



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0115984
(43) 공개일자 2022년08월19일

- (51) 국제특허분류(Int. Cl.)
H04N 19/157 (2014.01) H04N 19/105 (2014.01)
H04N 19/186 (2014.01) H04N 19/70 (2014.01)
- (52) CPC특허분류
H04N 19/157 (2015.01)
H04N 19/105 (2015.01)
- (21) 출원번호 10-2022-7023277
- (22) 출원일자(국제) 2020년11월17일
심사청구일자 없음
- (85) 번역문제출일자 2022년07월06일
- (86) 국제출원번호 PCT/US2020/060852
- (87) 국제공개번호 WO 2021/137962
국제공개일자 2021년07월08일
- (30) 우선권주장
62/954,843 2019년12월30일 미국(US)

- (71) 출원인
알리바바 그룹 홀딩 리미티드
케이만군도, 그랜드 케이만, 조지 타운, 피.오.
박스 847, 원 캐피탈 플레이스, 플로어 4
- (72) 발명자
사르위, 모함메드, 고렘
미국 캘리포니아 95014 쿠퍼티노 메리먼 알디.
10485
예, 안
미국 캘리포니아 92130 샌디에고 펄만 웨이 5001
(뒷면에 계속)
- (74) 대리인
특허법인 광장리앤코

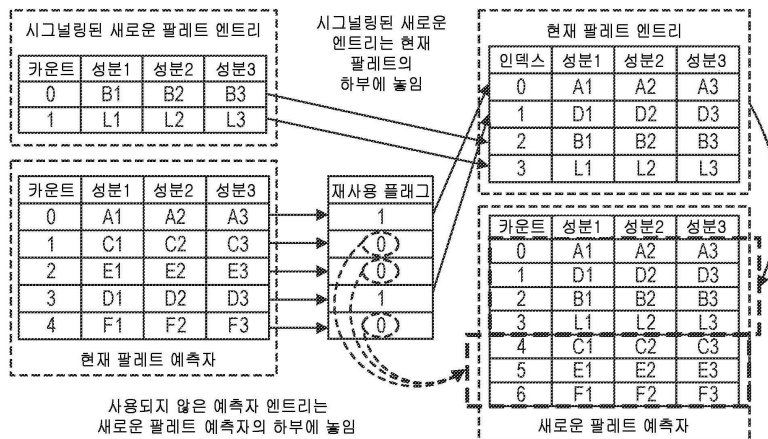
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 팔레트 모드에서 비디오 데이터를 코딩하기 위한 방법 및 장치

(57) 요약

본 개시는 비디오 데이터의 팔레트 코딩을 수행하기 위한 장치 및 방법을 제공한다. 특정 개시된 실시예에 따라, 방법은: 코딩 유닛(CU)의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 단계; 및 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여, CU에 대한 제1 최대 팔레트-포 크기를 결정하는 단계; CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계; 및 제1 최대 팔레트-포 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 포함한다.

대표도



팔레트 예측자를 업데이트하기 위한 예시적인 프로세스

(52) CPC특허분류

HO4N 19/186 (2015.01)

HO4N 19/70 (2015.01)

(72) 발명자

루오, 지안콩

미국 뉴저지 08558 스킨먼 리치몬드 디알. 28

라오, 루-링

중국 베이징 차오양 디스트릭트 왕징 동 위안 더 4
디스트릭트 라디언스 진 후이 타워

명세서

청구범위

청구항 1

팔레트 코딩 방법(palette coding method)으로서,

코딩 유닛(CU)의 루마 성분(luma component) 및 상기 CU의 크로마 성분(chroma component)이 팔레트 모드에서 공동으로(jointly) 또는 개별적으로(separately) 코딩되는지를 결정하는 단계; 및

상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 공동으로 코딩되는 것에 응답하여,

상기 CU에 대한 제1 최대 팔레트-표 크기(maximum palette-table size)를 결정하는 단계;

상기 CU에 대한 제1 최대 팔레트-예측자 크기(maximum palette-predictor size)를 결정하는 단계; 및

상기 제1 최대 팔레트-표 크기 및 상기 제1 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 포함하는,

팔레트 코딩 방법.

청구항 2

제1 항에 있어서,

상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 결정하는 단계는:

비디오 비트스트림에서 시그널링된 제1 선택스 요소(syntax element)의 값에 기초하여, 상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 결정하는 단계; 및

상기 비디오 비트스트림에서 시그널링된 상기 제1 선택스 요소의 값 및 제2 선택스 요소의 값에 기초하여, 상기 CU에 대한 상기 제1 최대 팔레트-예측자 크기를 결정하는 단계를 포함하는,

팔레트 코딩 방법.

청구항 3

제2 항에 있어서,

상기 CU에 대한 상기 제1 최대 팔레트-예측자 크기를 결정하는 단계는:

상기 CU에 대한 상기 제1 최대 팔레트-예측자 크기를 상기 제1 선택스 요소의 값과 상기 제2 선택스 요소의 값의 합인 것으로 결정하는 단계를 포함하는,

팔레트 코딩 방법.

청구항 4

제2 항에 있어서,

상기 제1 선택스 요소 및 상기 제2 선택스 요소는 상기 CU와 연관된 시퀀스 파라미터 세트(SPS) 또는 픽처 헤더(PH)에서 시그널링되는,

팔레트 코딩 방법.

청구항 5

제1 항에 있어서,

상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,

상기 제1 최대 팔레트-표 크기에 기초하여 상기 CU에 대한 제2 최대 팔레트-표 크기를 결정하는 단계,

상기 제1 최대 팔레트-예측자 크기에 기초하여 상기 CU에 대한 제2 최대 팔레트-예측자 크기를 결정하는 단계, 및

상기 제2 최대 팔레트-표 크기 및 상기 제2 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 더 포함하는,

팔레트 코딩 방법.

청구항 6

제1 항에 있어서,

상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 결정하는 단계는:

상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 제1 미리 결정된 값인 것으로 결정하는 단계; 및

상기 CU에 대한 제1 최대 팔레트-예측자 크기를 제2 미리 결정된 값인 것으로 결정하는 단계를 포함하는,

팔레트 코딩 방법.

청구항 7

제6 항에 있어서,

상기 제1 미리 결정된 값은 31이고; 그리고

상기 제2 미리 결정된 값은 63인,

팔레트 코딩 방법.

청구항 8

제6 항에 있어서,

상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,

상기 CU에 대한 제3 최대 팔레트-표 크기를 제3 미리 결정된 값인 것으로 결정하는 단계; 및

상기 제3 최대 팔레트-표 크기에 기초하여, 상기 CU를 예측하는 단계를 더 포함하고,

상기 제3 미리 결정된 값은 상기 제1 미리 결정된 값보다 작은,

팔레트 코딩 방법.

청구항 9

제8 항에 있어서,

상기 제3 미리 결정된 값은 15인,

팔레트 코딩 방법.

청구항 10

제8 항에 있어서,

상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 개별적으로 코딩되고 상기 CU가 단일 트리 슬라이스(single-tree slice)의 일부인 것에 응답하여, 상기 제1 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 더 포함하는,

팔레트 코딩 방법.

청구항 11

제8 항에 있어서,

상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 개별적으로 코딩되고 상기 CU가 단일 트리 슬라이스

스의 일부가 아닌 것에 응답하여,
 상기 CU에 대한 제3 최대 팔레트-예측자 크기를 제4 미리 결정된 값인 것으로 결정하는 단계; 및
 상기 제3 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 더 포함하고,
 상기 제4 미리 결정된 값은 상기 제2 미리 결정된 값보다 작은,
 팔레트 코딩 방법.

청구항 12

제11 항에 있어서,
 상기 제4 미리 결정된 값은 31인,
 팔레트 코딩 방법.

청구항 13

제1 항에 있어서,
 상기 CU를 포함하는 픽처 슬라이스가 단일 트리 슬라이스 또는 이중 트리 슬라이스(dual-tree slice)인지를 결정하는 단계; 및
 상기 픽처 슬라이스가 단일 트리 슬라이스인 것에 응답하여,
 상기 픽처 슬라이스의 슬라이스 헤더에서 시그널링된 제3 선택스 요소의 값에 기초하여, 상기 픽처 슬라이스 내의 상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 결정하는 단계; 및
 상기 슬라이스 헤더에서 시그널링된 상기 제3 선택스 요소의 값 및 제4 선택스 요소의 값에 기초하여, 상기 CU에 대한 상기 제1 최대 팔레트-예측자 크기를 결정하는 단계를 더 포함하는,
 팔레트 코딩 방법.

청구항 14

제13 항에 있어서,
 상기 픽처 슬라이스가 이중 트리 슬라이스인 것에 응답하여,
 상기 슬라이스 헤더에서 시그널링된 제5 선택스 요소의 값에 기초하여, 상기 CU에 대한 제4 최대 팔레트-표 크기를 결정하는 단계,
 상기 슬라이스 헤더에서 시그널링된 상기 제5 선택스 요소의 값 및 제6 선택스 요소의 값에 기초하여, 상기 CU에 대한 제4 최대 팔레트-예측자 크기를 결정하는 단계, 및
 상기 제4 최대 팔레트-표 크기 및 상기 제4 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 더 포함하는,
 팔레트 코딩 방법.

청구항 15

비디오 처리 장치로서,
 명령어를 저장하기 위한 적어도 하나의 메모리; 및
 상기 명령어를 실행하도록 구성된 적어도 하나의 프로세서를 포함하고,
 상기 명령어는 상기 장치로 하여금:
 코딩 유닛(CU)의 루마 성분 및 상기 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 것; 및
 상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 공동으로 코딩되는 것에 응답하여,

상기 CU에 대한 제1 최대 팔레트-표 크기를 결정하는 것;
상기 CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 것; 및
상기 제1 최대 팔레트-표 크기 및 상기 제1 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 것을 수행하게 하는,
비디오 처리 장치.

청구항 16

명령어의 세트를 저장하는 비일시적 컴퓨터 판독 가능 저장 매체로서,
상기 명령어의 세트는 비디오 처리 장치로 하여금 방법을 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하고,
상기 방법은:
코딩 유닛(CU)의 루마 성분 및 상기 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 단계; 및
상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 공동으로 코딩되는 것에 응답하여,
상기 CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계;
상기 CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계; 및
상기 제1 최대 팔레트-표 크기 및 상기 제1 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 포함하는,
비일시적 컴퓨터 판독 가능 저장 매체.

청구항 17

제16 항에 있어서,
상기 명령어의 세트는 상기 비디오 처리 장치로 하여금:
비디오 비트스트림에서 시그널링된 제1 신택스 요소의 값에 기초하여, 상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 결정하는 단계; 및
상기 비디오 비트스트림에서 시그널링된 상기 제1 신택스 요소의 값 및 제2 신택스 요소의 값에 기초하여, 상기 CU에 대한 상기 제1 최대 팔레트-예측자 크기를 결정하는 단계를 수행하게 하도록 상기 하나 이상의 처리 디바이스에 의해 실행 가능한,
비일시적 컴퓨터 판독 가능 저장 매체.

