] 본 개시물의 기법들은 무선 어플리케이션들 또는 설정들에 제한되지 않는다. 기법들은 다양한 멀티미디어 어플리케이션들, 예컨대, 지상파 (over-the-air) 텔레비전 방송들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예를 들어, 인터넷 (예를 들어, 하이퍼텍스트 전송 프로토콜 [HTTP] 을 통한 동적 적응형 스트리밍) 을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 어플리케이션들 중 임의의 것의 지원에서 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 비디오 코딩 시스템 (10) 은, 비디오 스트리밍, 비디오 플레이백, 비디오 방송, 및/또는 비디오 전화와 같은 어플리케이션들을 지원하기 위해 일방향 또는 양방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0030] 도 1a 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20) 및 출력 인터페이스 (22) 를 포함한다. 일부 경우들에서, 출력 인터페이스 (22) 는 변조기/복조기 (모뎀) 및/또는 송신기를 포함할 수도 있다. 소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡처 디바이스와 같은 소스, 예를 들어 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 피드 인터페이스, 및/또는 소스 비디오로서 컴퓨터 그래픽스 데이터를 생성하는 컴퓨터 그래픽스 시스템, 또는 그러한 소스들의 조합을 포함할 수도 있다. 일 예로서, 비디오 소스 (18) 가 비디오 카메라인 경우, 도 1b 의 예에 도시된 바와 같이 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 이른바 "카메라 폰들" 또는 "비디오 폰들" 을 형성할 수도 있다. 하지만, 본 개시물에 기재된 기법들은 일반적으로 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 어플리케이션들에 적용될 수도 있다.

[0031] 캡처된, 미리 캡처된, 컴퓨터 생성된 비디오가 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 데이터는 소스 디바이스 (12) 의 출력 인터페이스 (22) 를 통해 목적지 디바이스 (14) 에 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 (또는 대안으로), 디코딩 및/또는 플레이백을 위해, 목적지 디바이스 (14) 및 다른 디바이스들에 의한 이후의 액세스를 위해 저장 디바이스 (31) 상에 저장될 수도 있다. 도 1a 및 도 1b 에 도시된 비디오 인코더 (20) 는 도 2 에 도시된 비디오 인코더 (20) 또는 본 명세서에 기재된 임의의 다른 비디오 인코더를 포함할 수도 있다.

[0032] 도 1a 의 예에서, 목적지 디바이스 (14) 는 입력 인터페이스 (28), 비디오 디코더 (30) 및 디스플레이 디바이스 (32) 를 포함한다. 일부 경우들에서, 입력 인터페이스 (28) 는 수신기 및/또는 모뎀을 포함할 수도 있다. 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 링크 (16) 를 통해 및/또는 저장 디바이스 (31) 로부터 인코딩된 비디오 데이터를 수신할 수도 있다. 링크 (16) 를 통해 통신되거나 저장 디바이스 (31) 상에 저장된 인코딩된 비디오 데이터는, 비디오 데이터의 디코딩에 있어서, 비디오 디코더 (30) 와 같은 비디오 디코더에 의한 사용을 위해 비디오 인코더 (20) 에 의해 생성된 다양한 신택스 엘리먼트들을 포함할 수도 있다. 그러한 신택스 엘리먼트들은 통신 매체 상에서 송신되거나, 저장 매체 상에 저장되거나, 또는 파일 서버상에 저장된 인코딩된 비디오 데이터로 포함될 수도 있다. 도 1a 및 도 1b 에 도시된 비디오 디코더 (30) 는 도 3 에 도시된 비디오 디코더 (30) 또는 본 명세서에 기재된 임의의 다른 비디오 디코더를 포함할 수도 있다.

[0033] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합되거나 목적지 디바이스 (14) 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합형 디스플레이 디바이스를 포함할 수도 있고, 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 디스플레이의 다른 유형과 같은 다양한 디스플레이 디바이스들 중 임의의 것을 포함할 수도 있다.

[0034] 관련된 양태들에서, 도 1b 는 일 예의 비디오 코딩 시스템 (10') 을 나타내며, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 디바이스 (11) 상에 있거나 디바이스 (11) 의 부분이다. 디바이스 (11) 는 "스마트" 폰 등과 같은 전화기 핸드셋일 수도 있다. 디바이스 (11) 는 소스 디바이스 (12) 및 목적지 디바이스 (14) 와 동작가능하게 통신하는 (옵션으로 존재하는) 제어기/프로세서 디바이스 (13) 를 포함할 수도 있다. 도 1b 의 비디오 코딩 시스템 (10') 및 그 컴포넌트들은, 다르게는 도 1a 의 비디오 코딩 시스템 (10) 및 그 컴포넌트

들과 유사하다.

- [0035] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 HEVC 와 같은 비디오 압축표준에 따라 동작할 수도 있고, HEVC 테스트 모델 (HM) 에 순응할 수도 있다. 대안으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는, ITU-T H.264 표준과 같은, 대안으로 MPEG 4, 파트 10, AVC, 또는 그러한 표준들의 확장들로 지칭되는, 다른 전매 또는 산업 표준들에 따라 동작할 수도 있다. 하지만, 이 개시물의 기법들은 임의의 특정 코딩 표준에 제한되지 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 를 포함한다.
- [0036] 도 1a 및 도 1b 의 예들에는 나타나지 않았지만, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있고, 적절한 MUX-DEMUX 유닛들 또는 다른 하드웨어 및 소프트웨어를 포함하여, 공통 데이터 스트림 또는 별도의 데이터 스트림들에서 오디오 및 비디오 양자 모두의 인코딩을 핸들링할 수도 있다. 적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대 사용자 데이터그램 프로토콜 (UDP) 에 순응할 수도 있다.
- [0037] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각 임의의 다양한 적합한 인코더 회로부, 예컨대 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적 회로들 (ASIC들), 필드 프로그램가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어, 또는 이들의 임의의 조합들로서 구현될 수도 있다. 기법들이 부분적으로 소프트웨어로 구현되는 경우, 디바이스는 그 소프트웨어에 대한 명령들을 적합한 비일시적 컴퓨터 판독가능 매체에 저장할 수도 있고, 본 개시물의 기법들을 수행하기 위해 하나 이상의 프로세서들을 사용하여 하드웨어에서 그 명령들을 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있고, 이들 중 어느 하나는 결합된 인코더/디코더의 부분으로서 각각의 디바이스에 통합될 수도 있다.
- [0038] 비디오 코딩 프로세스
- [0039] 위에서 간단히 언급된 바와 같이, 비디오 인코더 (20) 는 비디오 데이터를 인코딩한다. 비디오 데이터는 하나 이상의 픽처들을 포함할 수도 있다. 픽처들의 각각은 비디오의 부분을 형성하는 스틸 이미지이다. 일부 경우들에서, 픽처는 비디오 "프레임" 으로서 지칭될 수도 있다. 비디오 인코더 (20) 가 비디오 데이터를 인코딩할 때, 비디오 인코더 (20) 는 비트스트림을 생성할 수도 있다. 비트스트림은 비디오 데이터의 코딩된 표현을 형성하는 비트들의 시퀀스를 포함할 수도 있다. 비트스트림은 코딩된 픽처들 및 연관된 데이터를 포함할 수도 있다. 코딩된 픽처는 픽처의 코딩된 표현이다.
- [0040] 비트스트림을 생성하기 위해, 비디오 인코더 (20) 는 비디오 데이터에서의 각각의 픽처에 대한 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 픽처들 상에서 인코딩 동작들을 수행할 때, 비디오 인코더 (20) 는 일련의 코딩된 픽처들 및 연관된 데이터를 생성할 수도 있다. 연관된 데이터는 비디오 파라미터 세트들 (VPS), 시퀀스 파라미터 세트들 (SPS), 픽처 파라미터 세트들 (PPS), 적응 파라미터 세트들 (APS), 및 다른 선택 구조들을 포함할 수도 있다. SPS 는 0 이상의 시퀀스의 픽처들에 적용가능한 파라미터들을 포함할 수도 있다. PPS 는 0 이상의 픽처들에 적용가능한 파라미터들을 포함할 수도 있다. APS 는 0 이상의 픽처들에 적용가능한 파라미터들을 포함할 수도 있다. APS 에서의 파라미터들은 PPS 에서의 파라미터들 보다 변경하기 더 쉬운 파라미터들일 수도 있다.
- [0041] 코딩된 픽처들을 생성하기 위해서, 비디오 인코더 (20) 는 동일 사이즈의 비디오 블록들로 픽처를 파티셔닝할 수도 있다. 비디오 블록은 2 차원 어레이의 샘플들일 수도 있다. 비디오 블록들의 각각은 트리블록과 연관된다. 일부 경우들에서, 트리블록은 대형 코딩 유닛 (LCU) 으로 지칭될 수도 있다. HEVC 의 트리블록들은 H.264/AVC 과 같은, 이전 표준들의 매크로블록들과 대략 유사할 수도 있다. 하지만, 트리블록은 특정 사이즈에 반드시 제한되지 않으며 하나 이상의 코딩 유닛들 (CU) 을 포함할 수도 있다. 비디오 인코더 (20) 는 쿼드트리 파티셔닝을 사용하여 트리블록들의 비디오 블록들을 CU들과 연관된 비디오 블록들, 이로써 명칭 "트리블록들" 로 파티셔닝한다.
- [0042] 일부 예들에서, 비디오 인코더 (20) 는 픽처를 복수의 슬라이스들로 파티셔닝할 수도 있다. 슬라이스들의 각각은 정수의 CU들을 포함할 수도 있다. 일부 경우들에서, 슬라이스는 정수의 트리블록들을 포함한다. 다른 경우들에서, 슬라이스의 경계는 트리블록 내에 있을 수도 있다.
- [0043] 픽처 상에서 인코딩 동작을 수행하는 것의 부분으로서, 비디오 인코더 (20) 는 픽처의 각각의 슬라이스 상에서 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 슬라이스 상에서 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 슬라이스와 연관된 인코딩된 데이터를 생성할 수도 있다. 슬라이스와 연관된 인코딩된

데이터는 "코딩된 슬라이스" 로서 지칭될 수도 있다.

- [0044] 코딩된 슬라이스를 생성하기 위해서, 비디오 인코더 (20) 는 슬라이스에서의 각각의 트리블록 상에서 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 트리블록 상에서 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 코딩된 트리블록을 생성할 수도 있다. 코딩된 트리블록은 트리블록의 인코딩된 버전을 나타내는 데이터를 포함할 수도 있다.
- [0045] 비디오 인코더 (20) 가 코딩된 슬라이스를 생성할 때, 비디오 인코더 (20) 는 래스터 스캔 순서에 따라 슬라이스에서의 트리블록들 상에서 인코딩 동작들을 수행 (예를 들어, 인코딩) 할 수도 있다. 예를 들어, 비디오 인코더 (20) 는, 비디오 인코더 (20) 가 슬라이스에서의 트리블록들의 각각의 인코딩할 때까지, 슬라이스에서 트리블록들의 최상위 로우에 걸쳐 좌측에서 우측으로, 그 다음 트리블록들의 다음 하위 로우에 걸쳐 좌측에서 우측으로 진행되는 순서로 슬라이스의 트리블록들을 인코딩한다.
- [0046] 래스터 스캔 순서에 따라 트리블록들을 인코딩하는 것의 결과로서, 주어진 트리블록의 좌측으로 그리고 위의 트리블록들이 인코딩될 수도 있지만, 주어진 트리블록들의 우측으로 그리고 아래의 트리블록들은 아직 인코딩되지 않았다. 결과적으로, 비디오 인코더 (20) 는 주어진 트리블록을 인코딩할 때 주어진 트리블록의 좌측으로 그리고 위의 트리블록들을 인코딩하는 것에 의해 생성된 정보에 액세스하는 것이 가능할 수도 있다. 하지만, 비디오 인코더 (20) 는 주어진 트리블록을 인코딩할 때 주어진 트리블록의 우측으로 그리고 아래의 트리블록들을 인코딩하는 것에 의해 생성된 정보에 액세스하는 것이 가능하지 않을 수도 있다.
- [0047] 코딩된 트리블록을 생성하기 위해서, 비디오 인코더 (20) 는 트리블록의 비디오 블록 상에서 쿼드트리 파티셔닝을 회귀적으로 수행하여 비디오 블록을 계속해서 더 작은 비디오 블록들로 분할할 수도 있다. 작은 비디오 블록들의 각각은 상이한 CU 와 연관될 수도 있다. 예를 들어, 비디오 인코더 (20) 는 트리블록의 비디오 블록을 4 개의 동일한 사이즈의 서브블록들로 파티셔닝하고, 서브블록들의 하나 이상을 4 개의 동일한 사이즈의 서브 서브블록들로 파티셔닝하는 등등이다. 파티셔닝된 CU 는, 비디오 블록이 다른 CU들과 연관된 비디오 블록들로 파티셔닝되는 CU 일 수도 있다. 파티셔닝되지 않은 CU 는, 비디오 블록이 다른 CU들과 연관된 비디오 블록들로 파티셔닝되지 않은 CU 일 수도 있다.
- [0048] 비트스트림에서의 하나 이상의 신택스 엘리먼트들은, 비디오 인코더가 트리블록의 비디오 블록을 파티셔닝할 수 있는 최대 회수를 표시할 수도 있다. CU 의 비디오 블록은 형상이 정사각형일 수도 있다. CU 의 비디오 블록의 사이즈 (예를 들어, CU 의 사이즈) 는 8x8 픽셀들부터 최대 64x64 픽셀들 이상인 트리블록의 비디오 블록의 사이즈 (예를 들어, 트리블록의 사이즈) 까지의 범위일 수도 있다.
- [0049] 비디오 인코더 (20) 는 z-스캔 순서에 따라 트리블록의 각각의 CU 상에서 인코딩 동작들을 수행 (예를 들어, 인코딩) 할 수도 있다. 환언하면, 비디오 인코더 (20) 는 상부-좌측 CU, 상부-우측 CU, 하부-좌측 CU, 및 그 다음 하부-우측 CU 를, 그 순서로 인코딩할 수도 있다. 비디오 인코더 (20) 가 파티셔닝된 CU 상에서 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 z-스캔 순서에 따라 파티셔닝된 CU 의 비디오 블록의 서브블록들과 연관된 CU들을 인코딩할 수도 있다. 환언하면, 비디오 인코더 (20) 는 상부-좌측 서브블록과 연관된 CU, 상부-좌측 서브블록과 연관된 CU, 하부-좌측 서브블록과 연관된 CU, 및 그 다음 하부-우측 서브블록과 연관된 CU 를, 그 순서로 인코딩할 수도 있다.
- [0050] z-스캔 순서에 따라 트리블록의 CU들을 인코딩하는 것의 결과로서, 주어진 CU 의 위, 위 그리고 좌측으로, 위 그리고 우측으로, 좌측, 및 아래 그리고 우측으로의 CU들은 인코딩되었을 수도 있다. 주어진 CU 의 아래 그리고 우측으로의 CU들은 아직 인코딩되지 않았다. 결과적으로, 비디오 인코더 (20) 는 주어진 CU 를 인코딩할 때 주어진 CU 에 이웃하는 일부 CU들을 인코딩하는 것에 의해 생성된 정보에 액세스하는 것이 가능할 수도 있다. 하지만, 비디오 인코더 (20) 는 주어진 CU 를 인코딩할 때 주어진 CU 에 이웃하는 다른 CU들을 인코딩하는 것에 의해 생성된 정보에 액세스하는 것이 가능하지 않을 수도 있다.
- [0051] 비디오 인코더 (20) 가 파티셔닝되지 않은 CU 를 인코딩할 때, 비디오 인코더 (20) 는 CU 에 대해 하나 이상의 예측 유닛 (PU) 들을 생성할 수도 있다. CU 의 PU들의 각각은 CU 의 비디오 블록 내에서 상이한 비디오 블록과 연관될 수도 있다. 비디오 인코더 (20) 는 CU 의 각각의 PCU 에 대해 예측된 비디오 블록을 생성할 수도 있다. PU 의 예측된 비디오 블록은 샘플들의 블록일 수도 있다. 비디오 인코더 (20) 는 PU 에 대해 예측된 비디오 블록을 생성하기 위해 인트라 예측 또는 인터 예측을 사용할 수도 있다.
- [0052] 비디오 인코더 (20) 가 인트라 예측을 사용하여 PU 의 예측된 비디오 블록을 생성할 때, 비디오 인코더 (20) 는 PU 와 연관된 픽처의 디코딩된 샘플들에 기초하여 PU 의 예측된 비디오 블록을 생성할 수도 있다. 비디오