청구항 18

제16 항에 있어서,
상기 명령어의 세트는 상기 비디오 처리 장치로 하여금:
상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
상기 제1 최대 팔레트-표 크기에 기초하여 상기 CU에 대한 제2 최대 팔레트-표 크기를 결정하는 단계,
상기 제1 최대 팔레트-예측자 크기에 기초하여 상기 CU에 대한 제2 최대 팔레트-예측자 크기를 결정하는 단계, 및
상기 제2 최대 팔레트-표 크기 및 상기 제2 최대 팔레트-예측자 크기에 기초하여 상기 CU를 예측하는 단계를 수행하게 하도록 상기 하나 이상의 처리 디바이스에 의해 실행 가능한,
비일시적 컴퓨터 판독 가능 저장 매체.

청구항 19

제16 항에 있어서,

상기 명령어의 세트는 상기 비디오 처리 장치로 하여금:

상기 CU에 대한 상기 제1 최대 팔레트-표 크기를 제1 미리 결정된 값인 것으로 결정하는 단계; 및

상기 CU에 대한 제1 최대 팔레트-예측자 크기를 제2 미리 결정된 값인 것으로 결정하는 단계를 수행하게 하도록 상기 하나 이상의 처리 디바이스에 의해 실행 가능한,

비일시적 컴퓨터 판독 가능 저장 매체.

청구항 20

제19 항에 있어서,

상기 명령어의 세트는 상기 비디오 처리 장치로 하여금:

상기 루마 성분 및 상기 크로마 성분이 상기 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,

상기 CU에 대한 제3 최대 팔레트-표 크기를 제3 미리 결정된 값인 것으로 결정하는 단계; 및

상기 제3 최대 팔레트-표 크기에 기초하여, 상기 CU를 예측하는 단계를 수행하게 하도록 상기 하나 이상의 처리 디바이스에 의해 실행 가능하고,

상기 제3 미리 결정된 값은 상기 제1 미리 결정된 값보다 작은,

비일시적 컴퓨터 판독 가능 저장 매체.

발명의 설명

기술 분야

[0001] **관련 출원에 대한 상호 참조**

[0002] 본 개시는 2019년 12월 30일에 출원된 미국 가출원 제62/954,843호를 우선권으로 주장하고, 상기 출원은 그 전체가 본원에 참조로 통합된다.

[0003] 본 개시는 일반적으로 비디오 처리에 관한 것이며, 더 상세하게는, 팔레트 모드(palette mode)에서 루마(luma) 및 크로마 성분(chroma component)에 대한 코딩 트리 구조(coding tree structure)에 기초하여 최대 팔레트-표 크기(maximum palette-table size) 및 최대 팔레트-예측자 크기(maximum palette-predictor size)를 시그널링 및 결정하기 위한 방법 및 장치에 관한 것이다.

배경 기술

[0004] 비디오는 시각적 정보를 캡처하는 정적 픽처(또는 "프레임")의 세트이다. 저장 메모리 및 송신 대역폭을 감소시키기 위해, 비디오는 저장 또는 송신 이전에 압축되고, 디스플레이 전에 압축 해제될 수 있다. 압축 프로세스는 보통 인코딩으로 지칭되고, 압축 해제 프로세스는 보통 디코딩으로 지칭된다. 가장 일반적으로 예측, 변환, 양자화, 엔트로피 코딩(entropy coding) 및 인-루프 필터링(in-loop filtering)에 기초하는, 표준화된 비디오 코딩 기술을 사용하는 다양한 비디오 코딩 포맷이 있다. 특정 비디오 코딩 포맷을 지정하는 고효율 비디오 코딩(High Efficiency Video Coding, HEVC/H.265) 표준, 다용도 비디오 코딩(Versatile Video Coding, VVC/H.266) 표준 AVS 표준과 같은 비디오 코딩 표준은 표준화 기구에서 개발한다. 점점 더 많은 진보된 비디오 코딩 기술이 비디오 표준에 채택되고 있고, 새로운 비디오 코딩 표준의 코딩 효율성은 점점 더 높아지고 있다.

발명의 내용

[0005] 일부 실시예에서, 예시적인 팔레트 코딩 방법은: 코딩 유닛(CU)의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 단계; 및 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계; CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계; 및 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초

하여 CU를 예측하는 단계를 포함한다.

[0006] 일부 실시예에서, 예시적인 비디오 처리 장치는 명령어를 저장하기 위한 적어도 하나의 메모리, 및 적어도 하나의 프로세서를 포함한다. 적어도 하나의 프로세서는 장치로 하여금: CU의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 것; 및 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 것; CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 것; 및 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.

[0007] 일부 실시예에서, 예시적인 비일시적 컴퓨터-판독 가능 저장 매체는 명령어의 세트를 저장한다. 명령어의 세트는 비디오 처리 장치로 하여금: CU의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 것; 및 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 것; CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 것; 및 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.

도면의 간단한 설명

[0008] 본 개시의 실시예 및 다양한 양상은 다음의 상세한 설명 및 첨부 도면에 예시된다. 도면에 도시된 다양한 피쳐(feature)는 축척대로 도시된 것은 아니다.

도 1은 본 개시의 일부 실시예에 따른, 예시적인 비디오 시퀀스의 구조를 도시하는 개략도이다.

도 2a는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 예시적인 인코딩 프로세스를 도시하는 개략도이다.

도 2b는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 다른 예시적인 인코딩 프로세스를 도시하는 개략도이다.

도 3a는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 예시적인 디코딩 프로세스를 도시하는 개략도이다.

도 3b는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 다른 예시적인 디코딩 프로세스를 도시하는 개략도이다.

도 4는 본 개시의 일부 실시예와 일치하는, 비디오를 인코딩하거나 또는 디코딩하기 위한 예시적인 장치의 블록도이다.

도 5는 본 개시의 일부 실시예에 따른, 팔레트 모드에서 코딩된 예시적인 블록의 개략도를 도시한다.

도 6은 본 개시의 일부 실시예에 따른, 코딩 유닛을 인코딩한 이후에, 팔레트-예측자를 업데이트하기 위한 예시적인 프로세스의 개략도를 도시한다.

도 7은 본 개시의 일부 실시예에 따른, 예시적인 균일한 최대 예측자 크기 및 최대 팔레트 크기를 도시하는 예시적인 표 1을 도시한다.

도 8은 본 개시의 일부 실시예에 따른, 예시적인 최대 예측자 크기 및 최대 팔레트 크기를 도시하는 예시적인 표 2를 도시한다.

도 9는 본 개시의 일부 실시예에 따른, 미리 정의된 최대 팔레트-예측자 크기 및 최대 팔레트 크기를 사용하기 위한 예시적인 디코딩 프로세스를 도시하는 예시적인 표 3을 도시한다.

도 10은 본 개시의 일부 실시예에 따른, 미리 정의된 최대 팔레트-예측자 크기 및 최대 팔레트 크기를 사용하기 위한 예시적인 팔레트 코딩 선택스 표를 도시하는 예시적인 표 4를 도시한다.

도 11은 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 예시적인 도출을 도시하는 예시적인 표 5를 도시한다.

도 12는 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 다른 예시적인 도출을 도시하는 예시적인 표 6을 도시한다.

도 13은 본 개시의 일부 실시예에 따른, 예시적인 SPS(sequence parameter set) 선택스 표를 도시하는 예시적인 표 7을 도시한다.

도 14는 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 다른 예시적인 도출을 도시하는 예시적인 표 8을 도시한다.

도 15는 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 다른 예시적인 도출을 도시하는 예시적인 표 9를 도시한다.

도 16은 본 개시의 일부 실시예에 따른, 예시적인 PH(picture header) 선택스를 도시하는 예시적인 표 10을 도시한다.

도 17은 본 개시의 일부 실시예에 따른, I, P 및 B 슬라이스에 대한 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 예시적인 도출을 도시하는 예시적인 표 11을 도시한다.

도 18은 본 개시의 일부 실시예에 따른, 예시적인 SH(slice header) 선택스를 도시하는 예시적인 표 12를 도시한다.

도 19는 본 개시의 일부 실시예에 따른, 예시적인 팔레트 코딩 방법의 흐름도를 도시한다.

발명을 실시하기 위한 구체적인 내용

- [0009] 이제 예시적인 실시예에 대한 참조가 상세히 이루어질 것이며, 이의 예시는 첨부 도면에 예시된다. 다음 설명은 달리 표시되지 않는 한 상이한 도면에서 동일한 번호가 동일하거나 유사한 요소를 나타내는 첨부 도면을 참조한다. 예시적인 실시예의 다음 설명에서 제시된 구현은 본 발명과 일치하는 모든 구현을 나타내는 것은 아니다. 그 대신에, 이들은 첨부된 청구범위에서 인용된 본 발명에 관련된 양상과 일치하는 장치 및 방법의 예시일 뿐이다. 본 개시의 특정 양상은 아래에서 더 상세히 설명된다. 참조로 통합된 용어 및/또는 정의와 상충하는 경우, 본원에 제공된 용어 및 정의가 우선한다.
- [0010] ITU-T VCEG(ITU-T Video Coding Expert Group) 및 ISO/IEC MPEG(ISO/IEC Moving Picture Expert Group)의 JVET(Joint Video Experts Team)는 현재 다용도 비디오 코딩(VVC/H.266) 표준을 개발하고 있다. VVC 표준은 그의 이전 버전인 고효율 비디오 코딩(HEVC/H.265) 표준의 압축 효율성을 두 배로 높이는 것을 목표로 한다. 다시 말해, VVC의 목표는 대역폭의 절반을 사용하여 HEVC/H.265와 동일한 주관적 품질을 달성하는 것이다.
- [0011] 대역폭의 절반을 사용하여 HEVC/H.265와 동일한 주관적 품질을 달성하기 위해, JVET는 공동 탐색 모델(joint exploration model; JEM) 참조 소프트웨어를 사용하여 HEVC를 넘는 기술을 개발하고 있다. 코딩 기술이 JEM에 통합됨에 따라, JEM은 HEVC보다 상당히 더 높은 코딩 성능을 달성하였다.
- [0012] VVC 표준은 최근에 개발되었고, 더 나은 압축 성능을 제공하는 더 많은 코딩 기술을 계속 포함하고 있다. VVC는 HEVC, H.264/AVC, MPEG2, H.263 등과 같은 최신 비디오 압축 표준에서 사용된 것과 동일한 하이브리드 비디오 코딩 시스템에 기초한다.
- [0013] 비디오는 시각적 정보를 저장하기 위해 시간적인 시퀀스로 배열된 정적 픽처(또는 "프레임")의 세트이다. 비디오 캡처 디바이스(예컨대, 카메라)는 이 픽처를 시간적인 시퀀스로 캡처하고 저장하는데 사용될 수 있고, 비디오 재생 디바이스(예컨대, 디스플레이의 기능을 갖는 텔레비전, 컴퓨터, 스마트폰, 태블릿 컴퓨터, 비디오 플레이어 또는 임의의 최종-사용자 단말기)는 이러한 픽처를 시간적인 시퀀스로 디스플레이하기 위해 사용될 수 있다. 또한, 일부 애플리케이션에서, 비디오 캡처링 디바이스는 감시, 회의 또는 실시간 방송과 같이 캡처된 비디오를 비디오 재생 디바이스(예컨대, 모니터를 갖는 컴퓨터)에 실시간으로 송신할 수 있다.
- [0014] 이러한 애플리케이션에 의해 필요한 저장 공간 및 송신 대역폭을 감소시키기 위해, 비디오는 저장 또는 송신 이전에 압축되고, 디스플레이 전에 압축 해제될 수 있다. 압축 및 압축 해제는 프로세서(예컨대, 일반 컴퓨터의 프로세서) 또는 특수 하드웨어에 의해 실행되는 소프트웨어에 의해 구현될 수 있다. 압축을 위한 모듈은 일반적으로 "인코더"로 지칭되고, 압축 해제를 위한 모듈은 일반적으로 "디코더"로 지칭된다. 인코더 및 디코더를 집합적으로 "코덱(codec)"으로 지칭될 수 있다. 인코더 및 디코더는 임의의 다양한 적합한 하드웨어, 소프트웨어, 또는 이들의 조합 중 임의의 것으로 구현될 수 있다. 예컨대, 인코더 및 디코더의 하드웨어 구현은 하나 이상의 마이크로프로세서, 디지털 신호 프로세서(DSP), 애플리케이션-특정 집적 회로(application-specific integrated circuits, ASIC), 필드 프로그래밍 가능 게이트 어레이(Field-Programmable Gate Array, FPGA), 이산 로직 또는 이의 임의의 조합과 같은 회로를 포함할 수 있다. 인코더 및 디코더의 소프트웨어 구현은 컴퓨터

판독 가능 매체에 고정된 프로그램 코드, 컴퓨터-실행가능 명령어, 펌웨어 또는 임의의 적합한 컴퓨터 구현 알고리즘 또는 프로세스를 포함할 수 있다. 비디오 압축 및 압축 해제에는 MPEG-1, MPEG-2, MPEG-4, H.26x 시리즈 등과 같은 다양한 알고리즘 또는 표준에 의해 구현될 수 있다. 일부 애플리케이션에서, 코덱은 제1 코딩 표준으로부터 비디오를 압축 해제하고 제2 코딩 표준을 사용하여 압축 해제된 비디오를 재압축할 수 있으며, 이 경우 코덱은 "트랜스코더(transcoder)"로 지칭될 수 있다.

- [0015] 비디오 인코딩 프로세스는 픽처를 재구성하는 데 사용될 수 있는 유용한 정보를 식별하고 유지할 수 있고, 재구성에 중요하지 않은 정보를 폐기할 수 있다. 폐기되고 중요하지 않은 정보가 완전히 재구성될 수 없는 경우, 이러한 인코딩 프로세스는 "손실이 있는 것(lossy)"으로 지칭될 수 있다. 그렇지 않으면, 이는 "무손실"로 지칭될 수 있다. 대부분의 인코딩 프로세스는 손실이 있고, 이는 필요한 저장 공간 및 송신 대역폭을 감소시키기 위한 트레이드오프이다.
- [0016] 인코딩되는 픽처(현재 픽처로 지칭됨)의 유용한 정보는 참조 픽처(예컨대, 이전에 인코딩되거나 재구성된 픽처)에 대한 변경을 포함할 수 있다. 이러한 변경은 픽셀의 위치 변경, 광도 변경 또는 색상 변경을 포함할 수 있으며, 그 중 위치 변경이 주로 관련된다. 객체를 나타내는 픽셀의 그룹의 위치 변경은 참조 픽처와 현재 픽처 사이의 객체의 움직임을 반영할 수 있다.
- [0017] 다른 픽처를 참조하지 않고 코딩된 픽처(즉, 자신이 그 자신의 참조 픽처인 것)는 "I-픽처"로 지칭된다. 참조 픽처로서 이전 픽처를 사용하여 코딩된 픽처는 "P-픽처"로 지칭된다. 참조 픽처로서 이전 픽처 및 미래 픽처 둘 모두를 사용하여 코딩된 픽처(즉, 참조가 "양방향성"임)는 "B-픽처"로 지칭된다.
- [0018] 도 1은 본 개시의 일부 실시예에 따른 예시적인 비디오 시퀀스(100)의 구조를 도시한다. 비디오 시퀀스(100)는 실시간 비디오 또는 캡처되고 보관되는 비디오일 수 있다. 비디오(100)는 실제 비디오, 컴퓨터-생성된 비디오(예컨대, 컴퓨터 게임 비디오)또는 이들의 조합(예컨대, 증강-현실 효과를 갖는 실제 비디오)일 수 있다. 비디오 시퀀스(100)는 비디오 캡처 디바이스(예컨대, 카메라), 이전에 캡처된 비디오를 포함하는 비디오 아카이브(예컨대, 저장 디바이스에 저장된 비디오 파일), 또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스(예컨대, 비디오 브로드캐스트 트랜시버)로부터 입력될 수 있다.
- [0019] 도 1에 도시된 바와 같이, 비디오 시퀀스(100)는 픽처(102, 104, 106 및 108)를 포함하는, 시간선을 따라 시간적으로 배열된 일련의 픽처를 포함할 수 있다. 픽처(102 내지 106)는 연속적이고, 픽처(106)와 픽처(108) 사이에 더 많은 픽처가 있다. 도 1에서, 픽처(102)는 I-픽처이고, 이의 참조 픽처는 픽처(102) 그 자체이다. 픽처(104)는 P-픽처이고, 이의 참조 픽처는 화살표로 나타난 바와 같이 픽처(102)이다. 픽처(106)는 B-픽처이고, 이의 참조 픽처는 화살표로 나타난 바와 같이 픽처(104 및 108)이다. 일부 실시예에서, 픽처(예컨대, 픽처(104))의 참조 픽처는 픽처 직전 또는 직후가 아닐 수 있다. 예컨대, 픽처(104)의 참조 픽처는 픽처(102)에 선행하는 픽처일 수 있다. 픽처(102 내지 106)의 참조 픽처는 단지 예시이며, 본 개시는 도 1에 도시된 예시로서 참조 픽처의 실시예를 제한하지 않는다는 것이 유의되어야 한다.
- [0020] 전형적으로, 비디오 코덱은 이러한 태스크(task)의 연산 복잡성에 기인하여 전체의 픽처를 동시에 인코딩하거나 또는 디코딩하지 않는다. 오히려, 이는 픽처를 기본 세그먼트로 분할할 수 있고, 픽처를 세그먼트별로 인코딩하거나 또는 디코딩할 수 있다. 이러한 기본 세그먼트는 본 개시에서 기본 처리 유닛(basic processing units; BPU)으로 지칭된다. 예컨대, 도 1에서의 구조(110)는 비디오 시퀀스(100)의 픽처(예컨대, 픽처(102 내지 108) 중 임의의 것)의 예시적인 구조를 도시한다. 구조(110)에서, 픽처는 4X4 기본 처리 유닛으로 분할되고, 이의 경계는 점선으로 도시된다. 일부 실시예에서, 기본 처리 유닛은 일부 비디오 코딩 표준(예컨대, MPEG 패밀리, H.261, H.263, 또는 H.264/AVC)에서 "매크로블록"으로 지칭될 수 있거나, 일부 다른 비디오 코딩 표준(예컨대, H.265/HEVC 또는 H.266/VVC)에서 "코딩 트리 유닛("CTU")"으로 지칭될 수 있다. 기본 처리 유닛은 128×128, 64×64, 32×32, 16×16, 4×8, 16×32와 같은 픽처에서의 가변적인 크기, 또는 픽셀의 어느 임의의 형태 및 크기를 가질 수 있다. 기본 처리 유닛의 크기 및 형태는 기본 처리 유닛에서 유지될 세부사항의 레벨 및 코딩 효율의 밸런스에 기초하여 픽처에 대해 선택될 수 있다.
- [0021] 기본 처리 유닛은 컴퓨터 메모리에 (예컨대, 비디오 프레임 버퍼에) 저장된 상이한 타입의 비디오 데이터의 그룹을 포함할 수 있는 논리적 유닛일 수 있다. 예컨대, 색상 픽처의 기본 처리 유닛은 무색의 밝기 정보를 나타내는 루마 성분(Y), 색상 정보를 나타내는 하나 이상의 크로마 성분(예컨대, Cb 및 Cr) 및 연관된 신텍스 요소를 포함할 수 있고, 여기서 루마 및 크로마 성분은 동일한 크기의 기본 처리 유닛을 가질 수 있다. 루마 및 크로마 성분은 일부 비디오 코딩 표준(예컨대, H.265/HEVC 또는 H.266/VVC)에서 "코딩 트리 블록("CTB)"으로 지칭될 수 있다. 기본 처리 유닛에 대해 수행된 임의의 동작은 그의 루마 및 크로마 성분의 각각에 대해 반복적

으로 수행될 수 있다.