인코더 (20) 가 인트라 예측을 사용하여 CU 의 PU들의 예측된 비디오 블록들을 생성하는 경우, CU 는 인트라 예측된 CU 이다. 비디오 인코더 (20) 가 인터 예측을 사용하여 PU 의 예측된 비디오 블록을 생성하는 경우, 비디오 인코더 (20) 는 PU 와 연관된 픽처 이외에 하나 이상의 픽처들의 디코딩된 샘플들에 기초하여 PU 의 예측된 비디오 블록을 생성할 수도 있다. 비디오 인코더 (20) 가 인터 예측을 사용하여 CU 의 PU들의 예측된 비디오 블록들을 생성하는 경우, CU 는 인터 예측된 CU 이다.

[0053] 또한, 비디오 인코더 (20) 가 인터 예측을 사용하여 PU 에 대해 예측된 비디오 블록을 생성할 때, 비디오 인코더 (20) 는 PU 에 대한 모션 정보를 생성할 수도 있다. PU 에 대한 모션 정보는 PU 에 대한 하나 이상의 참조 블록들을 표시할 수도 있다. PU 의 각각의 참조블록은 참조 픽처내의 비디오 블록일 수도 있다. 참조 픽처는 PU 와 연관된 픽처 이외의 픽처일 수도 있다. 일부 경우들에서, PU 의 참조 블록은 PU 의 "참조 샘플" 로서 또한 지칭될 수도 있다. 비디오 인코더 (20) 는 PU 의 참조 블록들에 기초하여 PU 에 대해 예측된 비디오 블록을 생성할 수도 있다.

[0054] 비디오 인코더 (20) 가 CU 의 하나 이상의 PU들에 대해 예측된 비디오 블록들을 생성한 후, 비디오 인코더 (20) 는 CU 의 PU들에 대해 예측된 비디오 블록들에 기초하여 CU 에 대한 잔차 데이터를 생성할 수도 있다. CU 에 대한 잔차 데이터는 CU 의 원래 비디오 블록과 CU 의 PU들에 대해 예측된 비디오 블록들에서의 샘플들 사이의 차이를 표시할 수도 있다.

[0055] 또한, 파티셔닝되지 않은 CU 상에서 인코딩 동작을 수행하는 부분으로서, 비디오 인코더 (20) 는 CU 의 잔차 데이터 상에서 회귀적 쿼드트리 파티셔닝을 수행하여 CU 의 잔차 데이터를 CU 의 변환 유닛 (TU) 들과 연관된 잔차 데이터의 하나 이상의 블록들 (예를 들어, 잔차 비디오 블록들) 로 파티셔닝할 수도 있다. CU 의 각각의 TU 는 상이한 잔차 비디오 블록과 연관될 수도 있다.

[0056] 비디오 인코더 (20) 는 TU들과 연관된 잔차 비디오 블록들에 하나 이상의 변환들을 적용하여 TU들과 연관된 변환 계수 블록들 (예를 들어, 변환 계수들의 블록들) 을 생성할 수도 있다. 개념적으로, 변환 계수 블록은 변환 계수들의 2 차원 (2D) 행렬일 수도 있다.

[0057] 변환 계수 블록을 생성한 후, 비디오 인코더 (20) 는 변환 계수 블록 상에서 양자화 프로세스를 수행할 수도 있다. 양자화는 일반적으로, 변환 계수들이 양자화되어 가능하게는 변환 계수들을 나타내기 위해 사용되는 데이터의 양을 감소시켜, 추가적인 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스는 변환 계수들의 일부 또는 전부와 연관된 비트 심도를 감소시킬 수도 있다. 예를 들어, n-비트 변환 계수는 양자화 동안 m-비트 변환 계수로 라운드 다운될 수도 있으며, 여기서 n 은 m 보다 크다.

[0058] 비디오 인코더 (20) 는 각각의 CU 를 양자화 파라미터 (QP) 값과 연관시킬 수도 있다. CU 와 연관된 QP 값은, 비디오 인코더 (20) 가 CU 와 연관된 변환 계수 블록들을 어떻게 양자화하는지를 결정할 수도 있다. 비디오 인코더 (20) 는 CU 와 연관된 QP 값을 조정하는 것에 의해 CU 와 연관된 변환 계수블록들에 적용된 양자화 정도를 조정할 수도 있다.

[0059] 비디오 인코더 (20) 가 변환 계수 블록을 양자화한 후, 비디오 인코더 (20) 는 양자화된 변환 계수 블록에서의 변환 계수들을 나타내는 신택스 엘리먼트들의 세트들을 생성할 수도 있다. 비디오 인코더 (20) 는 엔트로피 인코딩 동작들, 예컨대 콘텍스트 적응형 이진 산술 코딩 (CABAC) 동작들을, 이들 신택스 엘리먼트들의 일부에 적용할 수도 있다. 다른 엔트로피 코딩 기법들, 예컨대 콘텍스트 적응형 가변 길이 코딩 (CAVLC), 확률 인터벌 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 이진 산술 코딩이 또한 사용될 수 있다.

[0060] 비디오 인코더 (20) 에 의해 생성된 비트스트림은 일련의 네트워크 추상화 계층 (NAL) 유닛들을 포함할 수도 있다. NAL 유닛들의 각각은 데이터를 포함하는 바이트들 및 NAL 유닛에서의 데이터의 타입의 표시를 포함하는 신택스 구조일 수도 있다. 예를 들어, NAL 유닛은 비디오 파라미터 세트, 시퀀스 파라미터 세트, 픽처 파라미터 세트, 코딩된 슬라이스, SEI, 액세스 유닛 딜리미터, 필터 데이터, 또는 다른 타입의 데이터를 나타내는 데이터를 포함할 수도 있다. NAL 유닛에서의 데이터는 다양한 신택스 구조를 포함할 수도 있다.

[0061] 비디오 디코더 (30) 는 비디오 인코더 (20) 에 의해 생성된 비트스트림을 수신할 수도 있다. 비트스트림은 비디오 인코더 (20) 에 의해 인코딩된 비디오 데이터의 코딩된 표현을 포함할 수도 있다. 비디오 디코더 (30) 가 비트스트림을 수신할 때, 비디오 디코더 (30) 는 비트스트림 상에서 파싱 동작을 수행할 수도 있다. 비디오 디코더 (30) 가 파싱 동작을 수행할 때, 비디오 디코더 (30) 는 비트스트림으로부터 신택스 엘리먼트들을 추출할 수도 있다. 비디오 디코더 (30) 가 비트스트림으로부터 추출된 신택스 엘리먼트에 기초하여 비디오 데이터의 픽처들을 재구성할 수도 있다. 신택스 엘리먼트들에 기초하여 비디오 데이터를 재구성하기

위한 프로세스는 일반적으로 선택스 엘리먼트들을 생성하기 위해 비디오 인코더 (20) 에 의해 수행된 프로세스에 상반될 수도 있다.

[0062] 비디오 디코더 (30) 가 CU 와 연관된 선택스 엘리먼트들을 추출한 후, 비디오 디코더 (30) 는 선택스 엘리먼트들에 기초하여 CU 의 PU들에 대한 예측된 비디오 블록들을 생성할 수도 있다. 부가적으로, 비디오 디코더 (30) 는 CU 의 TU들과 연관된 변환 계수 블록들을 역 양자화할 수도 있다. 비디오 디코더 (30) 는 CU 의 TU 와 연관된 잔차 비디오 블록들을 재구성하기 위해 변환 계수 블록들 상에서 역 변환을 수행할 수도 있다. 예측된 비디오 블록을 생성하고 잔차 비디오 블록들을 재구성한 후, 비디오 디코더 (30) 는 예측된 비디오 블록들 및 잔차 비디오 블록들에 기초하여 CU 의 비디오 블록을 재구성할 수도 있다. 이러한 방식으로, 비디오 디코더 (30) 는 비트스트림에서 선택스 엘리먼트에 기초하여 CU들의 비디오 블록들을 재구성할 수도 있다.

[0063] 비디오 인코더

[0064] 도 2 는 이 개시물에 기재된 양태들에 따른 기법들을 구현할 수도 있는 비디오 인코더 (20) 의 일 예를 도시하는 블록 다이어그램이다. 비디오 인코더 (20) 는 예컨대 HEVC 에 대해, 비디오 프레임의 단일 계층을 프로세싱하도록 구성될 수도 있다. 또한, 비디오 인코더 (20) 는 이 개시물의 기법들 중 임의의 것 또는 모두를 수행하도록 구성될 수도 있다. 일부 예들에서, 이 개시물에 기재된 기법들은 비디오 인코더 (20) 의 다양한 컴포넌트들 사이에서 공유될 수도 있다. 일부 예들에서, 부가적으로 또는 대안으로, 프로세서 (미도시) 는 이 개시물에 기재된 기법들 중 임의의 것 또는 모두를 수행하도록 구성될 수도 있다.

[0065] 설명을 위해, 이 개시물은 HEVC 코딩의 컨텍스트에서 비디오 인코더 (20) 를 기재한다. 하지만, 이 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다. 도 2 에 도시된 예는 단일 계층 코덱을 위한 것이다. 하지만, 소정의 실시형태들에서, 비디오 인코더 (20) 의 일부 또는 전부는 멀티 계층 코덱의 프로세싱을 위해 카피될 수도 있다.

[0066] 비디오 인코더 (20) 는 비디오 슬라이스들 내에서 비디오 블록들의 인트라 및 인터 코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 또는 픽처 내의 비디오에서 공간적 리던던시를 감소 또는 제거하기 위해 공간적 예측에 의존한다. 인터 코딩은 비디오 시퀀스의 인접한 프레임들 또는 픽처들 내의 비디오에서 시간적 리던던시를 감소 또는 제거하기 위해 시간적 예측에 의존한다. 인트라 모드 (I 모드) 는 임의의 몇몇 공간 기반 코딩 모드들을 지칭할 수도 있다. 인터 모드들, 예컨대 일방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 는 임의의 몇몇 시간 기반 코딩 모드들을 지칭할 수도 있다.

[0067] 도 2 의 예에서, 비디오 인코더 (20) 는 복수의 기능 컴포넌트들을 포함한다. 비디오 인코더 (20) 의 기능 컴포넌트들은, 예측 프로세싱 유닛 (100), 잔차 생성 유닛 (102), 변환 프로세싱 유닛 (104), 양자화 유닛 (106), 역 양자화 유닛 (108), 역 변환 유닛 (110), 재구성 유닛 (112), 필터 유닛 (113), 디코딩된 픽처 버퍼 (114), 및 엔트로피 코딩 유닛 (116) 을 포함한다. 예측 프로세싱 유닛 (100) 은 인트라 예측 유닛 (121), 모션 추정 유닛 (122), 모션 보상 유닛 (124), 인트라 예측 유닛 (126), 및 계층간 예측 유닛 (128) 을 포함한다. 다른 예들에서, 비디오 인코더 (20) 는 더 많거나, 더 적거나, 상이한 기능 컴포넌트들을 포함할 수도 있다. 또한, 모션 추정 유닛 (122) 및 모션 보상 유닛 (124) 은 고도로 통합될 수도 있지만, 예시의 목적을 위해 도 2 의 예에서는 별도로 나타낸다.

[0068] 비디오 인코더 (20) 는 비디오 데이터를 수신할 수도 있다. 비디오 인코더 (20) 는 다양한 소스들로부터 비디오 데이터를 수신할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 비디오 소스 (18)(예를 들어, 도 1a 또는 도 1b 에 나타냄) 또는 다른 소스로부터 비디오 데이터를 수신할 수도 있다. 비디오 데이터는 일련의 픽처들을 나타낼 수도 있다. 비디오 데이터를 인코딩하기 위해서, 비디오 인코더 (20) 는 픽처의 각각의 상에서 인코딩 동작을 수행할 수도 있다. 픽처 상에서 인코딩 동작을 수행하는 것의 부분으로서, 비디오 인코더 (20) 는 픽처의 각각의 슬라이스 상에서 인코딩 동작들을 수행할 수도 있다. 슬라이스 상에서 인코딩 동작을 수행하는 것의 부분으로서, 비디오 인코더 (20) 는 슬라이스에서의 트리블록들 상에서 인코딩 동작들을 수행할 수도 있다.

[0069] 트리블록 상에서 인코딩 동작을 수행하는 것의 부분으로서, 예측 프로세싱 유닛 (100) 은 트리블록의 비디오 블록 상에서 쿼드트리 파티셔닝을 수행하여 계속해서 더 작은 비디오 블록들로 비디오 블록을 분할할 수도 있다. 작은 비디오 블록들의 각각은 상이한 CU 와 연관될 수도 있다. 예를 들어, 예측 프로세싱 유닛 (100) 은 트리블록의 비디오 블록을 4 개의 동일한 사이즈의 서브블록들로 파티셔닝하고, 서브블록들의 하나 이상을 4 개의 동일한 사이즈의 서브 서브블록들로 파티셔닝하는 등등일 수도 있다.

- [0070] CU들과 연관된 비디오 블록들의 사이즈는 8x8 샘플들부터 최대 64x64 샘플들 이상인 트리블록의 사이즈까지의 범위일 수도 있다. 이 개시물에서, “NxN” 및 “N 바이 N” 은 수직 및 수평 치수들에 관한 비디오 블록의 샘플 치수들, 예를 들어 16x16 샘플들 또는 16 바이 16 샘플들을 지칭하기 위해 상호교환가능하게 사용될 수도 있다. 일반적으로, 16x16 비디오 블록은 수직 방향에서의 16 개의 샘플들 ($y = 16$) 및 수평 방향에서의 16 개의 샘플들 ($x = 16$) 을 갖는다. 마찬가지로, NxN 블록은 일반적으로 수직 방향에서의 N 개의 샘플들 및 수평 방향에서의 N 개의 샘플들을 가지며, 여기서 N 은 음이 아닌 정수 값을 나타낸다.
- [0071] 또한, 트리블록 상에서 인코딩 동작을 수행하는 것의 부분으로서, 예측 프로세싱 유닛 (100) 은 트리블록에 대한 계위적 쿼드트리 데이터 구조를 생성할 수도 있다. 예를 들어, 트리블록은 쿼드트리 데이터 구조의 루트 노드에 대응할 수도 있다. 예측 프로세싱 유닛 (100) 이 트리블록의 비디오 블록을 4 개의 서브블록들로 파티셔닝하는 경우, 루트 노드는 쿼드트리 데이터 구조에서 4 개의 자식 노드들을 갖는다. 자식 노드들의 각각은 서브블록들 중 하나와 연관된 CU 에 대응한다. 예측 프로세싱 유닛 (100) 이 서브블록들 중 하나를 4 개의 서브블록들로 파티셔닝하는 경우, 서브블록과 연관된 CU 에 대응하는 노드는 4 개의 자식 노드들을 가질 수도 있으며, 그 각각은 서브 서브블록들 중 하나와 연관된 CU 에 대응한다.
- [0072] 쿼드트리 데이터 구조의 각각의 노드는 대응 트리블록 또는 CU 에 대해 선택스 데이터 (예를 들어, 선택스 엘리먼트) 를 포함할 수도 있다. 예를 들어, 쿼드트리에서의 노드는, 노드에 대응하는 CU 의 비디오 블록이 4 개의 서브블록들로 파티셔닝 (스플릿) 되는지 여부를 표시하는 스플릿 플래그를 포함할 수도 있다. CU 에 대한 선택스 엘리먼트들은 회귀적으로 정의될 수도 있고, CU 의 비디오 블록이 서브블록들로 스플릿되는지 여부에 의존할 수도 있다. 비디오 블록이 파티셔닝되지 않는 CU 는 쿼드트리 데이터 구조에서 리프 (leaf) 노드에 대응할 수도 있다. 코딩된 트리블록은 대응 트리블록에 대해 쿼드트리 데이터 구조에 기초한 데이터를 포함할 수도 있다.
- [0073] 비디오 인코더 (20) 는 트리블록의 각각의 파티셔닝되지 않은 CU 상에서 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 파티셔닝되지 않은 CU 상에서 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 파티셔닝되지 않은 CU 의 인코딩된 표현을 나타내는 데이터를 생성한다.
- [0074] CU 상에서 인코딩 동작을 수행하는 것의 부분으로서, 예측 프로세싱 유닛 (100) 은 CU 의 하나 이상의 PU들 중에서 CU 의 비디오 블록을 파티셔닝할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 다양한 PU 사이즈들을 지원할 수도 있다. 특정 CU 의 사이즈가 $2N_x2N_y$ 인 것을 상정하면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 $2N_x2N_y$ 또는 N_xN_y 의 PU 사이즈들, 및 $2N_x2N_y$, $2N_xN_y$, N_x2N_y , N_xN_y , $2N_xnU$, nL_x2N_y , nR_x2N_y 등의 대칭 PU 사이즈들에서의 인터 예측을 지원할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 또한 $2N_xnU$, $2N_xnD$, nL_x2N_y , 및 nR_x2N_y 의 PU 사이즈들에 대해 비대칭 파티셔닝을 지원할 수도 있다. 일부 예들에서, 예측 프로세싱 유닛 (100) 은 우측 각도들에서 CU 의 비디오 블록의 측면들과 만나지 않는 경계를 따라 CU 의 PU들 중에서 CU 의 비디오 블록을 파티셔닝하기 위해 기하학적 파티셔닝을 수행할 수도 있다.
- [0075] 인터 예측 유닛 (121) 은 CU 의 각각의 PU 상에서 인터 예측을 수행할 수도 있다. 인터 예측은 시간적 압축을 제공할 수도 있다. PU 상에서 인터 예측을 수행하기 위해, 모션 추정 유닛 (122) 은 PU 에 대해 모션 정보를 생성할 수도 있다. 모션 보상 유닛 (124) 은 CU 와 연관된 픽처 (예를 들어, 참조 픽처들) 이외의 픽처들의 디코딩된 샘플들 및 모션 정보에 기초하여 PU 에 대해 예측된 비디오 블록을 생성할 수도 있다. 이 개시물에 있어서, 모션 보상 유닛 (124) 에 의해 생성된 예측된 비디오 블록은 인터 예측된 비디오 블록으로서 지칭될 수도 있다.
- [0076] 슬라이스들은 I 슬라이스들, P 슬라이스들, 또는 B 슬라이스들일 수도 있다. 모션 추정 유닛 (122) 및 모션 보상 유닛 (124) 은, PU 가 I 슬라이스, P 슬라이스, 또는 B 슬라이스인지 여부에 따라 CU 의 PU 에 대한 상이한 동작들을 수행할 수도 있다. I 슬라이스에서, 모든 PU들은 인트라 예측된다. 따라서, PU 가 I 슬라이스 내에 있으면, 모션 추정 유닛 (122) 및 모션 보상 유닛 (124) 은 PU 상에서 인터 예측을 수행하지 않는다.
- [0077] PU 가 P 슬라이스 내에 있으면, PU 를 포함하는 픽처는 "리스트 0" 으로서 지칭된 참조 픽처들의 리스트와 연관된다. 리스트 0 내의 참조 픽처들 각각은 다른 픽처들의 인터 예측을 위해 사용될 수도 있는 샘플들을 포함한다. 모션 추정 유닛 (122) 이 P 슬라이스에서 PU 에 관하여 모션 추정 동작을 수행하는 경우, 모션 추정 유닛 (122) 은 PU 에 대한 참조 블록에 대해 리스트 0 에서 참조 픽처들을 검색할 수도 있다. PU 의 참조 블록은, PU 의 비디오 블록에서의 샘플들에 매우 밀접하게 대응하는 샘플들의 세트, 예를 들어 샘플들의 블록들일 수도 있다. 모션 추정 유닛 (122) 은, 참조 픽처에서의 샘플들의 세트가 PU 의 비디오 블록에서의 샘플들에 어떻게 밀접하게 대응하는지를 결정하기 위해 다양한 행렬들을 사용할 수도 있다. 예를 들어, 모션 추

정 유닛 (122) 은, 참조 픽처에서의 샘플들의 세트가 절대 차의 합 (sum of absolute difference; SAD), 제곱 차의 합 (sum of square difference; SSD), 또는 다른 차이 메트릭들에 의해 PU 의 비디오 블록에서의 샘플들에 어떻게 밀접하게 대응하는지를 결정할 수도 있다.

[0078] P 슬라이스에서 PU 의 참조 블록을 식별한 후에, 모션 추정 유닛 (122) 은 PU 와 참조 블록 간의 공간 변위를 나타내는 모션 벡터 및 참조 블록을 포함하는 리스트 0 에서의 참조 픽처를 나타내는 참조 인덱스를 생성할 수도 있다. 다양한 예들에서, 모션 추정 유닛 (122) 은 변하는 정확도의 정도에 대한 모션 벡터들을 생성할 수도 있다. 예를 들어, 모션 추정 유닛 (122) 은 1/4 샘플 정확도, 1/8 샘플 정확도, 또는 다른 분수적 샘플 정확도에서 모션 벡터들을 생성할 수도 있다. 분수적 샘플 정확도의 경우에서, 참조 블록 값들은 참조 픽처에서 정수-포지션 샘플 값들로부터 보간될 수도 있다. 모션 추정 유닛 (122) 은 PU 의 모션 정보로서 모션 벡터 및 참조 인덱스를 출력할 수도 있다. 모션 보상 유닛 (124) 은 PU 의 모션 정보에 의해 식별된 참조 블록에 기초하여 PU 의 예측된 비디오 블록을 생성할 수도 있다.

[0079] PU 가 B 슬라이스 내에 있으면, PU 를 포함하는 픽처는 "리스트 0" 및 "리스트 1" 로서 지칭된 참조 픽처들의 2 개의 리스트들과 연관될 수도 있다. 일부 예들에서, B 슬라이스를 포함하는 픽처는 리스트 0 및 리스트 1 의 조합인 리스트 조합과 연관될 수도 있다.

[0080] 또한, PU 가 B 슬라이스 내에 있으면, 모션 추정 유닛 (122) 은 PU 에 대한 단-방향 예측 또는 양-방향 예측을 수행할 수도 있다. 모션 추정 유닛 (122) 이 PU 에 대한 단-방향 예측을 수행한 경우, 모션 추정 유닛 (122) 은 PU 에 대한 참조 블록에 대해 리스트 0 또는 리스트 1 의 참조 픽처들을 검색할 수도 있다. 모션 추정 유닛 (122) 은 그 후, PU 와 참조 블록 간의 공간적 변위를 나타내는 모션 벡터 및 참조 블록을 포함하는 리스트 0 또는 리스트 1 에서의 참조 픽처를 나타내는 참조 인덱스를 생성할 수도 있다. 모션 추정 유닛 (122) 은 PU 의 모션 정보로서 참조 인덱스, 예측 방향 표시자, 및 모션 벡터를 출력할 수도 있다. 예측 방향 표시자는, 참조 인덱스가 리스트 0 또는 리스트 1 에서 참조 픽처를 나타내는지 여부를 나타낼 수도 있다. 모션 보상 유닛 (124) 은 PU 의 모션 정보에 의해 표시된 참조 블록에 기초하여 PU 의 예측된 비디오 블록을 생성할 수도 있다.

[0081] 모션 추정 유닛 (122) 이 PU 에 대한 양-방향 예측을 수행하는 경우, 모션 추정 유닛 (122) 은 PU 에 대한 참조 블록에 대해 리스트 0 에서의 참조 픽처들을 검색할 수도 있고, 또한 PU 에 대한 다른 참조 블록에 대해 리스트 1 에서 참조 픽처들을 검색할 수도 있다. 모션 추정 유닛 (122) 은 그 후, 참조 블록들과 PU 간의 공간적 변위를 나타내는 모션 벡터들 및 참조 블록들을 포함하는 리스트 0 및 리스트 1 에서의 참조 픽처들을 나타내는 참조 인덱스들을 생성할 수도 있다. 모션 추정 유닛 (122) 은 PU 의 모션 정보로서 PU 의 모션 벡터들 및 참조 인덱스들을 출력할 수도 있다. 모션 보상 유닛 (124) 은 PU 의 모션 정보에 의해 표시된 참조 블록에 기초하여 PU 의 예측된 비디오 블록을 생성할 수도 있다.