- [0022] 비디오 코딩은 다수의 동작 스테이지를 가지며, 이의 예시는 도 2a-2b 및 3a-3b에 도시된다. 각각의 스테이지에 대해, 기본 처리 유닛의 크기는 처리하기에 여전히 너무 클 수 있으며, 따라서 본 개시에서 "기본 처리 서브-유닛"으로 지칭되는 세그먼트로 더 분할될 수 있다. 일부 실시예에서, 기본 처리 서브-유닛은 일부 비디오 코딩 표준(예컨대, MPEG 패밀리, H.261, H.263, 또는 H.264/AVC)에서 "블록"으로 지칭될 수 있거나, 일부 다른 비디오 코딩 표준(예컨대, H.265/HEVC 또는 H.266/VVC)에서 "코딩 유닛"("CU")으로 지칭될 수 있다. 기본 처리 서브-유닛은 기본 처리 유닛과 동일하거나 더 작은 크기를 가질 수 있다. 기본 처리 유닛과 유사하게, 기본 처리 서브-유닛은 또한, 논리적 유닛이며, 이는 컴퓨터 메모리에 (예컨대, 비디오 프레임 버퍼에) 저장된 상이한 타입의 비디오 데이터(예컨대, Y, Cb, Cr 및 연관된 신텍스 요소)의 그룹을 포함할 수 있다. 기본 처리 서브-유닛에 대해 수행된 임의의 동작은 그의 루마 및 크로마 성분의 각각에 대해 반복적으로 수행될 수 있다. 이러한 분할은 처리 요구에 의존하는 추가적인 레벨로 수행될 수 있다는 것이 유의되어야 한다. 또한, 상이한 스테이지가 상이한 체계를 사용하여 기본 처리 유닛을 분할할 수 있다는 것이 유의되어야 한다.
- [0023] 예컨대, 모드 결정 스테이지(그의 예시가 도 2b에 도시됨)에서, 인코더는 기본 처리 유닛에 대해 어떤 예측 모드(예컨대, 인트라-픽처 예측 또는 인터-픽처 예측)를 사용할지를 결정할 수 있으며, 이러한 기본 처리 유닛은 이러한 결정을 하기엔 너무 클 수 있다. 인코더는 기본 처리 유닛을 다수의 기본 처리 서브-유닛(예컨대, H.265/HEVC 또는 H.266/VVC에서와 같이 CU)으로 분할할 수 있으며, 각각의 개별적인 기본 처리 서브-유닛에 대해 예측 타입을 결정할 수 있다.
- [0024] 다른 예시로서, 예측 스테이지(그의 예시가 도 2a-2b에 도시됨)에서, 인코더는 기본 처리 서브-유닛(예컨대, CU)의 레벨에서 예측 동작을 수행할 수 있다. 하지만, 일부 경우에서, 기본 처리 서브-유닛은 처리하기에 여전히 너무 클 수 있다. 인코더는 기본 처리 서브-유닛을 예측 동작이 수행될 수 있는 레벨의 더 작은 세그먼트(예컨대, H.265/HEVC 또는 H.266/VVC에서 "예측 블록" 또는 "PB"로 지칭됨)로 더 분할할 수 있다.
- [0025] 다른 예시로서, 변환 스테이지(그의 예시가 도 2a-2b에 도시됨)에서, 인코더는 잔차 기본 처리 서브-유닛(예컨대, CU)에 대한 변환 동작을 수행할 수 있다. 하지만, 일부 경우에서, 기본 처리 서브-유닛은 처리하기에 여전히 너무 클 수 있다. 인코더는 기본 처리 서브-유닛을 예측 동작이 수행될 수 있는 레벨의 더 작은 세그먼트(예컨대, H.265/HEVC 또는 H.266/VVC에서 "변환 블록" 또는 "TB"로 지칭됨)로 더 분할할 수 있다. 동일한 기본 처리 서브-유닛의 분할 체계는 예측 스테이지 및 변환 스테이지에서 상이할 수 있다는 것이 유의되어야 한다. 예컨대, H.265/HEVC 또는 H.266/VVC에서, 동일한 CU의 예측 블록 및 변환 블록은 상이한 크기 및 개수를 가질 수 있다.
- [0026] 도 1의 구조(110)에서, 기본 처리 유닛(112)은 그의 경계가 점선으로 도시된, 3X3 기본 처리 서브-유닛으로 더 분할된다. 동일한 픽처의 상이한 기본 처리 유닛이 상이한 체계의 기본 처리 서브-유닛으로 분할될 수 있다.
- [0027] 일부 구현에서, 비디오 인코딩 및 디코딩에 대한 병렬 처리 및 오류 복원의 능력을 제공하기 위해, 픽처는 픽처의 영역에 대해, 인코딩 또는 디코딩 프로세스가 픽처의 임의의 다른 영역으로부터의 정보에 의존하지 않도록, 처리를 위한 영역으로 분할될 수 있다. 다시 말해, 픽처의 각각의 영역은 독립적으로 처리될 수 있다. 이렇게 함으로써, 코덱은 픽처의 상이한 영역을 병렬로 처리할 수 있으며, 따라서 코딩 효율을 증가시킨다. 또한, 영역의 데이터가 처리에서 훼손되거나 또는 네트워크 송신에서 분실될 때, 코덱은 훼손되거나 또는 분실된 데이터에 대한 의존(reliance) 없이, 동일한 픽처의 다른 영역을 올바르게 인코딩 또는 디코딩할 수 있으며, 따라서 오류 복원의 능력을 제공한다. 일부 비디오 코딩 표준에서, 픽처는 상이한 타입의 영역으로 분할될 수 있다. 예컨대, H.265/HEVC 및 H.266/VVC는 두 개의 타입의 영역: "슬라이스" 및 "타일"을 제공한다. 또한, 비디오 시퀀스(100)의 상이한 픽처가 픽처를 영역으로 분할하기 위한 상이한 파티션 체계를 가질 수 있다는 것이 유의되어야 한다.
- [0028] 예컨대, 도 1에서, 구조(110)는 그의 경계가 구조(110) 내에서 실선으로 도시된 세 개의 영역(114, 116 및 118)으로 분할된다. 영역(114)은 네 개의 기본 처리 유닛을 포함한다. 영역(116 및 118)의 각각은 여섯 개의 기본 처리 유닛을 포함한다. 도 1에서 구조(110)의 기본 처리 유닛, 기본 처리 서브-유닛 및 영역은 단지 예시이며, 본 개시는 이의 실시예를 제한하지 않는다는 것이 유의되어야 한다.
- [0029] 도 2a는 본 개시의 실시예와 일치하는, 예시적인 인코딩 프로세스(200A)의 개략도를 도시한다. 예컨대, 인코딩 프로세스(200A)는 인코더에 의해 수행될 수 있다. 도 2a에 도시된 바와 같이, 인코더는 프로세스(200A)에 따라 비디오 시퀀스(202)를 비디오 비트스트림(228)으로 인코딩할 수 있다. 도 1에서의 비디오 시퀀스(100)와 유사

하게, 비디오 시퀀스(202)는 시간적인 순서로 배열된 픽처("원본 픽처"로 지칭됨)의 세트를 포함할 수 있다. 도 1에서의 구조(110)와 유사하게, 비디오 시퀀스(202)의 각각의 원본 픽처는 인코더에 의해 기본 처리 유닛, 기본 처리 서브-유닛 또는 처리를 위한 영역으로 분할될 수 있다. 일부 실시예에서, 인코더는 비디오 시퀀스(202)의 각각의 원본 픽처에 대한 기본 처리 유닛의 레벨에서 프로세스(200A)를 수행할 수 있다. 예컨대, 인코더는 프로세스(200A)를 반복 방식으로 수행할 수 있으며, 여기서 인코더는 프로세스(200A)의 하나의 반복으로 기본 처리 유닛을 인코딩할 수 있다. 일부 실시예에서, 인코더는 비디오 시퀀스(202)의 각각의 원본 픽처의 영역(예컨대, 영역(114-118))에 대해 프로세스(200A)를 병렬로 수행할 수 있다.

[0030] 도 2a에서, 인코더는 예측 데이터(206) 및 예측된 BPU(208)를 생성하기 위해 비디오 시퀀스(202)의 원본 픽처의 기본 처리 유닛("원본 BPU"로 지칭됨)을 예측 스테이지(204)로 공급할 수 있다. 인코더는 잔차 BPU(210)를 생성하기 위해 원본 BPU로부터 예측된 BPU(208)를 감산할 수 있다. 인코더는 양자화된 변환 계수(216)를 생성하기 위해, 잔차 BPU(210)를 변환 스테이지(212) 및 양자화 스테이지(214)에 공급할 수 있다. 인코더는 비디오 비트스트림(228)을 생성하기 위해, 예측 데이터(206) 및 양자화된 변환 계수(216)를 이진 코딩 스테이지(226)에 공급할 수 있다. 구성요소(202, 204, 206, 208, 210, 212, 214, 216, 226 및 228)는 "순방향 경로(forward path)"로서 지칭될 수 있다. 프로세스(200A) 동안, 양자화 스테이지(214) 이후에, 인코더는 재구성된 잔차 BPU(222)를 생성하기 위해, 양자화된 변환 계수(216)를 역양자화 스테이지(218) 및 역변환 스테이지(220)에 공급할 수 있다. 인코더는 프로세스(200A)의 다음 반복을 위해 예측 스테이지(204)에서 사용되는 예측 참조(224)를 생성하기 위해, 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있다. 프로세스(200A)의 구성요소(218, 220, 222 및 224)는 "재구성 경로"로 지칭될 수 있다. 재구성 경로는 인코더 및 디코더 둘 모두가 예측을 위해 동일한 참조 데이터를 사용하는 것을 보장하도록 사용될 수 있다.

[0031] 인코더는 (순방향 경로에서) 원본 픽처의 각각의 원본 BPU를 인코딩하고, (재구성 경로에서) 원본 픽처의 다음 원본 BPU를 인코딩하기 위한 예측된 참조(224)를 생성하기 위해 프로세스(200A)를 반복적으로 수행할 수 있다. 원본 픽처의 모든 원본 BPU를 인코딩한 이후에, 인코더는 비디오 시퀀스(202)에서 다음 픽처를 인코딩하도록 진행할 수 있다.

[0032] 프로세스(200A)를 참조하면, 인코더는 비디오 캡처링 디바이스(예컨대, 카메라)에 의해 생성되는 비디오 시퀀스(202)를 수신할 수 있다. 본원에서 사용된 "수신하다"라는 용어는 수신, 입력, 취득, 리트리브(retrieve), 획득, 판독, 액세스 또는 데이터를 입력하기 위한 임의의 방식에서의 임의의 동작을 지칭할 수 있다.

[0033] 현재 반복에서의 예측 스테이지(204)에서, 인코더는 원본 BPU 및 예측 참조(224)를 수신할 수 있고, 예측 데이터(206) 및 예측된 BPU(208)를 생성하기 위해 예측 동작을 수행할 수 있다. 예측 참조(224)는 프로세스(200A)의 이전의 반복의 재구성 경로로부터 생성될 수 있다. 예측 스테이지(204)의 목적은 예측 데이터(206) 및 예측 참조(224)로부터 원본 BPU를 예측된 BPU(208)로서 재구성하기 위해 사용될 수 있는 예측 데이터(206)를 추출함으로써 정보 리던던시(information redundancy)를 감소시키는 것이다.

[0034] 이상적으로, 예측된 BPU(208)는 원본 BPU와 동일할 수 있다. 하지만, 비-이상적 예측 및 재구성 동작에 기인하여, 예측된 BPU(208)는 일반적으로 원본 BPU와는 약간 상이하다. 이러한 차이를 기록하기 위해, 예측된 BPU(208)를 생성한 이후에, 인코더는 잔차 BPU(210)를 생성하기 위해, 원본 BPU로부터 이를 감산할 수 있다. 예컨대, 인코더는 원본 BPU의 대응하는 픽셀의 값으로부터 예측된 BPU(208)의 픽셀의 값(예컨대, 그레이스케일 값 또는 RGB 값)을 감산할 수 있다. 잔차 BPU(210)의 각각의 픽셀은 원본 BPU 및 예측된 BPU(208)의 대응하는 픽셀 사이에서 이러한 감산의 결과로서 잔차 값을 가질 수 있다. 원본 BPU와 비교하여, 예측 데이터(206) 및 잔차 BPU(210)는 더 적은 수의 비트를 가질 수 있지만, 이들은 현저한 품질 저하 없이 원본 BPU를 재구성하는데 사용될 수 있다. 따라서, 원본 BPU가 압축된다.

[0035] 잔차 BPU(210)를 더 압축하기 위해, 변환 스테이지(212)에서, 인코더는 이를 2차원 "기본 패턴" - 각각의 기본 패턴은 "변환 계수와 연관된 - 으로 분해함으로써 잔차 BPU(210)의 공간 리던던시를 저감할 수 있다. 기본 패턴은 동일한 크기(예컨대, 잔차 BPU(210)의 크기)를 가질 수 있다. 각각의 기본 패턴은 잔차 BPU(210)의 변동 주파수(variation frequency)(예컨대, 밝기 변동의 주파수) 성분을 나타낼 수 있다. 기본 패턴 중 어느 것도 임의의 다른 기본 패턴의 임의의 조합(예컨대, 선형 조합)으로부터 재생성될 수 없다. 다시 말해, 분해는 잔차 BPU(210)의 변동을 주파수 도메인으로 분해할 수 있다. 이러한 분해는 함수의 이산 푸리에 변환과 유사하며, 여기서 기본 패턴은 이산 푸리에 변환의 기본 함수(예컨대, 삼각 함수)와 유사하고 변환 계수는 기본 함수와 연관된 계수와 유사하다.

[0036] 상이한 변환 알고리즘이 상이한 기본 패턴을 사용할 수 있다. 다양한 변환 알고리즘은 예컨대, 이산 코사인 변

환, 이산 사인 변환 등과 같은 변환 스테이지(212)에서 사용될 수 있다. 변환 스테이지(212)에서의 변환은 역으로 이루어질 수 있다(invertible). 즉, 인코더는 변환의 역동작("역변환"으로 지칭됨)에 의해 잔차 BPU(210)를 복원할 수 있다. 예컨대, 잔차 BPU(210)의 픽셀을 복원하기 위해, 역변환은 기본 패턴의 대응하는 픽셀의 값을 개개의 연관된 계수를 곱하고, 그 결과 값을 더하여 가중합을 생성할 수 있다. 비디오 코딩 표준에 대해, 인코더 및 디코더 둘 모두는 동일한 변환 알고리즘(따라서, 동일한 기본 패턴)을 사용할 수 있다. 따라서, 인코더는 변환 계수만을 기록할 수 있고, 이로부터 디코더는 인코더로부터 기본 패턴을 수신하지 않으면서, 잔차 BPU(210)를 재구성할 수 있다. 잔차 BPU(210)와 비교하여, 변환 계수는 더 적은 수의 비트를 가질 수 있지만, 이들은 현저한 품질 저하 없이 잔차 BPU(210)를 재구성하는데 사용될 수 있다. 따라서, 잔차 BPU(210)가 더 압축된다.

[0037] 인코더는 양자화 스테이지(214)에서 변환 계수를 더 압축할 수 있다. 변환 프로세스에서, 상이한 기본 패턴이 상이한 변동 주파수(예컨대, 밝기 변동 주파수)를 나타낼 수 있다. 인간의 눈은 일반적으로 낮은-주파수 변동을 더 잘 인식하기 때문에, 인코더는 디코딩에서 현저한 품질 저하를 초래하지 않으면서 높은-주파수 변동의 정보를 무시할 수 있다. 예컨대, 양자화 스테이지(214)에서, 인코더는 각각의 변환 계수를 정수 값("양자화 파라미터"로 지칭됨)으로 나누고, 몫을 그의 가장 가까운 정수로 반올림함으로써, 양자화된 변환 계수(216)를 생성할 수 있다. 이러한 동작 이후에, 고주파수 기본 패턴의 일부 변환 계수는 0으로 변환될 수 있고, 저주파수 기본 패턴의 변환 계수는 더 작은 정수로 변환될 수 있다. 인코더는 0-값 양자화된 변환 계수(216)를 무시할 수 있으며, 이에 의해 변환 계수는 더 압축된다. 또한, 양자화 프로세스는 역으로 이루어질 수 있고, 여기서 양자화된 변환 계수(216)는 양자화의 역동작("역양자화"로 지칭됨)에서 변환 계수로 재구성될 수 있다.

[0038] 인코더가 반올림 동작에서 이러한 나눗셈의 나머지를 무시하기 때문에, 양자화 스테이지(214)는 손실이 있을 수 있다. 전형적으로, 양자화 스테이지(214)는 프로세스(200A)의 대부분의 정보 손실에 기여할 수 있다. 정보 손실이 크면 클수록, 양자화된 변환 계수(216)가 더 적은 비트를 필요로 할 수 있다. 상이한 레벨의 정보 손실을 획득하기 위해, 인코더는 양자화 파라미터의 상이한 값 또는 양자화 프로세스의 임의의 다른 파라미터를 사용할 수 있다.

[0039] 이진 코딩 스테이지(226)에서, 인코더는 예컨대, 엔트로피 코딩, 가변 길이 코딩, 산술 코딩, 허프만 코딩(Huffman coding), 컨텍스트-적응적 이진 산술 코딩(context-adaptive binary arithmetic coding) 또는 임의의 다른 무손실 또는 손실 압축 알고리즘과 같은 이진 코딩 기술을 사용하여 예측 데이터(206) 및 양자화된 변환 계수(216)를 인코딩할 수 있다. 일부 실시예에서, 예측 데이터(206) 및 양자화된 변환 계수(216) 이외에, 인코더는 예컨대, 예측 스테이지(204)에서 사용되는 예측 모드, 예측 동작의 파라미터, 변환 스테이지(212)에서의 변환 타입, 양자화 프로세스의 파라미터(예컨대, 양자화 파라미터), 인코더 제어 파라미터(예컨대, 비트레이트 제어 파라미터) 등과 같은 다른 정보를 이진 코딩 스테이지(226)에서 인코딩할 수 있다. 인코더는 비디오 비트스트림(228)을 생성하기 위해 이진 코딩 스테이지(226)의 출력 데이터를 사용할 수 있다. 일부 실시예에서, 비디오 비트스트림(228)은 네트워크 송신을 위해 더 패킷화될 수 있다.

[0040] 프로세스(200A)의 재구성 경로를 참고하면, 역양자화 스테이지(218)에서, 인코더는 재구성된 변환 계수를 생성하기 위해 양자화된 변환 계수(216)에 역양자화를 수행할 수 있다. 역변환 스테이지(220)에서, 인코더는 재구성된 변환 계수에 기초하여, 재구성된 잔차 BPU(222)를 생성할 수 있다. 인코더는 프로세스(200A)의 다음 반복에서 사용될 예측 참조(224)를 생성하기 위해, 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있다.

[0041] 프로세스(200A)의 다른 변형은 비디오 시퀀스(202)를 인코딩하기 위해 사용될 수 있다는 것이 유의되어야 한다. 일부 실시예에서, 프로세스(200A)의 스테이지는 인코더에 의해 상이한 순서로 수행될 수 있다. 일부 실시예에서, 프로세스(200A)의 하나 이상의 스테이지는 단일 스테이지로 결합될 수 있다. 일부 실시예에서, 프로세스(200A)의 단일 스테이지는 다수의 스테이지로 분할될 수 있다. 예컨대, 변환 스테이지(212) 및 양자화 스테이지(214)가 단일 스테이지로 결합될 수 있다. 일부 실시예에서, 프로세스(200A)는 추가적인 스테이지를 포함할 수 있다. 일부 실시예에서, 프로세스(200A)는 도 2a에서의 하나 이상의 스테이지를 생략할 수 있다.

[0042] 도 2b는 본 개시의 실시예와 일치하는, 다른 예시적인 인코딩 프로세스(200B)의 개략도를 도시한다. 프로세스(200B)는 프로세스(200A)로부터 수정될 수 있다. 예컨대, 프로세스(200B)는 하이브리드 비디오 인코딩 표준(예컨대, H.26x 시리즈)을 따르는 인코더에 의해 사용될 수 있다. 프로세스(200A)에 비해, 프로세스(200B)의 순방향 경로는 모드 결정 스테이지(230)를 추가적으로 포함하고, 예측 스테이지(204)를 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)로 분할한다. 프로세스(200B)의 재구성 경로는 루프 필터 스테이지(232) 및 버퍼(234)를 추가적으로 포함한다.