[0082] 일부 경우들에서, 모션 추정 유닛 (122) 은 PU 에 대한 모션 정보의 풀 세트를 엔트로피 인코딩 유닛 (116) 에 출력하지 않는다. 오히려, 모션 추정 유닛 (122) 은 다른 PU 의 모션 정보를 참조하여 PU 의 모션 정보를 시그널링할 수도 있다. 예를 들어, 모션 추정 유닛 (122) 은, PU 의 모션 정보가 이웃하는 PU 의 모션 정보에 충분히 유사하다는 것을 결정할 수도 있다. 이 예에서, 모션 추정 유닛 (122) 은 PU 와 연관된 신덱스 구조에서, PU 가 이웃하는 PU 와 동일한 모션 정보를 갖는다는 것을 나타내는 값을 비디오 디코더 (30) 에 나타낼 수도 있다. 다른 예에서, 모션 추정 유닛 (122) 은 PU 와 연관된 신덱스 구조에서, 이웃하는 PU 및 모션 벡터 차이 (MVD) 를 식별할 수도 있다. 모션 벡터 차이는 표시된 이웃하는 PU 의 모션 벡터와 PU 의 모션 벡터 간의 차이를 나타낸다. 비디오 디코더 (30) 는 표시된 이웃하는 PU 의 모션 벡터 및 모션 벡터 차이를 사용하여, PU 의 모션 벡터를 결정할 수도 있다. 제 2 PU 의 모션 정보를 시그널링할 때 제 1 PU 의 모션 정보를 참조함으로써, 비디오 인코더 (20) 는 더 적은 비트들을 사용하여 제 2 PU 의 모션 정보를 시그널링할 수도 있다.

[0083] CU 상에서 인코딩 동작을 수행하는 것의 부분으로서, 인트라 예측 유닛 (126) 은 CU 의 PU들 상에서 인트라 예측을 수행할 수도 있다. 인트라 예측은 공간적 압축을 제공할 수도 있다. 인트라 예측 유닛 (126) 이 PU 상에서 인트라 예측을 수행하는 경우, 인트라 예측 유닛 (126) 은 동일한 픽처에서 다른 PU들의 디코딩된 샘플들에 기초하여 PU 에 대한 예측 데이터를 생성할 수도 있다. PU 에 대한 예측 데이터는 예측된 비디오 블록 및 다양한 신덱스 엘리먼트들을 포함할 수도 있다. 인트라 예측 유닛 (126) 은 I 슬라이스들, P 슬라이스들, 및 B 슬라이스들에서의 PU들 상에서 인트라 예측을 수행할 수도 있다.

[0084] PU 상에서 인트라 예측을 수행하기 위해, 인트라 예측 유닛 (126) 은 다수의 인트라 예측 모드들을 사용하여 PU

에 대한 예측 데이터의 다수의 세트들을 생성할 수도 있다. 인트라 예측 유닛 (126) 이 인트라 예측을 사용하여 PU 에 대한 예측 데이터의 세트를 생성하는 경우, 인트라 예측 유닛 (126) 은 인트라 예측 모드와 연관된 방향 및/또는 기울기 (gradient) 에서 PU 의 비디오 블록을 가로지르는 이웃하는 PU들의 비디오 블록들로부터 샘플들을 확장할 수도 있다. PU들, CU들, 및 트리블록들에 대한 좌측에서 우측으로, 상부에서 하부로의 인코딩 순서를 가정하면, 이웃하는 PU들은 PU 의 위, 위 및 우측, 위 및 좌측, 또는 좌측에 있을 수도 있다. 인트라 예측 유닛 (126) 은 PU 의 사이즈에 따라, 다양한 수의 인트라 예측 모드들, 예를 들어 33 개의 방향 인트라 예측 모드들을 사용할 수도 있다.

[0085] 예측 프로세싱 유닛 (100) 은 PU 에 대한 모션 보상 유닛 (124) 에 의해 생성된 예측 데이터 또는 PU 에 대한 인트라 예측 유닛 (126) 에 의해 생성된 예측 데이터 중에서부터 PU 에 대한 예측 데이터를 선택할 수도 있다. 일부 예들에서, 예측 프로세싱 유닛 (100) 은 예측 데이터의 세트들의 레이트/왜곡 메트릭들에 기초하여 PU 에 대한 예측 데이터를 선택한다.

[0086] 예측 프로세싱 유닛 (100) 이 인트라 예측 유닛 (126) 에 의해 생성된 예측 데이터를 선택하면, 예측 프로세싱 유닛 (100) 은 PU 들에 대한 예측 데이터를 생성하기 위해 사용되었던 인트라 예측 모드, 예를 들어 선택된 인트라 예측 모드를 시그널링할 수도 있다. 예측 프로세싱 유닛 (100) 은 선택된 인트라 예측 모드를 다양한 방식들로 시그널링할 수도 있다. 예를 들어, 선택된 인트라 예측 모드는 이웃하는 PU 의 인트라 예측 모드와 동일할 것 같다. 다시 말해, 이웃하는 PU 의 인트라 예측 모드는 현재 PU 에 대한 최고 확률의 모드일 수도 있다. 따라서, 예측 프로세싱 유닛 (100) 은, 선택된 인트라 예측 모드가 이웃하는 PU 의 인트라 예측 모드와 동일하다는 것을 나타내도록 신택스 엘리먼트를 생성할 수도 있다.

[0087] 위에서 논의된 바와 같이, 비디오 인코더 (20) 는 계층간 예측 유닛 (128) 을 포함할 수도 있다. 계층간 예측 유닛 (128) 은 SVC 에서 이용 가능한 하나 이상의 상이한 계층들 (예를 들어, 베이스 또는 참조 계층) 을 사용하여 현재 블록 (예를 들어, EL 에서의 현재 블록) 을 예측하도록 구성된다. 이러한 예측은 계층간 예측으로서 지칭될 수도 있다. 계층간 예측 유닛 (128) 은 예측 방법들을 이용하여 계층간 리던던시를 감소시키고, 이에 의해 코딩 효율성을 개선하고 연산적 리소스 요건들을 감소시킨다. 계층간 예측의 일부 예들은 계층간 인트라 예측, 계층간 모션 예측, 및 계층간 잔차 예측을 포함할 수도 있다. 계층간 인트라 예측은 강화 계층에서 현재 블록을 예측하도록 베이스 계층에서 공통-위치된 블록들의 재구성성을 사용한다. 계층간 모션 예측은 강화 계층에서 모션을 예측하도록 베이스 계층의 모션 정보를 사용한다. 계층간 잔차 예측은 강화 계층의 잔차를 예측하도록 베이스 계층의 잔차를 사용한다.

[0088] 예측 프로세싱 유닛 (100) 이 CU 의 PU들에 대한 예측 데이터를 선택한 후, 잔차 생성 유닛 (102) 은 CU 의 비디오 블록으로부터 CU 의 PU들의 예측된 비디오 블록들을 감산함으로써 (예를 들어, 마이너스 부호로 표시됨) CU 에 대한 잔차 데이터를 생성할 수도 있다. CU 의 잔차 데이터는 CU 의 비디오 블록에서 샘플들의 상이한 샘플 컴포넌트들에 대응하는 2D 잔차 비디오 블록들을 포함할 수도 있다. 예를 들어, 잔차 데이터는 CU 의 PU들의 예측된 비디오 블록들에서의 샘플들의 루미넌스 컴포넌트들과 CU 의 원래의 비디오 블록에서의 샘플들의 루미넌스 컴포넌트들 간의 차이들에 대응하는 잔차 비디오 블록을 포함할 수도 있다. 또한, CU 의 잔차 데이터는 CU 의 PU들의 예측된 비디오 블록들에서의 샘플들의 크로미넌스 컴포넌트들과 CU 의 원래의 비디오 블록에서의 샘플들의 크로미넌스 컴포넌트들 간의 차이들에 대응하는 잔차 비디오 블록들을 포함할 수도 있다.

[0089] 예측 프로세싱 유닛 (100) 은 CU 의 잔차 비디오 블록들을 서브-블록들로 파티셔닝하도록 쿼드트리 파티셔닝을 수행할 수도 있다. 각각의 분할되지 않은 잔차 비디오 블록은 CU 의 상이한 TU 와 연관될 수도 있다. CU 의 TU들과 연관된 잔차 비디오 블록들의 사이즈들 및 포지션들은 CU 의 PU들과 연관된 비디오 블록들의 사이즈들 및 포지션들에 기초하거나 기초하지 않을 수도 있다. "잔차 쿼드트리 (RQT)" 로서 알려진 쿼드트리 구조는 잔차 비디오 블록들 각각과 연관된 노드들을 포함할 수도 있다. CU 의 TU들은 RQT 의 리프 노드들에 대응할 수도 있다.

[0090] 변환 프로세싱 유닛 (104) 은 TU 와 연관된 잔차 비디오 블록에 하나 이상의 변환들을 적용함으로써 CU 의 각각의 TU 에 대한 하나 이상의 변환 계수 블록들을 생성할 수도 있다. 변환 계수 블록들 각각은 변환 계수들의 2D 행렬일 수도 있다. 변환 프로세싱 유닛 (104) 은 TU 와 연관된 잔차 비디오 블록에 다양한 변환들을 적용할 수도 있다. 예를 들어, 변환 프로세싱 유닛 (104) 은 이산 코사인 변환 (DCT), 방향 변환, 또는 개념적으로 유사한 변환을 TU 와 연관된 잔차 비디오 블록에 적용할 수도 있다.

[0091] 변환 프로세싱 유닛 (104) 이 TU 와 연관된 변환 계수 블록을 생성한 후에, 양자화 유닛 (106) 은 변환 계수 블록에서 변환 계수들을 양자화할 수도 있다. 양자화 유닛 (106) 은 CU 와 연관된 QP 값에 기초하여 CU 의 TU

와 연관된 변환 계수 블록을 양자화할 수도 있다.

- [0092] 비디오 인코더 (20) 는 다양한 방식들에서 CU 와 QP 값을 연관시킬 수도 있다. 예를 들어, 비디오 인코더 (20) 는 CU 와 연관된 트리블록 상에서 레이트-왜곡 분석을 수행할 수도 있다. 레이트-왜곡 분석에서, 비디오 인코더 (20) 는 트리블록 상에서 다수 회 인코딩 동작을 수행함으로써 트리블록의 다중 코딩된 표현들을 생성할 수도 있다. 비디오 인코더 (20) 는, 비디오 인코더 (20) 가 트리블록의 상이한 인코딩된 표현들을 생성하는 경우 CU 와 상이한 QP 값들을 연관시킬 수도 있다. 비디오 인코더 (20) 는, 소정의 QP 값이 최하위 비트레이트 및 왜곡 메트릭을 갖는 트리블록의 코딩된 표현으로 CU 와 연관되는 경우 그 소정의 QP 값이 CU 와 연관된다는 것을 시그널링할 수도 있다.
- [0093] 역양자화 유닛 (108) 및 역변환 유닛 (110) 은 변환 계수 블록에 역양자화 및 역변환들을 각각 적용하여, 변환 계수 블록으로부터 잔차 비디오 블록을 재구성할 수도 있다. 재구성 유닛 (112) 은 예측 프로세싱 유닛 (100) 에 의해 생성된 하나 이상의 예측된 비디오 블록들로부터 대응하는 샘플들에 재구성된 잔차 비디오 블록을 추가하여, TU 와 연관된 재구성된 비디오 블록을 생성할 수도 있다. 이 방식에서 CU 의 각각의 TU 에 대한 비디오 블록들을 재구성함으로써, 비디오 인코더 (20) 는 CU 의 비디오 블록을 재구성할 수도 있다.
- [0094] 재구성 유닛 (112) 이 CU 의 비디오 블록을 재구성한 후에, 필터 유닛 (113) 은 CU 와 연관된 비디오 블록에서 블로킹 아티팩트들을 감소시키기 위해 디블로킹 동작을 수행할 수도 있다. 하나 이상의 디블로킹 동작들을 수행한 후에, 필터 유닛 (113) 은 디코딩된 픽처 버퍼 (114) 에 CU 의 재구성된 비디오 블록을 저장할 수도 있다. 모션 추정 유닛 (122) 및 모션 보상 유닛 (124) 은 재구성된 비디오 블록을 포함하는 참조 픽처를 사용하여 후속의 픽처들의 PU들 상에서 인터 예측을 수행할 수도 있다. 또한, 인트라 예측 유닛 (126) 은 디코딩된 픽처 버퍼 (114) 에서 재구성된 비디오 블록들을 사용하여 CU 와 동일한 픽처에서의 다른 PU들 상에서 인트라 예측을 수행할 수도 있다.
- [0095] 엔트로피 인코딩 유닛 (116) 은 비디오 인코더 (20) 의 다른 기능적 컴포넌트들로부터 데이터를 수신할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (116) 은 양자화 유닛 (106) 으로부터 변환 계수 블록들을 수신할 수도 있고, 예측 프로세싱 유닛 (100) 으로부터 선택스 엘리먼트들을 수신할 수도 있다. 엔트로피 인코딩 유닛 (116) 이 데이터를 수신하는 경우, 엔트로피 인코딩 유닛 (116) 은 하나 이상의 엔트로피 인코딩 동작들을 수행하여 엔트로피 인코딩된 데이터를 생성할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 CAVLC 동작, CABAC 동작, 변수-대-변수 (V2V) 길이 코딩 동작, 선택스 기반 컨텍스트 적응적 이진 산술 코딩 (SBAC) 동작, 확률 구간 파티셔닝 엔트로피 (PIPE) 코딩 동작, 또는 다른 타입의 엔트로피 인코딩 동작을 데이터 상에 수행할 수도 있다. 엔트로피 인코딩 유닛 (116) 은 엔트로피 인코딩된 데이터를 포함하는 비트스트림을 출력할 수도 있다.
- [0096] 데이터 상에서 엔트로피 인코딩 동작을 수행하는 것의 부분으로서, 엔트로피 인코딩 유닛 (116) 은 컨텍스트 모델을 선택할 수도 있다. 엔트로피 인코딩 유닛 (116) 이 CABAC 동작을 수행하고 있다면, 컨텍스트 모델은 특정 값들을 갖는 특정 bin들의 확률들의 추정치들을 나타낼 수도 있다. CABAC 의 컨텍스트에서, 용어 "bin" 은 선택스 엘리먼트의 이진화된 버전의 비트를 지칭하는데 사용된다.
- [0097] 비디오 디코더
- [0098] 도 3 은 본 개시물에 설명된 양태들에 따른 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록 다이어그램이다. 비디오 디코더 (30) 는, 예컨대 HEVC 에 대해 비디오 프레임의 단일 계층을 프로세싱하도록 구성될 수도 있다. 또한, 비디오 디코더 (30) 는 본 개시물의 기법들 중 어느 하나 또는 전부를 수행하도록 구성될 수도 있다. 일부 예들에서, 본 개시물에 설명된 기법들은 비디오 디코더 (30) 의 다양한 컴포넌트들 사이에 공유될 수도 있다. 일부 예들에서, 부가적으로 또는 대안으로, 프로세서 (미도시) 는 본 개시물에 설명된 기법들 중 어느 하나 또는 전부를 수행하도록 구성될 수도 있다.
- [0099] 설명의 목적을 위해, 본 개시물은 HEVC 코딩의 컨텍스트에서 비디오 디코더 (30) 를 설명한다. 그러나, 본 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용 가능할 수도 있다. 도 3 에 도시된 예는 단일 계층 코덱이다. 하지만, 소정의 구현들에서, 비디오 디코더 (30) 의 일부 또는 전부는 멀티 계층 코덱의 프로세싱을 위해 듀플리케이션될 수도 있다.
- [0100] 도 3 의 예에서, 비디오 디코더 (30) 는 복수의 기능적 컴포넌트들을 포함한다. 비디오 디코더 (30) 의 기능적 컴포넌트들은 엔트로피 디코딩 유닛 (150), 예측 프로세싱 유닛 (152), 역양자화 유닛 (154), 역변환 유닛 (156), 재구성 유닛 (158), 필터 유닛 (159), 및 디코딩된 픽처 버퍼 (160) 를 포함한다. 예측 프로세싱 유