- [0043] 일반적으로, 예측 기술은 두 개의 타입: 공간 예측 및 시간 예측으로 카테고리화될 수 있다. 공간 예측(예컨대, 인트라-픽처 예측 또는 "인트라 예측")은 현재 BPU를 예측하기 위해, 동일한 픽처에서 하나 이상의 이미 코딩된 이웃하는 BPU로부터의 픽셀을 사용할 수 있다. 즉, 공간 예측에서의 예측 참조(224)는 이웃하는 BPU를 포함할 수 있다. 공간 예측은 픽처의 내재적인 공간 리던던시를 감소시킬 수 있다. 시간 예측(예컨대, 인터-픽처 예측 또는 "인터 예측")은 현재 BPU를 예측하기 위해 하나 이상의 이미 코딩된 픽처로부터의 영역을 사용할 수 있다. 즉, 시간 예측에서의 예측 참조(224)는 코딩된 픽처를 포함할 수 있다. 시간 예측은 픽처의 내재적인 시간 리던던시를 감소시킬 수 있다.
- [0044] 프로세스(200B)를 참조하면, 순방향 경로에서, 인코더는 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)에서 예측 동작을 수행한다. 예컨대, 공간 예측 스테이지(2042)에서, 인코더는 인트라 예측을 수행할 수 있다. 인코딩되는 픽처의 원본 BPU에 대해, 예측 참조(224)는 동일한 픽처에서 (순방향 경로에서) 인코딩되고 (재구성 경로에서) 재구성되는 하나 이상의 이웃하는 BPU를 포함할 수 있다. 인코더는 이웃하는 BPU를 외삽함으로써 예측된 BPU(208)를 생성할 수 있다. 외삽 기법은 예컨대, 선형 외삽(linear extrapolation) 또는 내삽(interpolation), 다항식 외삽 또는 내삽 등을 포함할 수 있다. 일부 실시예에서, 인코더는 가령, 예측된 BPU(208)의 각각의 픽셀에 대해 대응하는 픽셀의 값을 외삽함으로써, 픽셀 레벨에서 외삽을 수행할 수 있다. 외삽을 위해 사용된 이웃하는 BPU는 (예컨대, 원본 BPU의 상부에서) 수직 방향, (예컨대, 원본 BPU의 좌측에서) 수평 방향, (예컨대, 원본 BPU의 좌측-하단, 우측-하단, 좌측-상단 또는 우측-상단에서) 대각선 방향, 또는 사용된 비디오 코딩 표준에서 정의된 임의의 방향에서와 같은 다양한 방향으로부터 원본 BPU에 대해 위치될 수 있다. 인트라 예측에 대해, 예측 데이터(206)는 예컨대, 원본 BPU에 대한 사용된 이웃하는 BPU의 위치(예컨대, 좌표), 사용된 이웃하는 BPU의 크기, 외삽의 파라미터, 사용된 이웃하는 BPU의 방향 등을 포함할 수 있다.
- [0045] 다른 예에 대해, 시간 예측 스테이지(2044)에서, 인코더는 인터 예측을 수행할 수 있다. 현재 픽처의 원본 BPU에 대해, 예측 참조(224)는 (순방향 경로에서) 인코딩되고 (재구성된 경로에서) 재구성된 하나 이상의 픽처("참조 픽처"로 지칭됨)를 포함할 수 있다. 일부 실시예에서, 참조 픽처는 BPU별로 인코딩되고 재구성될 수 있다. 예컨대, 인코더는 재구성된 BPU를 생성하기 위해, 재구성된 참조 BPU(222)를 예측된 BPU(208)에 더할 수 있다. 동일한 픽처의 모든 재구성된 BPU가 생성될 때, 인코더는 재구성된 픽처를 참조 픽처로서 생성할 수 있다. 인코더는 참조 픽처의 범주("검색 윈도우"로 지칭됨)에서 매칭 영역을 검색하기 위해, "움직임 추정"의 동작을 수행할 수 있다. 참조 픽처에서 검색 윈도우의 위치는 현재 픽처에서 원본 BPU의 위치에 기초하여 결정될 수 있다. 예컨대, 검색 윈도우는 현재 픽처에서 원본 BPU와 참조 픽처에서 동일한 좌표를 갖는 위치에 중심이 맞춰질 수 있고, 미리 결정된 거리에 대해 확장될 수 있다. 인코더가 검색 윈도우에서 원본 BPU와 유사한 영역을 (예컨대, 픽셀-순환 알고리즘(pel-recursive algorithm), 블록-매칭 알고리즘 등을 사용함으로써) 식별할 때, 인코더는 이러한 영역을 매칭 영역으로서 결정할 수 있다. 매칭 영역은 원본 BPU로부터 상이한 치수(예컨대, 그보다 작거나, 이와 동일하거나, 그보다 크거나 또는 상이한 형태인)를 가질 수 있다. 참조 픽처 및 현재 픽처가 (예컨대, 도 1에 도시된 바와 같이) 시간선에서 시간적으로 분리되기 때문에, 시간이 지남에 따라, 매칭 영역이 원본 BPU의 위치로 "이동하는" 것으로 여겨질 수 있다. 인코더는 "움직임 벡터"로서 이러한 움직임의 방향 및 거리를 기록할 수 있다. (예컨대, 도 1에서의 픽처(106)와 같이) 다수의 참조 픽처가 사용될 때, 인코더는 매칭 영역을 검색하고, 각각의 참조 픽처에 대해 그의 연관된 움직임 벡터를 결정할 수 있다. 일부 실시예에서, 인코더는 개개의 매칭 참조 픽처의 매칭 영역의 픽셀값에 대해 가중치를 할당할 수 있다.
- [0046] 움직임 추정은 예컨대, 병진(translation), 회전, 주밍(zooming) 등과 같은 다양한 타입의 움직임을 식별하기 위해 사용될 수 있다. 인터 예측에 대해, 예측 데이터(206)는 예컨대, 매칭 영역의 위치(예컨대, 좌표), 매칭 영역과 관련된 움직임 벡터, 참조 픽처의 개수, 참조 픽처와 연관된 가중치 등을 포함할 수 있다.
- [0047] 예측된 BPU(208)를 생성하기 위해, 인코더는 "움직임 보상"의 동작을 수행할 수 있다. 움직임 보상은 예측 데이터(206)(예컨대, 움직임 벡터) 및 예측 참조(224)에 기초하여 예측된 BPU(208)를 재구성하는데 사용될 수 있다. 예컨대, 인코더는 움직임 벡터에 따라 참조 픽처의 매칭 영역을 이동시킬 수 있으며, 여기서 인코더는 현재 픽처의 원본 BPU를 예측할 수 있다. 다수의 참조 픽처가 (예컨대, 도 1에서의 픽처로서) 사용될 때, 인코더는 매칭 영역의 각각의 움직임 벡터 및 평균 픽셀 값에 따라 참조 픽처의 매칭 영역을 이동시킬 수 있다. 일부 실시예에서, 인코더가 각각의 매칭 참조 픽처의 매칭 영역의 픽셀값에 가중치를 할당한 경우, 인코더는 이동된 매칭 영역의 픽셀값의 가중치 합을 더할 수 있다.
- [0048] 일부 실시예에서, 인터 예측은 단방향 또는 양방향일 수 있다. 단방향 인터 예측은 현재 픽처에 대해 동일한 시간 방향으로 하나 이상의 참조 픽처를 사용할 수 있다. 예컨대, 도 1에서의 픽처(104)는 참조 픽처(즉, 픽처(102))가 픽처(104)에 선행하는 단방향 인터-예측된 픽처이다. 양방향 인터 예측은 현재 픽처에 대해 시간 방

향 둘 모두에서 하나 이상의 참조 픽처를 사용할 수 있다. 예컨대, 도 1에서의 픽처(106)는 참조 픽처(즉, 픽처(104 및 108))가 픽처(104)에 대해 시간 방향 둘 모두에 있는 양방향 인터-예측된 픽처이다.

[0049] 프로세스(200B)의 순방향 경로를 계속 참조하면, 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044) 이후에, 모드 결정 스테이지(230)에서, 인코더는 프로세스(200B)의 현재 반복에 대해 예측 모드(예컨대, 인트라 예측 또는 인터 예측 중 하나)를 선택할 수 있다. 예컨대, 인코더는 레이트-왜곡 최적화 기술(rate-distortion optimization technique)을 수행할 수 있으며, 여기서 인코더는 후보 예측 모드의 비트레이트 및 후보 예측 모드 하에서 재구성된 참조 픽처의 왜곡에 의존하여, 비용 함수의 값을 최소화하기 위해 예측 모드를 선택할 수 있다. 선택된 예측 모드에 의존하여, 인코더는 대응하는 예측된 BPU(208) 및 예측된 데이터(206)를 생성할 수 있다.

[0050] 프로세스(200B)의 재구성 경로에서, 순방향 경로에서 인트라 예측 모드가 선택된 경우, 예측 참조(224)(예컨대, 현재 픽처에서 인코딩되고 재구성된 현재 BPU)를 생성한 이후에, 인코더는 차후 사용을 위해(예컨대, 현재 픽처의 다음 BPU의 외삽을 위해), 예측 참조(224)를 공간 예측 스테이지(2042)로 직접적으로 공급할 수 있다. 순방향 경로에서 인터 예측 모드가 선택된 경우에, 예측 참조(224)(예컨대, 모든 BPU가 인코딩되고 재구성된 현재 픽처)를 생성한 이후에, 인코더는 인터 예측에 의해 도입된 왜곡(예컨대, 블로킹 아티팩트(blocking artifacts))를 감소시키거나 제거하기 위해 인코더가 예측 참조(224)에 루프 필터를 적용할 수 있는 루프 필터 스테이지(232)에 예측 참조(224)를 공급할 수 있다. 인코더는 예컨대, 디블로킹(deblocking), 샘플 적응적 오프셋, 적응적 루프 필터 등과 같은 다양한 루프 필터 기법을 루프 필터 스테이지(232)에 적용할 수 있다. 루프-필터링된 참조 픽처는 차후 사용을 위해(예컨대, 비디오 시퀀스(202)의 향후 픽처에 대한 인터-예측 참조 픽처로서 사용되기 위해) 버퍼(234)(또는 "디코딩된 픽처 버퍼")에 저장될 수 있다. 인코더는 시간 예측 스테이지(2044)에서 사용되도록 버퍼(234)에 하나 이상의 참조 픽처를 저장할 수 있다. 일부 실시예에서, 인코더는 양자화된 변환 계수(216), 예측 데이터(206) 및 다른 정보와 함께, 이진 코딩 스테이지(226)에서 루프 필터의 파라미터(예컨대, 루프 필터 강도)를 인코딩할 수 있다.

[0051] 도 3a는 본 개시의 실시예와 일치하는 예시적인 디코딩 프로세스(300A)의 개략도를 도시한다. 프로세스(300A)는 도 2a에서의 압축 프로세스(200A)에 대응하는 압축 해제 프로세스일 수 있다. 일부 실시예에서, 프로세스(300A)는 프로세스(200A)의 재구성 경로와 유사할 수 있다. 디코더는 프로세스(300A)에 따라 비디오 비트스트림(228)을 비디오 스트림(304)으로 디코딩할 수 있다. 비디오 스트림(304)은 비디오 시퀀스(202)와 매우 유사할 수 있다. 하지만, 압축 및 압축 해제 프로세스(예컨대, 도 2a-2b에서의 양자화 스테이지(214))에서의 정보 손실에 기인하여, 일반적으로 비디오 스트림(304)은 비디오 시퀀스(202)와 동일하지 않다. 도 2a-2b에서의 프로세스(200A 및 200B)와 유사하게, 디코더는 비디오 비트스트림(228)에서 인코딩된 각각의 픽처에 대해 기본 처리 유닛(BPU)의 레벨에서 프로세스(300A)를 수행할 수 있다. 예컨대, 디코더는 반복적인 방식으로 프로세스(300A)를 수행할 수 있으며, 여기서 디코더는 프로세스(300A)의 한 번의 반복으로 기본 처리 유닛을 디코딩할 수 있다. 일부 실시예에서, 디코더는 비디오 비트스트림(228)에서 인코딩된 각각의 픽처의 영역(예컨대, 영역(114 내지 118))에 대해 병렬로 프로세스(300A)를 수행할 수 있다.

[0052] 도 3a에서, 디코더는 인코딩된 픽처의 기본 처리 유닛("인코딩된 BPU"로 지칭됨)과 연관된 비디오 비트스트림(228)의 부분을 이진 디코딩 스테이지(302)로 공급할 수 있다. 이진 디코딩 스테이지(302)에서, 디코더는 그 부분을 예측 데이터(206) 및 양자화된 변환 계수(216)로 디코딩할 수 있다. 디코더는 재구성된 잔차 BPU(222)를 생성하기 위해 양자화된 변환 계수(216)를 역양자화 스테이지(218) 및 역변환 스테이지(220)에 공급할 수 있다. 디코더는 예측된 BPU(208)를 생성하기 위해 예측 데이터(206)를 예측 스테이지(204)로 공급할 수 있다. 디코더는 예측된 참조(224)를 생성하기 위해 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있다. 일부 실시예에서, 예측된 참조(224)는 버퍼(예컨대, 컴퓨터 메모리의 디코딩된 픽처 버퍼)에 저장될 수 있다. 디코더는 프로세스(300A)의 다음 반복에서 예측 동작을 수행하기 위해, 예측된 참조(224)를 예측 스테이지(204)에 공급할 수 있다.

[0053] 디코더는 인코딩된 픽처의 각각의 인코딩된 BPU를 디코딩하고, 인코딩된 픽처의 다음 인코딩된 BPU를 인코딩하기 위해 예측된 참조(224)를 생성하도록, 프로세스(300A)를 반복적으로 수행할 수 있다. 인코딩된 픽처의 모든 인코딩된 BPU를 디코딩한 이후에, 디코더는 디스플레이를 위해 픽처를 비디오 스트림(304)에 출력하고, 비디오 비트스트림(228)에서 다음 인코딩된 픽처를 디코딩하도록 진행할 수 있다.

[0054] 이진 디코딩 스테이지(302)에서, 디코더는 인코더에 의해 사용된 이진 코딩 기법(예컨대, 엔트로피 코딩, 가변 길이 코딩, 산술 코딩, 허프만 코딩, 컨텍스트-적응적 이진 산술 코딩 또는 임의의 다른 무손실 압축 알고리즘)

즘)의 역동작을 수행할 수 있다. 일부 실시예에서, 예측 데이터(206) 및 양자화된 변환 계수(216) 이외에, 디코더는 예컨대, 예측 모드, 예측 동작의 파라미터, 변환 타입, 양자화 프로세스의 파라미터(예컨대, 양자화 파라미터), 인코더 제어 파라미터(예컨대, 비트레이트 제어 파라미터) 등과 같은 다른 정보를 이진 디코딩 스테이지(302)에서 디코딩할 수 있다. 일부 실시예에서, 비디오 비트스트림(228)이 네트워크를 통해 패킷으로 송신되는 경우, 디코더는 이를 이진 디코딩 스테이지(302)에 공급하기 전에, 비디오 비트스트림(228)을 디패킷화(depacketize)할 수 있다.

[0055] 도 3b는 본 개시의 일부 실시예와 일치하는 다른 예시적인 디코딩 프로세스(300B)의 개략도를 도시한다. 프로세스(300B)는 프로세스(300A)로부터 수정될 수 있다. 예컨대, 프로세스(300B)는 하이브리드 비디오 코딩 표준(예컨대, H.26x 시리즈)에 따르는 디코더에 의해 사용될 수 있다. 프로세스(300A)와 비교하면, 프로세스(300B)는 예측 스테이지(204)를 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)로 추가적으로 분할하고, 루프 필터 스테이지(232) 및 버퍼(234)를 추가적으로 포함한다.

[0056] 프로세스(300B)에서, 디코딩되는 인코딩된 픽처("현재 픽처"로 지칭됨)의 인코딩된 기본 처리 유닛("현재 BPU"로 지칭됨)에 대해, 디코더에 의해 이진 디코딩 스테이지(302)로부터 디코딩된 예측 데이터(206)는 인코더에 의해 현재 BPU를 인코딩하는 데 어느 예측 모드가 사용되었는지에 의존하여, 다양한 타입의 데이터를 포함할 수 있다. 예컨대, 현재 BPU를 인코딩하기 위해, 인코더에 의해 인트라 예측이 사용된 경우, 예측 데이터(206)는 인트라 예측을 나타내는 예측 모드 표시자(예컨대, 플래그 값), 인트라 예측 동작의 파라미터 등을 포함할 수 있다. 인트라 예측 동작의 파라미터는 예컨대, 참조로서 사용된 하나 이상의 이웃하는 BPU의 위치(예컨대, 좌표), 이웃하는 BPU의 크기, 외삽의 파라미터, 원본 BPU에 대한 이웃하는 BPU의 방향 등을 포함할 수 있다. 다른 예에 대해, 현재 BPU를 인코딩하기 위해, 인코더에 의해 인터 예측이 사용된 경우, 예측 데이터(206)는 인터 예측을 나타내는 예측 모드 표시자(예컨대, 플래그 값), 인터 예측 동작의 파라미터 등을 포함할 수 있다. 인터 예측 동작의 파라미터는 예컨대, 현재 BPU와 연관된 참조 픽처의 개수, 참조 픽처와 각각 연관된 가중치, 각각의 참조 픽처에서의 하나 이상의 매칭 영역의 위치(예컨대, 좌표), 매칭 영역과 각각 연관된 하나 이상의 움직임 벡터 등을 포함할 수 있다.

[0057] 예측 모드 표시자에 기초하여, 디코더는 공간 예측 스테이지(2042)에서 공간 예측(예컨대, 인트라 예측)을 수행할지, 또는 시간 예측 스테이지(2044)에서 시간 예측(예컨대, 인터 예측)을 수행할지를 결정할 수 있다. 이러한 공간 예측 또는 시간 예측을 수행하는 것에 대한 세부사항이 도 2b에서 설명되고, 이하에서는 반복되지 않을 것이다. 이러한 공간 예측 또는 시간 예측을 수행한 이후에, 디코더는 예측된 BPU(208)를 생성할 수 있다. 도 3a에서 설명된 바와 같이, 디코더는 예측 참조(224)를 생성하기 위해 예측된 BPU(208) 및 재구성된 잔차 BPU(222)를 더한다.

[0058] 프로세스(300B)에서, 디코더는 프로세스(300B)의 다음 반복에서 예측 동작을 수행하기 위해, 예측된 참조(224)를 공간 예측 스테이지(2042) 또는 시간 예측 스테이지(2044)에 공급할 수 있다. 예컨대, 현재 BPU가 공간 예측 스테이지(2042)에서 인트라 예측을 사용하여 디코딩되는 경우, 예측 참조(224)(예컨대, 디코딩된 현재 BPU)를 생성한 이후에, 디코더는 차후 사용을 위해(예컨대, 현재 픽처의 다음 BPU의 외삽을 위해) 예측 참조(224)를 공간 예측 스테이지(2042)에 직접적으로 공급할 수 있다. 현재 BPU가 시간 예측 스테이지(2044)에서 인터 예측을 사용하여 디코딩되는 경우, 예측 참조(224)(예컨대, 모든 BPU가 디코딩된 참조 픽처)를 생성한 이후에, 인코더는 왜곡(예컨대, 블로킹 아티팩트)을 감소시키거나 제거하기 위해, 예측 참조(224)를 루프 필터 스테이지(232)에 공급할 수 있다. 도 2b에 설명된 방식으로, 디코더는 루프 필터를 예측 참조(224)에 적용할 수 있다. 루프-필터링된 참조 픽처는 차후 사용을 위해(예컨대, 비디오 비트스트림(228)의 향후 인코딩된 픽처에 대한 인터-예측 참조 픽처로서 사용되도록) 버퍼(234)(예컨대, 컴퓨터 메모리의 디코딩된 픽처 버퍼)에 저장될 수 있다. 디코더는 시간 예측 스테이지(2044)에서 사용되도록 버퍼(234)에 하나 이상의 참조 픽처를 저장할 수 있다. 일부 실시예에서, 예측 데이터(206)의 예측 모드 표시자가 인터 예측이 현재 BPU를 인코딩하는데 사용된 것을 나타낼 때, 예측 데이터는 루프 필터의 파라미터(예컨대, 루프 필터 강도)를 더 포함할 수 있다.

[0059] 도 4는 본 개시의 일부 실시예와 일치하는, 비디오를 인코딩하거나 또는 디코딩하기 위한 예시적인 장치(400)의 블록도이다. 도 4에 도시된 바와 같이, 장치(400)는 프로세서(402)를 포함할 수 있다. 프로세서(402)가 본원에서 설명된 명령어를 실행할 때, 장치(400)는 비디오 인코딩 또는 디코딩을 위한 특별화된 기계가 될 수 있다. 프로세서(402)는 정보를 조작하거나 또는 처리할 수 있는 임의의 타입의 회로일 수 있다. 예컨대, 프로세서(402)는 중앙 처리 유닛(또는 "CPU"), 그래픽 처리 유닛(또는 "GPU"), 신경 처리 유닛(neural processing unit, "NPU"), 마이크로제어기 유닛(microcontroller unit, "MCU"), 광학 프로세서, 프로그래머블 로직 제어기, 마이크로제어기, 마이크로프로세서, 디지털 신호 프로세서, 지적 재산권(intellectual property, IP)

코어, 프로그래머블 로직 어레이(Programmable Logic Array, PLA), 프로그래머블 어레이 로직(Programmable Array Logic, PAL), 일반 어레이 로직(Generic Array Logic, GAL), 복합 프로그래머블 논리 디바이스(Complex Programmable Logic Device, CPLD), 현장 프로그래머블 게이트 어레이(Field-Programmable Gate Array, FPGA), 시스템 온 칩(System On Chip, SoC), 애플리케이션-특정 집적 회로(Application-Specific Integrated Circuit, ASIC) 등의 임의의 개수의 임의의 조합을 포함할 수 있다. 일부 실시예에서, 프로세서(402)는 또한, 단일 로직 구성요소로서 그룹화되는 프로세서의 세트일 수 있다. 예컨대, 도 4에 도시된 바와 같이, 프로세서(402)는 프로세서(402a), 프로세서(402b) 및 프로세서(402n)를 포함하는 다수의 프로세서를 포함할 수 있다.

[0060] 장치(400)는 또한, 데이터(예컨대, 명령어의 세트, 컴퓨터 코드, 중간 데이터 등)를 저장하도록 구성되는 메모리(404)를 포함할 수 있다. 예컨대, 도 4에 도시된 바와 같이, 저장된 데이터는 프로그램 명령어(예컨대, 프로세스(200A, 200B, 300A 또는 300B)에서의 스테이지를 구현하기 위한 프로그램 명령어) 및 처리를 위한 데이터(예컨대, 비디오 시퀀스(202), 비디오 비트스트림(228) 또는 비디오 스트림(304))을 포함할 수 있다. 프로세서(402)는 프로그램 명령어 및 처리를 위한 데이터에(예컨대, 버스(410)를 통해) 액세스할 수 있고, 처리를 위한 데이터에 대해 동작 또는 조작을 수행하기 위해 프로그램 명령어를 실행할 수 있다. 메모리(404)는 고속 랜덤 액세스 저장 디바이스 또는 비-휘발성 저장 디바이스를 포함할 수 있다. 일부 실시예에서, 메모리(404)는 랜덤-액세스 메모리(random-access memory, RAM), 읽기-전용 메모리(ROM), 광학 디스크, 자기 디스크, 하드 드라이브, 솔리드-스테이트 드라이브(solid-state drive), 플래시 드라이브, 보안 디지털(SD) 카드, 메모리 스틱, 콤팩트 플래시(compact flash, CF) 카드 등의 임의의 개수의 임의의 조합을 포함할 수 있다. 메모리(404)는 또한, 단일 로직 구성요소로서 그룹화되는 메모리의 그룹(도 4에 미도시됨)일 수 있다.

[0061] 버스(410)는 내부 버스(예컨대, CPU-메모리 버스), 외부 버스(예컨대, 통합 직렬 버스 포트(universal serial bus port)), 주변 구성요소 상호연결 고속 포트(peripheral component interconnect express port)) 등과 같은, 장치(400) 내의 구성요소 사이에서 데이터를 전송하는 통신 디바이스일 수 있다.