닛 (152) 은 모션 보상 유닛 (162), 인트라 예측 유닛 (164), 및 계층간 예측 유닛 (166) 을 포함한다. 일부 예들에서, 비디오 디코더 (30) 는 도 2 의 비디오 인코더 (20) 에 대하여 설명된 인코딩 과정에 일반적으로 역순인 디코딩 과정을 수행할 수도 있다. 다른 예들에서, 비디오 디코더 (30) 는 더 많은, 더 적은, 또는 상이한 기능적 컴포넌트들을 포함할 수도 있다.

[0101] 비디오 디코더 (30) 는 인코딩된 비디오 데이터를 포함하는 비트스트림을 수신할 수도 있다. 비트스트림은 복수의 선택스 엘리먼트들을 포함할 수도 있다. 비디오 디코더 (30) 가 비트스트림을 수신하는 경우, 엔트로피 디코딩 유닛 (150) 은 비트스트림 상에서 파싱 동작을 수행할 수도 있다. 비트스트림 상에서 파싱 동작을 수행하는 것의 결과로서, 엔트로피 디코딩 유닛 (150) 은 비트스트림으로부터 선택스 엘리먼트들을 추출할 수도 있다. 파싱 동작을 수행하는 것의 부분으로서, 엔트로피 디코딩 유닛 (150) 은 비트스트림에서 엔트로피 인코딩된 선택스 엘리먼트들을 엔트로피 디코딩할 수도 있다. 예측 프로세싱 유닛 (152), 역양자화 유닛 (154), 역변환 유닛 (156), 재구성 유닛 (158), 및 필터 유닛 (159) 은 비트스트림으로부터 추출된 선택스 엘리먼트들에 기초하여 디코딩된 비디오 데이터를 생성하는 재구성 동작을 수행할 수도 있다.

[0102] 위에서 논의된 바와 같이, 비트스트림은 일련의 NAL 유닛들을 포함할 수도 있다. 비트스트림의 NAL 유닛들은 비디오 파라미터 세트 NAL 유닛들, 시퀀스 파라미터 세트 NAL 유닛들, 픽처 파라미터 세트 NAL 유닛들, SEI NAL 유닛들 등을 포함할 수도 있다. 비트스트림 상에서 파싱 동작을 수행하는 것의 부분으로서, 엔트로피 디코딩 유닛 (150) 은 시퀀스 파라미터 세트 NAL 유닛들로부터 시퀀스 파라미터 세트들, 픽처 파라미터 세트 NAL 유닛들로부터 픽처 파라미터 세트들, SEI NAL 유닛들로부터 SEI 데이터 등을 추출 및 엔트로피 디코딩하는 파싱 동작들을 수행할 수도 있다.

[0103] 또한, 비트스트림의 NAL 유닛들은 코딩된 슬라이스 NAL 유닛들을 포함할 수도 있다. 비트스트림 상에서 파싱 동작을 수행하는 것의 부분으로서, 엔트로피 디코딩 유닛 (150) 은 코딩된 슬라이스 NAL 유닛들로부터 코딩된 슬라이스들을 추출 및 엔트로피 디코딩하는 파싱 동작들을 수행할 수도 있다. 코딩된 슬라이스들 각각은 슬라이스 헤더 및 슬라이스 데이터를 포함할 수도 있다. 슬라이스 헤더는 슬라이스에 속하는 선택스 엘리먼트들을 포함할 수도 있다. 슬라이스 헤더에서 선택스 엘리먼트들은 슬라이스를 포함하는 픽처와 연관된 픽처 파라미터 세트들 식별하는 선택스들을 포함할 수도 있다. 엔트로피 디코딩 유닛 (150) 은 엔트로피 디코딩 동작들, 예컨대 CABAC 디코딩 동작들을, 코딩된 슬라이스 헤더에서의 선택스 엘리먼트들 상에서 수행하여, 슬라이스 헤더를 복구할 수도 있다.

[0104] 코딩된 슬라이스 NAL 유닛들로부터 슬라이스 데이터를 추출하는 것의 부분으로서, 엔트로피 디코딩 유닛 (150) 은 슬라이스 데이터에서 코딩된 CU들로부터 선택스 엘리먼트들을 추출하는 파싱 동작들을 수행할 수도 있다. 추출된 선택스 엘리먼트들은 변환 계수 블록들과 연관된 선택스 엘리먼트들을 포함할 수도 있다. 엔트로피 디코딩 유닛 (150) 은 그 후, 선택스 엘리먼트들의 일부 상에서 CABAC 디코딩 동작들을 수행할 수도 있다.

[0105] 엔트로피 디코딩 유닛 (150) 이 파티셔닝되지 않은 CU 상에서 파싱 동작을 수행한 후에, 비디오 디코더 (30) 는 파티셔닝되지 않은 CU 상에서 재구성 동작을 수행할 수도 있다. 파티셔닝되지 않은 CU 상에서 재구성 동작을 수행하기 위해, 비디오 디코더 (30) 는 CU 의 각각의 TU 상에서 재구성 동작을 수행할 수도 있다. CU 의 각각의 TU 에 대한 재구성 동작을 수행함으로써, 비디오 디코더 (30) 는 CU 와 연관된 잔차 비디오 블록을 재구성할 수도 있다.

[0106] TU 상에서 재구성 동작을 수행하는 것이 부분으로서, 역양자화 유닛 (154) 은 TU 와 연관된 변환 계수 블록을 역양자화, 예를 들어 양자화 해제할 수도 있다. 역양자화 유닛 (154) 은 H.264 디코딩 표준에 의해 정의되거나 HEVC 에 대해 제안된 역양자화 프로세스들에 유사한 방식으로 변환 계수 블록을 역양자화할 수도 있다. 역양자화 유닛 (154) 은 변환 계수 블록의 CU 에 대해 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터 (QP) 를 사용하여 양자화 정도, 및 유사하게는 적용할 역양자화 유닛 (154) 에 대한 역양자화 정도를 결정할 수도 있다.

[0107] 역양자화 유닛 (154) 이 변환 계수 블록을 역 양자화한 후에, 역변환 유닛 (156) 은 변환 계수 블록과 연관된 TU 에 대한 잔차 비디오 블록을 생성할 수도 있다. 역변환 유닛 (156) 은 TU 에 대한 잔차 블록을 생성하기 위해서 변환 계수 블록에 역변환을 적용할 수도 있다. 예를 들어, 역변환 유닛 (156) 은 역 DCT, 역정수 변환, 역 카루넬-루베 변환 (KLT), 역회전 변환, 역방향 변환, 또는 다른 역변환을 변환 계수 블록에 적용할 수도 있다. 일부 예들에서, 역변환 유닛 (156) 은 비디오 인코더 (20) 로부터의 시그널링에 기초하여 변환 계수 블록에 적용할 역변환을 결정할 수도 있다. 이러한 예들에서, 역변환 유닛 (156) 은 변환 계수 블록과 연관된 트리블록에 대한 쿼드트리의 루트 노드에서 시그널링된 변환에 기초하여 역변환을 결정할 수도 있다. 다

른 예들에서, 역변환 유닛 (156) 은 하나 이상의 코딩 특징들, 예컨대 블록 사이즈, 코딩 모드 등으로부터 역변환을 추론할 수도 있다. 일부 예들에서, 역변환 유닛 (156) 은 캐스케이드된 역변환을 적용할 수도 있다.

[0108] 일부 예들에서, 모션 보상 유닛 (162) 은 보간 필터들에 기초하여 보간을 수행함으로써 PU 의 예측된 비디오 블록을 리파이닝할 수도 있다. 서브-샘플 정확도로 모션 보상에 사용될 보간 필터들에 대한 식별자들은 신택스 엘리먼트들에 포함될 수도 있다. 모션 보상 유닛 (162) 은 PU 의 예측된 비디오 블록의 생성 동안 비디오 인코더 (20) 에 의해 사용된 동일한 보간 필터들을 사용하여, 참조 블록의 서브-정수 샘플들에 대한 보간된 값들을 계산할 수도 있다. 모션 보상 유닛 (162) 은 수신된 신택스 정보에 따라 비디오 인코더 (20) 에 의해 사용된 보간 필터들을 결정하고, 그 보간 필터들을 사용하여 예측된 비디오 블록을 생성할 수도 있다.

[0109] PU 가 인트라 예측을 사용하여 인코딩되면, 인트라 예측 유닛 (164) 은 인트라 예측을 수행하여 PU 에 대한 예측된 비디오 블록을 생성할 수도 있다. 예를 들어, 인트라 예측 유닛 (164) 은 비트스트림에서의 신택스 엘리먼트들에 기초하여 PU 에 대한 인트라 예측 모드를 결정할 수도 있다. 비트스트림은, 인트라 예측 유닛 (164) 이 PU 의 인트라 예측 모드를 결정하기 위해 사용할 수도 있는 신택스 엘리먼트들을 포함할 수도 있다.

[0110] 일부 경우들에서, 신택스 엘리먼트들은, 현재 PU 의 인트라 예측 모드를 결정하기 위해 인트라 예측 유닛 (164) 이 다른 PU 의 인트라 예측 모드를 사용한다는 것을 나타낼 수도 있다. 예를 들어, 현재 PU 의 인트라 예측 모드는 이웃하는 PU 의 인트라 예측 모드와 동일할 것 같을 수도 있다. 다시 말해, 이웃하는 PU 의 인트라 예측 모드는 현재 PU 에 대한 최고 확률의 모드일 수도 있다. 따라서, 이 예에서, 비트스트림은, PU 의 인트라 예측 모드가 이웃하는 PU 의 인트라 예측 모드와 동일하다는 것을 나타내는 작은 신택스 엘리먼트를 포함할 수도 있다. 인트라 예측 유닛 (164) 은 그 후, 공간적으로 이웃하는 PU들의 비디오 블록들에 기초하여 PU 에 대한 예측 데이터 (예를 들어, 예측된 샘플들) 를 생성하도록 인트라 예측 모드를 사용할 수도 있다.

[0111] 위에서 논의된 바와 같이, 비디오 디코더 (30) 는 또한, 계층간 예측 유닛 (166) 을 포함할 수도 있다. 계층간 예측 유닛 (128) 은 SHVC 에서 이용 가능한 하나 이상의 상이한 계층들 (예를 들어, 베이스 또는 참조 계층) 을 사용하여 현재 블록 (예를 들어, EL 에서의 현재 블록) 을 예측하도록 구성된다. 이러한 예측은 계층간 예측으로서 지칭될 수도 있다. 계층간 예측 유닛 (166) 은 예측 방법들을 이용하여 계층간 리던던시를 감소시키고, 이에 의해 코딩 효율성을 개선하고 연산적 리소스 요건들을 감소시킨다. 계층간 예측의 일부 예들은 계층간 인트라 예측, 계층간 모션 예측, 및 계층간 잔차 예측을 포함할 수도 있다. 계층간 인트라 예측은 강화 계층에서 현재 블록을 예측하도록 베이스 계층에서 공통-위치된 블록들의 재구성을 사용한다. 계층간 모션 예측은 강화 계층에서 모션을 예측하도록 베이스 계층의 모션 정보를 사용한다. 계층간 잔차 예측은 강화 계층의 잔차를 예측하도록 베이스 계층의 잔차를 사용한다. 계층간 예측 스킴들 각각은 이하에서 더 상세히 논의된다.

[0112] 재구성 유닛 (158) 은 CU 의 TU들과 연관된 잔차 비디오 블록들 및 CU 의 PU들의 예측된 블록들, 예를 들어 인트라-예측 데이터 또는 인터-예측 데이터를 적용 가능하게 사용하여 CU 의 비디오 블록을 재구성할 수도 있다. 따라서, 비디오 디코더 (30) 는 비트스트림에서의 신택스 엘리먼트들에 기초하여 잔차 비디오 블록 및 예측된 비디오 블록을 생성할 수도 있고, 예측된 비디오 블록 및 잔차 비디오 블록에 기초하여 비디오 블록을 생성할 수도 있다.

[0113] 재구성 유닛 (158) 이 CU 의 비디오 블록을 재구성한 후에, 필터 유닛 (159) 은 CU 와 연관된 비디오 블록에서 블록킹 아티팩트들을 감소시키기 위해 디블록킹 동작을 수행할 수도 있다. 필터 유닛 (159) 이 CU 와 연관된 블록킹 아티팩트들을 감소시키기 위해 디블록킹 동작을 수행한 후에, 비디오 디코더 (30) 는 디코딩된 픽처 버퍼 (160) 에 CU 의 비디오 블록을 저장할 수도 있다. 디코딩된 픽처 버퍼 (160) 는 도 1a 또는 도 1b 의 디스플레이 디바이스 (32) 와 같은 디스플레이 디바이스 상의 프리젠테이션, 인트라 예측, 및 후속의 모션 보상을 위해 참조 픽처들을 제공할 수도 있다. 예를 들어, 비디오 디코더 (30) 는 디코딩된 픽처 버퍼 (160) 에서의 비디오 블록들에 기초하여 다른 CU들의 PU들 상에서 인트라 예측 또는 인터 예측을 수행할 수도 있다.

[0114] 인트라 블록 카피 (IBC) 모드

[0115] 인트라 블록 카피 (IBC) 모드는 현재 블록 (예를 들어, 코딩 유닛, 파티셔닝 유닛 등) 이 현재 블록을 포함하는 동일한 픽처 내에서 유사한 재구성된 블록들에 기초하여 예측되는 것을 허용하는 코딩 모드이다. 현재 픽처 내에서 예측기 블록의 위치를 가리키는 블록 벡터는 IBC 모드에서 코딩되는 현재 블록과 관련하여 비트스트림에서 시그널링될 수도 있다.

[0116] IBC 모드 및 인터 모드의 통합

- [0117] 일부 구현들에서, IBC 모드는 인터 모드 (예를 들어, 현재 블록이 현재 블록을 포함하는 현재 픽처 이외의 픽처에서 유사한 재구성된 블록들에 기초하여 예측되는 것을 허용하는 코딩 모드) 와 통합된다. IBC 모드 및 인터 모드 양자 모두가 예측기 블록들의 위치를 식별하기 때문에, IBC 모드는 현재 픽처를 포함하는 참조 픽처 리스트를 사용하여 구현될 수도 있다. 예를 들어, 동일한 참조 픽처 리스트를 사용하여, 주어진 슬라이스와 연관된 참조 픽처 리스트에 현재 픽처를 부가하는 것에 의해, 주어진 슬라이스에서의 하나의 블록은 참조 픽처 리스트에서의 현재 픽처를 지칭하는 것에 의해 IBC 모드에서 코딩될 수도 있고, 주어진 슬라이스에서의 또 다른 블록은 참조 픽처 리스트에서 시간적 픽처를 지칭하는 것에 의해 인터 모드에서 코딩될 수도 있다.
- [0118] IBC 모드에서의 참조 픽처 마킹
- [0119] IBC 모드의 일부 구현들에서, 현재 슬라이스를 디코딩하기 전에, 현재 픽처는 장기 참조 픽처로서 마킹된다. 그 후, 현재 픽처의 디코딩이 완료된 후 현재 픽처는 단기 참조 픽처로서 마킹된다.
- [0120] IBC 모드와 인터 모드 사이의 차이
- [0121] 병합 모드 시그널링, 어드밴스드 모션 벡터 예측 (AMVP) 시그널링, AMVP 도출, 및 모션 벡터 차이 (MVD) 코딩을 포함한, IBC 모드와 연관된 시그널링 및 코딩 방법들은 인터 모드와 연관된 것들과 동일할 수도 있다. 일부 실시형태들에서, 인터 모드와 달리, IBC 모드에 대한 모션 벡터들이 정수 모션 벡터들이도록 요구될 수도 있다. 일부 실시형태들에서, 크로마 컴포넌트들을 제외한 루마 컴포넌트들에 대한 모션 벡터들만이 정수 모션 벡터들이도록 요구될 수도 있다. 대안으로, 루마 컴포넌트들을 제외한 크로마 컴포넌트들에 대한 모션 벡터들만이 정수 모션 벡터들이도록 요구될 수도 있다.
- [0122] 일부 경우들에서, IBC 모드에서 코딩된 블록은 대응 참조 픽처를 체크하는 것에 의해 종래 인터 코딩된 블록들과 차별화될 수 있다. 블록이 참조 픽처로서 현재 픽처만을 사용하면, 블록은 IBC 모드에서 코딩된다. 블록이 참조 픽처로서 블록을 포함한 현재 픽처 이외의 임의의 픽처를 사용하는 경우, 블록은 인터 모드에서 코딩된다.
- [0123] IBC 모드가 허용되는지 여부의 시그널링
- [0124] 일부 실시형태들에서, IBC 모드가 블록을 코딩하기 위해 사용되도록 허용되는지 여부를 표시하는 플래그는 비트 스트림에서 (예를 들어, VPS, SPS, PPS, 슬라이스 헤더 등에서) 시그널링될 수도 있다. 그러한 플래그는 코딩될 (예를 들어, 인코딩되거나 디코딩될) 블록을 포함하는 현재 픽처가 블록을 코딩 (또는 그 자체를 코딩) 하기 위해 참조 픽처로서 사용될 수 있는지 여부를 표시할 수도 있다. 대안으로 또는 부가적으로, 그러한 플래그는 현재 픽처가 현재 픽처의 참조 픽처 리스트에 부가될 수 있는지 여부를 표시할 수도 있다. 예를 들어, 플래그는 "curr_pic_as_ref_enabled_flag" 또는 "curr_pic_ref_enable_flag" 로 칭할 수도 있다.
- [0125] 참조 픽처 리스트 구축
- [0126] 표 1 내지 표 3 은 코딩될 현재 블록을 포함하는 현재 슬라이스의 참조 픽처 리스트를 구축하기 위한 일 예의 프로세스를 도시한다. 예를 들어, 변수 NumPicTotalCurr 은 현재 픽처의 예측을 위한 참조로서 사용하기 위해 이용가능한 픽처들의 총 수를 표시한다. 일부 실시형태들에서, 변수 NumPicTotalCurr 는 현재 픽처의 예측을 위한 참조로서 사용하기 위해 이용가능한 고유 픽처들의 총 수를 표시한다. 변수 umPicTotalCurr 는 다음과 같이 도출될 수도 있다.

```

NumPicTotalCurr = 0
for( i = 0; i < NumNegativePics[ CurrRpsIdx ]; i++)
    if( UsedByCurrPicS0[ CurrRpsIdx ][ i ] )
        NumPicTotalCurr++
for( i = 0; i < NumPositivePics[ CurrRpsIdx ]; i++)
    if( UsedByCurrPicS1[ CurrRpsIdx ][ i ] )
        NumPicTotalCurr++
for( i = 0; i < num_long_term_sps + num_long_term_pics; i++)
    if( UsedByCurrPicLt[ i ] )
        NumPicTotalCurr++
if( curr_pic_as_ref_enabled_flag )
    NumPicTotalCurr++
    
```

[0127]

[0128]

[표 1] NumPicTotalCurr 의 도출

[0129]

변수 NumRpsCurrTempList0 는 $\text{Max}(\text{num_ref_idx_l0_active_minus1} + 1, \text{NumPicTotalCurr})$ 와 동일하게 설정될 수도 있으며, 여기서 $\text{num_ref_idx_l0_active_minus1} + 1$ 는 슬라이스에서 특정된 참조 픽처들의 수를 나타낸다 (예를 들어, 슬라이스에서 블록들에 의해 지칭되는 참조 픽처들의 수). $\text{num_ref_idx_l0_active_minus1}$ 는 주어진 슬라이스의 슬라이스 세그먼트 헤더에서 시그널링될 수도 있다. 일부 실시형태들에서, $\text{num_ref_idx_l0_active_minus1} + 1$ 은 NumPicTotalCurr 보다 작다. 대안으로, $\text{num_ref_idx_l0_active_minus1} + 1$ 은 NumPicTotalCurr 보다 클 수도 있다. 변수 NumRpsCurrTempList0 은 슬라이스에 대한 참조 픽처 리스트 (예를 들어, RefPicList0) 를 생성하기 위해 사용될 임시 참조 픽처 리스트 (예를 들어, RefPicListTemp0) 의 사이즈를 표시할 수도 있다. 임시 리스트 RefPicListTemp0 는 다음과 같이 구축될 수도 있다:

```

rIdx = 0
while( rIdx < NumRpsCurrTempList0 ) {
    for( i = 0; i < NumPocStCurrBefore && rIdx < NumRpsCurrTempList0; rIdx++, i++)
        RefPicListTemp0[ rIdx ] = RefPicSetStCurrBefore[ i ]
    for( i = 0; i < NumPocStCurrAfter && rIdx < NumRpsCurrTempList0; rIdx++, i++)
        RefPicListTemp0[ rIdx ] = RefPicSetStCurrAfter[ i ]
    for( i = 0; i < NumPocLtCurr && rIdx < NumRpsCurrTempList0; rIdx++, i++)
        RefPicListTemp0[ rIdx ] = RefPicSetLtCurr[ i ]
    if( curr_pic_as_ref_enabled_flag )
        RefPicListTemp0[ rIdx++ ] = currPic
}
    
```

[0130]

[0131]

[표 2] RefPicListTemp0 의 구축

[0132]

위에 구축된 임시 리스트 RefPicListTemp0 에 기초하여, 슬라이스와 연관된 참조 픽처 리스트 RefPicList0 는 다음과 같이 구축될 수도 있다:

```

for( rIdx = 0; rIdx <= num_ref_idx_l0_active_minus1; rIdx++)
    RefPicList0[ rIdx ] = ref_pic_list_modification_flag_l0 ?
        RefPicListTemp0[ list_entry_l0[ rIdx ] ] : RefPicListTemp0[ rIdx ]
    
```

[0133]

[0134] **[표 3] RefPicList0 의 구축**

[0135] 플래그 ref_pic_list_modification_flag_10 은 참조 픽처 리스트가 (예를 들어, 리스트 0) 가 명시적으로 특정 되는지 (예를 들어, 비트스트림에서 시그널링된 list_entry_10[] 에서의 인덱스 값들의 리스트에 의해) 또는 암시적으로 결정되는지 (예를 들어, 임시 리스트 RefPicListTemp0 와 동일한 순서로 또는 list_entry_10 값들 이외의 값들에 기초하여) 를 표시할 수도 있다.

[0136] 참조 픽처 리스트로의 현재 픽처의 부가

[0137] IBC 모드의 일부 구현들에서, 참조 픽처 리스트 수정 (RPLM) 이 존재하지 않을 때, 현재 픽처는 코딩될 현재 블록을 포함하는 슬라이스와 연관된 참조 픽처 리스트에 강제로 부가된다. 예를 들어, 참조 픽처 리스트 수정은 참조 픽처 리스트에서의 참조 픽처들을 재순서화하기 위해 사용될 수도 있다. 표 4 는 참조 픽처 리스트에 현재 픽처를 추가하기 위한 일 예의 프로세스를 예시한다.

if(curr_pic_as_ref_enabled_flag && !ref_pic_list_modification_flag_10)
RefPicList0[num_ref_idx_l0_active_minus1] = CurrPic

[0138] **[표 4] RefPicList0 의 최종 엘리먼트로서 CurrPic 의 부가**

[0140] 최종 엘리먼트로서의 현재 픽처

[0141] 표 2 에 예시된 바와 같이, RefPicListTemp0 구축 프로세스의 WHILE 루프에 있어서, NumPicTotalCurr 이 num_ref_idx_l0_active_minus1 + 1 보다 작을 때, 동일한 참조 픽처가 임시 리스트 RefPicListTemp0 에 복수회 추가될 수 있다. 그러한 경우, RefPicListTemp0 구축 프로세스의 IF 조항에서 부가된 현재 픽처는, 적어도 한번 임시 리스트 RefPicListTemp0 에 부가된다.

[0142] 도 4 는 참조 픽처 리스트 (예를 들어, RefPicList0) 구축 프로세스 (400) 를 도시한다. 도 4 는 참조 픽처 리스트, 표 3 에 기재된 바와 같이 구축된 참조 픽처 리스트에 현재 픽처를 추가하기 전의 참조 픽처 리스트 (420), 및 표 4 에 따라 구축된 참조 픽처 리스트에 현재 픽처가 부가된 후의 참조 픽처 리스트 (430) 의 예측을 위한 참조로서 사용하기 위해 이용가능한 픽처들의 리스트 (410) 를 나타낸다.

[0143] 도 4 의 예에서, 변수 NumPicTotalCurr 는 2 와 동일하며, 이는 현재 픽처의 예측을 위해 참조로서 사용하기 위해 이용가능한 픽처들의 수가 2 인 것을 의미한다 (도 4 에 도시된 바와 같이, 현재 픽처 (C) 뿐만 아니라 시간적 픽처 (T) 를 포함할 수도 있음) 도 4 의 예에서, 변수 num_ref_idx_l0_active_minus1 + 1 는 5 와 동일하며, 이는 슬라이스로 지칭된 픽처들의 수가 5 인 것을 의미한다. 따라서, 도 4 의 예에서, NumPicTotalCurr 및 num_ref_idx_l0_active_minus1 + 1 보다 큰 변수 NumRpsCurrTempList0 는 5 와 동일하다. 따라서, 참조 픽처 리스트들 (420 및 430) 은 모두 5 개의 픽처들을 포함한다.

[0144] 참조 픽처 리스트에서의 반복 패턴

[0145] 도 4 의 예에서, 참조 픽처 리스트 (420) 는 {T, C, T, C, T} 를 포함하고, 참조 픽처 리스트 (430) 는 {T, C, T, C, C} 를 포함하며, 시간적 픽처 (T) 의 최종 인스턴스는 현재 픽처 (C) 의 인스턴스로 대체된다. 하지만, 일부 구현들에서, 참조 픽처들의 반복된 시퀀스 (예를 들어, T, C, T, C . . .) 를 포함하기 위해 참조 픽처 리스트에 대해 타당한 이유들이 있을 수도 있다. 예를 들어, 참조 픽처들의 반복된 시퀀스는 가중화된 예측을 위해 사용될 수도 있다. 도 4 의 예에서, 반복된 시퀀스는 참조 픽처 리스트의 말단에 부가된 현재 픽처에 의해 분열된다. 따라서, 그 경우, 그렇게 하는 것이 참조 픽처 리스트에서의 참조 픽처들의 반복된 시퀀스들을 분열하게 된다면, 현재 픽처들로 참조 픽처 리스트에서의 참조 픽처를 자동으로 대체하는 것 (예를 들어, 현재 픽처로 참조 픽처 리스트에서의 최종 픽처를 대체하는 것) 을 억제하는 것에 의해, 참조 픽처들의 반복된 시퀀스를 분열시키는 것을 억제하는 것이 바람직할 수도 있다.

[0146] 일부 실시형태들에서, 현재 픽처가 이미 참조 픽처 리스트에 포함되는 경우, 코더는 참조 픽처 리스트에서 현재 픽처로 참조 픽처를 대체하는 것을 억제한다. 예를 들어, 일부 실시형태들에서, 현재 픽처가 참조 픽처 리스트에 아직 포함되지 않으면, 코더는 현재 픽처로 참조 픽처 리스트에서의 최종 참조 픽처를 대체한다. 일부 실시형태들에서, 현재 픽처 (예를 들어, NumPicTotalCurr) 의 예측을 위한 참조로서 사용하기 위해 이용가능한 픽처들의 총 수가 슬라이스에서의 블록들에 의해 참조된 픽처들의 수 (예를 들어, num_ref_idx_l0_active_minus1 + 1) 보다 크지 않은 (예를 들어, 이하인) 경우, 코더는 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제한다. 일 예에서, 코더는 참조 픽처 리스트의 최종 엘리먼트

를 현재 픽처로 설정하는 것을 억제한다. 표 5 는 참조 픽처 리스트에 현재 픽처를 부가하기 위한 일 예의 프로세스를 예시한다:

<pre> if(curr_pic_as_ref_enabled_flag && !ref_pic_list_modification_flag_l0 && NumPicTotalCurr > num_ref_idx_l0_active_minus1 + 1) RefPicList0[num_ref_idx_l0_active_minus1] = CurrPic </pre>
--

[0147]

[표 5] CurrPic 이 참조 픽처 리스트에 있는 경우 CurrPic 를 부가하는 것을 억제하는 것

[0148] 일부 실시형태들에서, 코더는 NumRpsCurrTempList0 이 num_ref_idx_l0_active_minus1 + 1 보다 큰지 여부를 결정하고, NumRpsCurrTempList0 이 num_ref_idx_l0_active_minus1 + 1 보다 크지 않은 (예를 들어, 이하인) 경우, 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제할 수도 있다. 일부 실시형태들에서, 코더는 NumRpsCurrTempList0 이 num_ref_idx_l0_active_minus1 + 1 와 동일한지 여부를 결정하고, NumRpsCurrTempList0 이 num_ref_idx_l0_active_minus1 + 1 와 동일한 경우, 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제할 수도 있다.

[0150] 일부 실시형태들에서, 코더는 num_ref_idx_l0_active_minus1 + 1 이 NumPicTotalCurr 보다 큰지 여부를 결정하고, num_ref_idx_l0_active_minus1 + 1 이 NumPicTotalCurr 보다 큰 경우, 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제할 수도 있다. 일부 실시형태들에서, 코더는 num_ref_idx_l0_active_minus1 + 1 이 NumRpsCurrTempList0 보다 큰지 여부를 결정하고, num_ref_idx_l0_active_minus1 + 1 이 NumRpsCurrTempList0 보다 큰 경우, 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제할 수도 있다.

[0151] 참조 픽처 리스트에 현재 픽처를 부가하기 위한 예시의 프로세스

[0152] 도 5 는 본 개시물의 양태들에 따른 현재 블록을 포함하는 슬라이스와 연관된 참조 픽처 리스트를 구축하기 위한 방법 (500) 을 도시하는 플로우차트이다. 도 5 에 도시된 단계들은 비디오 인코더 (예를 들어, 비디오 인코더 (20)), 비디오 디코더 (예를 들어, 비디오 디코더 (30)), 또는 임의의 다른 컴포넌트에 의해 수행될 수도 있다. 편의를 위해, 방법 (500) 은, 비디오 인코더 (20), 비디오 디코더 (30), 또는 다른 컴포넌트일 수도 있는, 비디오 코더 (또한 간단히 코더로서도 지칭됨) 에 의해 수행되는 것으로 기재된다.

[0153] 블록 (502) 에서, 코더는 (i) 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들의 총 수를 나타내는 제 1 수, 및 (ii) 참조 픽처 리스트에 포함될 참조 픽처들의 총 수를 나타내는 제 2 수를 결정한다. 일부 실시형태들에서, 인터 모드 또는 인트라 블록 카피 모드 중 적어도 하나에서의 예측을 위해 이용가능한 참조 픽처들은 모두 디코딩된 픽처들을 포함한다. 다른 실시형태에서, 참조 픽처들은 모두 디코딩된 픽처 버퍼에서의 디코딩된 픽처들을 포함한다. 다른 실시형태에서, 참조 픽처들은 모두 현재 픽처 및 인터 모드에서의 현재 픽처에서의 블록을 예측하는데 사용하기 위해 이용가능한 참조 픽처들을 포함한다. 또 다른 실시형태에서, 참조 픽처들은 (i) 현재 픽처를 예측하기 위해 이용가능하고 출력 순서에서 현재 픽처를 선행하는 단기 참조 픽처들의 총 수 (예를 들어, NumPocStCurrBefore 으로 표시됨), (ii) 현재 픽처를 예측하기 위해 이용가능하고 출력 순서에서 현재 픽처를 후속하는 단기 참조 픽처들의 총 수 (예를 들어, NumPocStCurrAfter 로 표시됨), (iii) 현재 픽처를 예측하기 위해 이용가능한 장기 참조 픽처들의 총 수 (예를 들어, NumPocLtCurr 로 표시됨), 및 (iv) 현재 픽처를 포함한다. 일부 그러한 실시형태들에서, 현재 픽처가 총 수에 포함되는지 여부는 (예를 들어, curr_pic_as_ref_enabled_flag 로 표시된) 현재 픽처에서의 블록을 예측하기 위해 사용되도록 현재 픽처가 허용되는지 여부에 기초한다. 일부 실시형태들에서, 제 1 수는 HVC 의 변수 NumPicTotalCurr 로 표시되고, 제 2 수는 HEVC 의 변수 num_ref_idx_l0_active_minus1 + 1 로 표시된다. 도 5 에 도시되지는 않았지만, 블록 (502) 전에, 코더는 표 1 내지 표 3 에 예시된 프로세스들에 따라 참조 픽처를 구축할 수도 있다.

[0154] 블록 (504) 에서, 코더는 제 2 수가 제 1 수 이상인지 여부를 결정한다. 코더가 제 2 수가 제 1 수 이상이라고 코더가 결정하는 경우, 방법 (500) 은 블록 (506) 으로 진행한다. 그렇지 않으면, 방법 (500) 은 블록 (508) 로 진행한다.

[0155] 블록 (506) 에서, 코더는 예측될 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체하는 것을 억제한다. 일부 실시형태들에서, 코더는 현재 픽처로 참조 픽처 리스트에서의 최종 포지션에 참조 픽처를 대체하는 것을 억제한다. 다른 실시형태에서, 코더는 현재 픽처가 아닌 참조 픽처 리스트의 최종 포지션에서 참조 픽처를

현재 픽처로 대체하는 것을 억제한다.

- [0156] 블록 (508) 에서, 코더는 현재 픽처로 참조 픽처 리스트에서의 참조 픽처를 대체한다. 일부 실시형태들에서, 코더는 현재 픽처로 참조 픽처 리스트의 최종 포지션에서의 참조 픽처를 대체한다. 다른 실시형태에서, 코더는 현재 픽처가 아닌 참조 픽처 리스트의 최종 포지션에서의 참조 픽처를 현재 픽처로 대체한다. 다른 실시형태에서, 코더는 참조 픽처가 현재 픽처인지 여부에 관계 없이 현재 픽처로 참조 픽처 리스트의 최종 포지션에서의 참조 픽처들을 대체한다.
- [0157] 도 5 에 도시되지는 않았지만, 코더는 참조 픽처 리스트를 사용하여 현재 픽처에 현재 블록에 대한 예측 블록을 결정할 수도 있다. 추가로, 코더는 예측 블록에 기초하여 현재 블록을 인코딩할 수도 있다. 대안으로 또는 부가적으로, 코더는 예측 블록에 기초하여 현재 블록을 디코딩할 수도 있다.
- [0158] 일부 실시형태들에서, 코더는 현재 픽처 (또는 현재 픽처에 대응하는 이미지 데이터 또는 비디오 데이터) 를 캡처하도록 구성된 카메라를 또한 포함하는 장치에 포함된다. 일부 실시형태들에서, 코더는 현재 픽처 (또는 현재 픽처에 대응하는 이미지 데이터 또는 비디오 데이터) 를 디스플레이하도록 구성된 디스플레이를 또한 포함하는 장치에 포함된다.
- [0159] 다른 참조 픽처 리스트들
- [0160] 이러한 구축의 참조 픽처의 하나 이상의 기법들은 일 예로서 참조 픽처 리스트 0 에 기초하여 본 명세서에 기재된다. 하지만, 유사한 기법들이 다른 타입의 참조 픽처 리스트들로 확장될 수도 있다. 예를 들어, 이 개시물의 기법들은 리스트 0 또는 10 을 리스트 1 또는 11 로 각각 대체하는 것에 의해 참조 픽처 리스트 1 로 확장될 수도 있다.
- [0161] 다른 고려사항들
- [0162] 정보 및 신호들은 다양한 상이한 기술들 및 기법들 중 임의의 것을 사용하여 표현될 수도 있다. 예를 들어, 상기 설명을 통해 참조될 수도 있는 데이터, 명령들, 커맨드들, 정보, 신호들, 비트들, 심볼들, 및 칩들은 전압들, 전류들, 전자기파들, 자기장들 또는 자기 입자들, 광학 필드들 또는 입자들, 이들의 임의의 조합에 의해 표현될 수도 있다.
- [0163] 본원에서 개시된 실시형태들과 관련하여 설명된 다양한 예시적인 논리적 블록들, 모듈들, 회로들, 및 알고리즘 단계들은 전자 하드웨어, 컴퓨터 소프트웨어, 또는 양자의 조합들로서 구현될 수도 있다. 하드웨어 및 소프트웨어의 이 교환가능성을 명확하게 예시하기 위하여, 다양한 예시적인 컴포넌트들, 블록들, 및 단계들은 일반적으로 그 기능성의 측면에서 위에서 설명되었다. 이러한 기능성이 하드웨어 또는 소프트웨어로서 구현되는지 여부는 특정한 애플리케이션과, 전체 시스템에 부과된 설계 제약들에 종속된다. 당업자들은 각각의 특정한 애플리케이션을 위한 다양한 방법들로 설명된 기능성을 구현할 수도 있지만, 이러한 구현 판단들은 본 개시물의 범위로부터의 이탈을 야기시키는 것으로서 해석되지 않아야 한다.
- [0164] 본원에서 설명된 기법들은 하드웨어, 소프트웨어, 펌웨어, 또는 그 임의의 조합으로 구현될 수도 있다. 이러한 기법들은 범용 컴퓨터들, 무선 통신 디바이스 핸드셋들, 또는 무선 통신 디바이스 핸드셋들 및 다른 디바이스들에서의 애플리케이션을 포함하는 다수의 용도들을 가지는 집적 회로 디바이스들과 같은 다양한 디바이스들 중의 임의의 것에서 구현될 수도 있다. 디바이스들 또는 컴포넌트들로서 설명된 임의의 특징들은 집적된 로직 디바이스에서 함께, 또는 개별적이지만 상호 동작가능한 로직 디바이스들로서 별도로 구현될 수도 있다. 소프트웨어로 구현될 경우, 기법들은, 실행될 때, 위에서 설명된 방법들 중의 하나 이상을 수행하는 명령들을 포함하는 프로그램 코드를 포함하는 컴퓨터-판독가능한 데이터 저장 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터-판독가능 데이터 저장 매체는 패키징 재료들을 포함할 수도 있는 컴퓨터 프로그램 제품의 일부를 형성할 수도 있다. 컴퓨터-판독가능 매체는 동기식 랜덤 액세스 메모리 (synchronous dynamic random access memory; SDRAM) 와 같은 랜덤 액세스 메모리 (random access memory; RAM), 판독-전용 메모리 (read-only memory; ROM), 비-휘발성 랜덤 액세스 메모리 (non-volatile random access memory; NVRAM), 전기적 소거가능 프로그래밍가능 판독-전용 메모리 (electrically erasable programmable read-only memory; EEPROM), 플래시 메모리 (FLASH memory), 자기 또는 광학 데이터 저장 매체들 등과 같은 메모리 또는 데이터 저장 매체들을 포함할 수도 있다. 부가적으로 또는 대안적으로, 기법들은 전파된 신호들 또는 파 (wave) 들과 같이, 명령들 또는 데이터 구조들의 형태로 프로그램 코드를 반송하거나 통신하며 컴퓨터에 의해 액세스, 판독, 및/또는 실행될 수 있는 컴퓨터-판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.
- [0165] 프로그램 코드는, 하나 이상의 디지털 신호 프로세서 (digital signal processor; DSP) 들, 범용 마이크로프로

세서들, 애플리케이션 특정 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍 가능한 로직 어레이 (field programmable logic array; FPGA) 들, 또는 다른 등가의 집적 또는 개별 로직 회로부와 같은 하나 이상의 프로세서들을 포함할 수도 있는 프로세서에 의해 실행될 수도 있다. 이러한 프로세서는 본 개시에서 설명된 기법들 중의 임의의 것을 수행하도록 구성될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있지만, 대안적으로, 프로세서는 임의의 기존의 프로세서, 제어기, 마이크로제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한, 컴퓨팅 디바이스들의 조합, 예컨대, DSP 및 마이크로프로세서, 복수의 마이크로프로세서들, DSP 코어와 함께 하나 이상의 마이크로프로세서들, 또는 임의의 다른 이러한 구성의 조합으로서 구현될 수도 있다. 따라서, 본원에서 이용된 바와 같은 용어 "프로세서" 는 상기한 구조, 상기 구조의 임의의 조합, 또는 본원에서 설명된 기법들의 구현을 위해 적당한 임의의 다른 구조 또는 장치 중의 임의의 것을 지칭할 수도 있다. 게다가, 일부의 양태들에서, 본원에서 설명된 기능성은, 인코딩 및 디코딩을 위해 구성되거나, 조합된 비디오 인코더-디코더 (combined video encoder-decoder; CODEC) 내에 통합된 전용 소프트웨어 또는 하드웨어 내에 제공될 수도 있다. 또한, 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

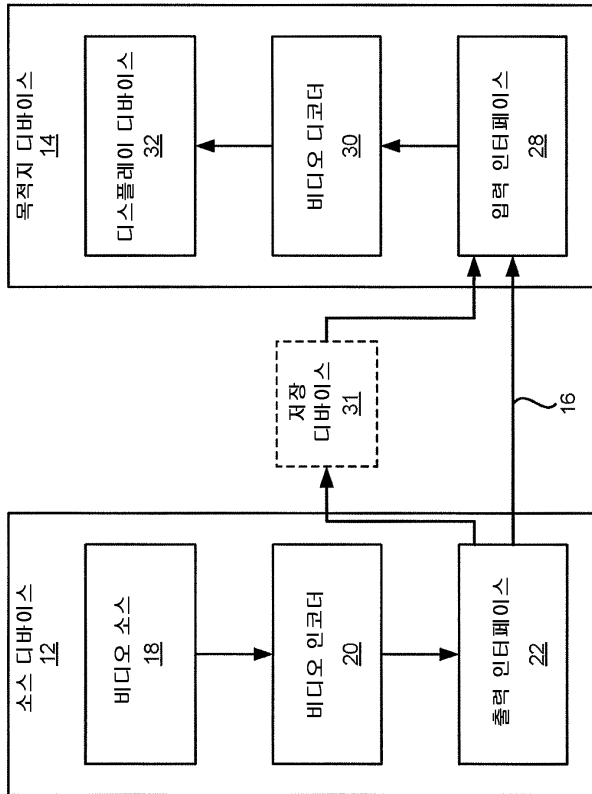
[0166] 본 개시물의 기법들은 무선 핸드셋, 집적 회로 (IC), 또는 IC 들의 세트 (예를 들어, 칩 세트) 를 포함하는 광범위한 디바이스들 또는 장치들로 구현될 수도 있다. 개시된 기술들을 수행하도록 구성된 디바이스들의 기능적 양태를 강조하기 위해 다양한 소자들, 모듈들, 또는 유닛들이 본 개시물에서 설명되었지만, 반드시 상이한 하드웨어 유닛들에 의해 실현될 필요는 없다. 오히려, 전술한 바와 같이 다양한 유닛들은 적합한 소프트웨어 및/또는 펌웨어와 관련되어, 전술된 하나 이상의 프로세서들을 포함하여 상호 동작적인 하드웨어 유닛들의 집합에 의해 제공되고 또는 코텍 하드웨어 유닛에 결합될 수도 있다.

[0167] 상술한 것은 다양한 실시형태들과 관련하여 기재되었지만, 하나의 실시형태로부터의 피쳐들 또는 엘리먼트들은 이 개시물의 기법들로부터 벗어나지 않으면서 다른 실시형태들과 조합될 수도 있다. 하지만, 개별 실시형태들 간 피쳐들의 조합들이 반드시 거기에 제한되지 않는다. 본 발명의 다양한 실시형태들이 설명되었다. 이들 및 다른 예들은 하기의 특허청구범위 내에 있다.

도면

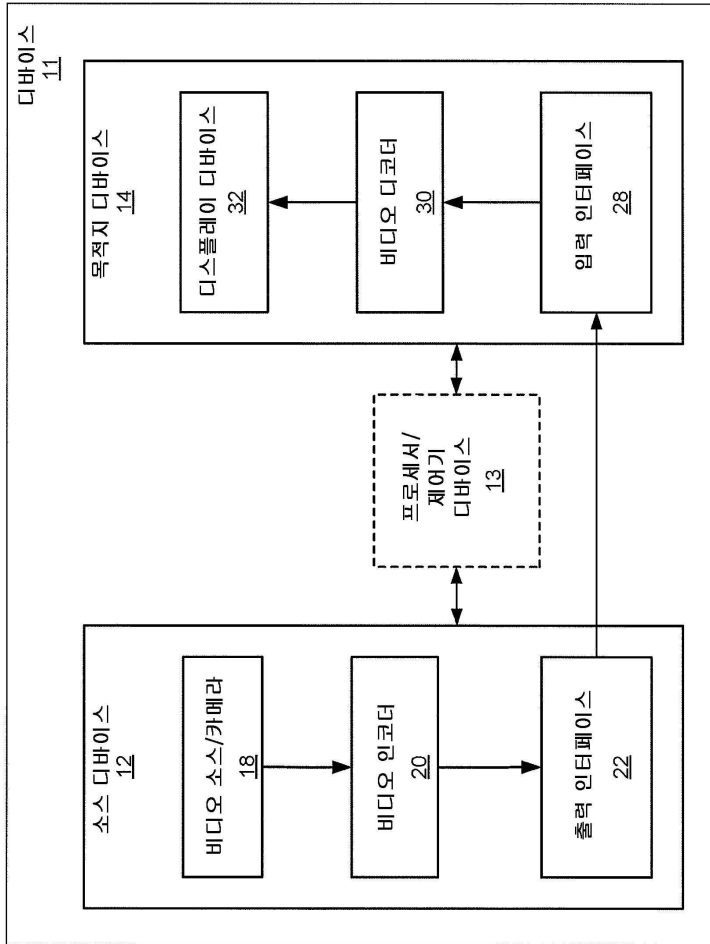
도면1a

10

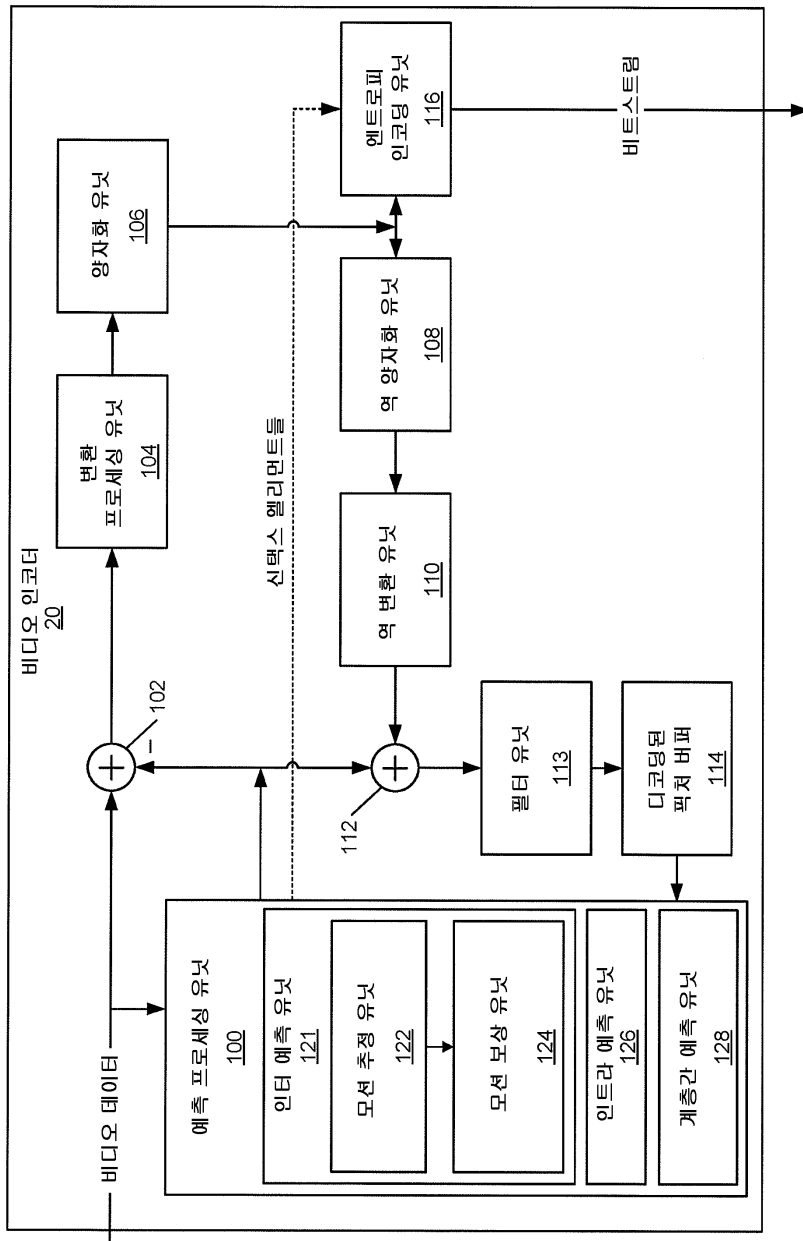


도면1b

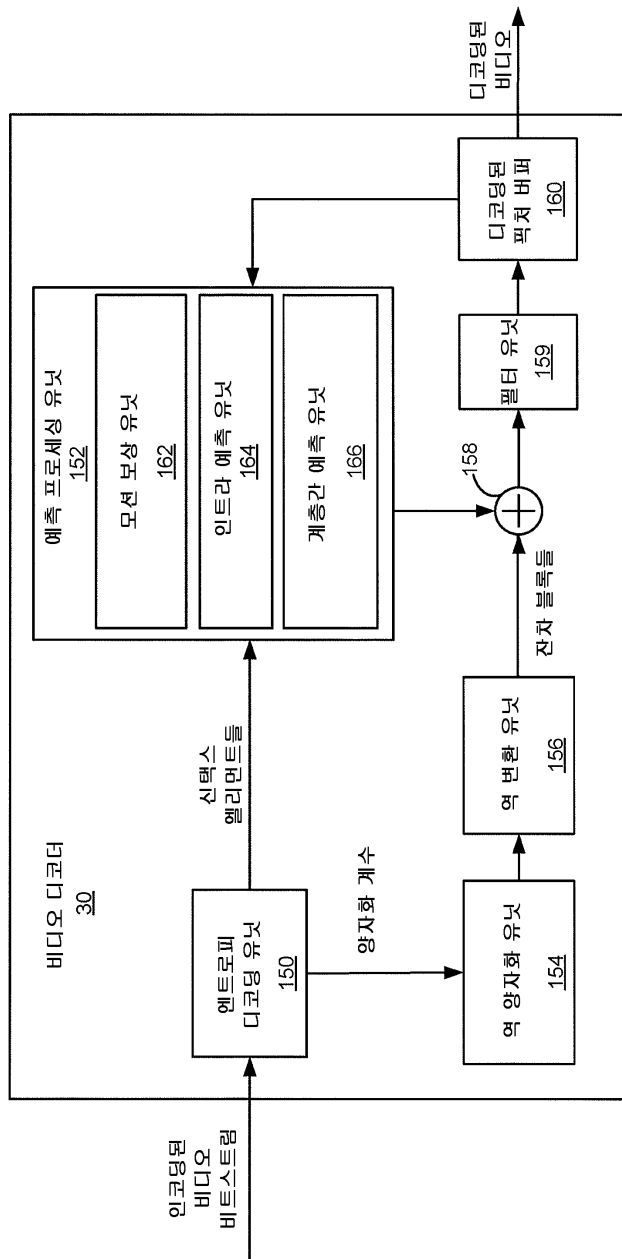
10'



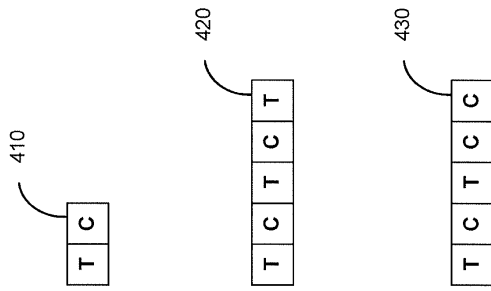
도면2



도면3



도면4



400 ↗

도면5

500 ↘