[0062] 모호성을 야기하지 않으면서 설명의 용이함을 위해, 프로세서(402) 및 다른 데이터 처리 회로는 본 개시에서, 집합적으로 "데이터 처리 회로"로 지칭된다. 데이터 처리 회로는 전체적으로 하드웨어로 구현되거나, 또는 소프트웨어, 하드웨어 또는 펌웨어의 조합으로 구현될 수 있다. 덧붙여, 데이터 처리 회로는 단일 독립 모듈이거나, 또는 장치(400)의 임의의 다른 구성요소로 전체적으로 또는 부분적으로 결합될 수 있다.

[0063] 장치(400)는 네트워크(예컨대, 인터넷, 인트라넷, 근거리 통신망, 모바일 통신 네트워크 등)와의 유선 통신 또는 무선 통신을 제공하기 위해 네트워크 인터페이스(406)를 더 포함할 수 있다. 일부 실시예에서, 네트워크 인터페이스(406)는 네트워크 인터페이스 제어기(network interface controller, NIC), 무선 주파수(radio frequency, RF) 모듈, 트랜스폰더(transponder), 트랜시버, 모뎀, 라우터, 게이트웨이, 유선 네트워크 어댑터, 무선 네트워크 어댑터, 블루투스 어댑터, 적외선 어댑터, 근거리 통신("NFC") 어댑터, 셀룰러 네트워크 칩 등의 임의의 개수의 임의의 조합을 포함할 수 있다.

[0064] 일부 실시예에서, 선택적으로, 장치(400)는 하나 이상의 주변 디바이스에 대한 연결을 제공하기 위한 주변 인터페이스(408)를 더 포함할 수 있다. 도 4에 도시된 바와 같이, 주변 디바이스는 커서 제어 디바이스(예컨대, 마우스, 터치패드 또는 터치스크린), 키보드, 디스플레이(예컨대, 음극선관 디스플레이, 액정 디스플레이 또는 발광 다이오드 디스플레이), 비디오 입력 디바이스(예컨대, 비디오 아카이브(video archive)에 결합된 입력 인터페이스 또는 카메라) 등을 포함할 수 있지만, 이에 제한되지 않는다.

[0065] 비디오 코덱(예컨대, 프로세스(200A, 200B, 300A 또는 300B)를 수행하는 코덱)이 장치(400)에서 임의의 소프트웨어 또는 하드웨어 모듈의 임의의 조합으로 구현될 수 있다는 것이 유의되어야 한다. 예컨대, 프로세스(200A, 200B, 300A 또는 300B)의 일부 또는 모든 스테이지는 메모리(404)에 로딩될 수 있는 프로그램 명령어와 같은 장치(400)의 하나 이상의 소프트웨어 모듈로서 구현될 수 있다. 다른 예에 대해, 프로세스(200A, 200B, 300A 또는 300B)의 일부 또는 모든 스테이지는 특수화된 데이터 처리 회로(예컨대, FPGA, ASIC, NPU 등)와 같은 장치(400)의 하나 이상의 하드웨어 모듈로서 구현될 수 있다.

[0066] 양자화 및 역양자화 기능 블록(예컨대, 도 2a 또는 도 2b의 양자화 스테이지(214) 및 역양자화 스테이지(218), 도 3a 또는 도 3b의 역양자화 스테이지(218))에서, 양자화 파라미터(QP)는 예측 잔차에 적용되는 양자화(및 역양자화)의 양을 결정하기 위해 사용된다. 픽처 또는 슬라이스의 코딩을 위해 사용되는 초기 QP 값은, 예컨대, 픽처 파라미터 세트(Picture Parameter Set; PPS)에서의 init_qp_minus26 신택스 요소를 사용하고 슬라이스 헤더에서의 slice_qp_delta 신택스 요소를 사용하여 고레벨(high level)에서 시그널링될 수 있다. 또한, QP 값은 양자화 그룹의 입도(granularity)에서 전송된 델타 QP 값을 사용하여 각각의 CU에 대한 로컬 레벨에서 적응될

수 있다.

- [0067] VVC에서, 팔레트 모드는 4:4:4 색상 포맷으로 사용될 수 있다. 팔레트 모드가 활성화될 때, CU 크기가 64×64 이하인 경우, CU 레벨에서 팔레트 모드가 사용되는지를 나타내는 플래그가 송신된다.
- [0068] 도 5는 본 개시의 일부 실시예에 따른, 팔레트 모드에서 코딩된 예시적인 블록(500)의 개략도를 도시한다. 도 5에 도시된 바와 같이, 팔레트 모드가 현재 CU(예컨대, 블록(500))를 코딩하는 데 활용되는 경우, CU의 각각의 위치(예컨대, 위치(501), 위치(502), 위치(503) 또는 위치(504))에서의 샘플 값은 작은 세트의 대표 색상 값으로 표현된다. 세트는 "팔레트" 또는 "팔레트-표"(예컨대, 팔레트(510))로 지칭된다. 팔레트 색상에 가까운 값을 갖는 샘플 위치에 대해, 대응하는 팔레트 인덱스(예컨대, 인덱스 0, 인덱스 1, 인덱스 2 또는 인덱스 3)가 시그널링된다. 일부 개시된 실시예에 따라, 팔레트-표 외부의 색상 값은 이스케이프 인덱스(escape index)(예컨대, 인덱스 4)를 시그널링함으로써 지정될 수 있다. 그런 다음, 이스케이프 색상 인덱스를 사용하는 CU에서의 모든 위치에 대해, (양자화된) 색상 성분 값이 이 위치의 각각에 대해 시그널링된다.
- [0069] 팔레트-표를 코딩하기 위해, 팔레트-예측자가 유지된다. 팔레트-예측자는 비-웨이브프론트(non-wavefront)의 경우에 대해 각각의 슬라이스의 시작에서, 그리고 웨이브프론트의 경우에 대해 각각의 CTU 행의 시작에서 0(예컨대, 비어 있음)으로 초기화된다. 일부 경우에서, 팔레트-예측자는 또한 타일의 시작에서 0으로 초기화될 수 있다. 도 6은 본 개시의 일부 실시예에 따른, 코딩 유닛을 인코딩 및 디코딩한 이후에, 팔레트-예측자를 업데이트하기 위한 예시적인 프로세스의 개략도를 도시한다. 도 6에 도시된 바와 같이, 팔레트-예측자에서의 각각의 엔트리에 대해, 그것이 현재 CU의 현재 팔레트-표에 포함될 것인지를 나타내기 위해 재사용 플래그가 시그널링된다. 재사용 플래그는 0의 실행-길이 코딩을 사용하여 전송되며, 그런 다음 새로운 팔레트 엔트리의 수 및 새로운 팔레트 엔트리에 대한 성분 값이 시그널링된다. 팔레트 코딩된CU를 인코딩 및/또는 디코딩한 이후에, 팔레트-예측자는 현재 팔레트-표를 사용하여 업데이트되고, 현재 팔레트-표에서 재사용되지 않는 이전 팔레트-예측자로부터의 엔트리는 허용된 최대 크기에 도달할 때까지 새로운 팔레트-예측자의 끝에 추가된다.
- [0070] 일부 실시예에서, 이스케이프 심볼이 현재 CU에 존재하는지를 나타내기 위해, 각각의 CU에 대해 이스케이프 플래그가 시그널링된다. 이스케이프 심볼이 존재하면, (도 5에 도시된 바와 같이) 팔레트-표는 하나씩 증분되고, 마지막 인덱스는 이스케이프 심볼에 할당된다.
- [0071] 도 5를 참조하면, CU 내의 샘플의 팔레트 인덱스는 팔레트 인덱스 맵을 형성한다. 인덱스 맵은 수평 또는 수직 트레이버스 스캔(horizontal or vertical traverse scans)을 사용하여 코딩된다. 스캔 순서는 인덱스 요소 palette_transpose_flag를 사용하여 비트스트림에서 명시적으로 시그널링된다. 팔레트 인덱스 맵은 인덱스-실행 모드(index-run mode) 또는 인덱스-복사 모드(index-copy mode)를 사용하여 코딩된다.
- [0072] 일부 실시예에 따라, I 슬라이스의 트리 구조는 시퀀스 파라미터 세트(SPS) 인덱스에서 인덱스 요소 "qtbtt_dual_tree_intra_flag"를 통해 시그널링된다. 1과 동일한 인덱스 요소 qtbtt_dual_tree_intra_flag는, I 슬라이스의 루마 및 크로마 성분에 대해 2개의 별개의 coding_tree 인덱스 구조가 각각 사용됨을 나타낸다. 0과 동일한 인덱스 요소 qtbtt_dual_tree_intra_flag는, I 슬라이스의 루마 및 크로마 성분에 대해 별개의 coding_tree 인덱스 구조가 사용되지 않음을 나타낸다. 또한, P 및 B 슬라이스는 항상 단일-트리 슬라이스로 코딩된다. 개시된 실시예와 일치하여, I 픽처는, 인코딩/디코딩 프로세스에서 다른 픽처를 참조하지 않고, 인트라 코딩된 픽처이다. P 픽처 및 B 픽처 둘 모두는 인터 코딩된 픽처이며, 이는 다른 픽처를 참조하여 디코딩된다. P 픽처와 B 픽처 사이의 차이는, P 픽처의 각각의 블록은 각각의 참조 픽처의 최대 하나의 블록만을 참조할 수 있는 반면, B 픽처의 각각의 블록은 각각의 참조 픽처의 최대 2개의 블록을 참조할 수 있다는 것이다.
- [0073] 일부 실시예에 따라, 이중 루마/크로마 트리를 갖는 슬라이스의 경우, 상이한 팔레트(예컨대, 상이한 팔레트-표)는 루마(Y 성분) 및 크로마(Cb 및 Cr 성분)에 별개로 적용된다. 이중 트리 슬라이스(즉, 이중 루마/크로마 트리)의 경우에, 루마 팔레트-표의 각각의 엔트리는 Y 값을 포함하고, 크로마 팔레트-표의 각각의 엔트리는 Cb 및 Cr 값 둘 모두를 포함한다. 단일 트리의 슬라이스의 경우, 팔레트는 Y, Cb 및 Cr 성분에 공동으로 적용된다(예컨대, 팔레트-표의 각각의 엔트리는 Y, Cb 및 Cr 값을 포함한다). 또한, 4:2:0 및 4:2:2 색상 포맷과 같은 특정 색상 포맷의 경우, 단일-트리 슬라이스의 코딩 유닛(CU)은 허용 가능한 최소 크로마 코딩 블록 크기에 대한 제한에 기인하여, 별개의 루마 및 크로마 트리를 가질 수 있다. 따라서, 이러한 색상 포맷의 경우, 단일 트리 슬라이스 내의 CU는 로컬 이중-트리 구조(예컨대, 슬라이스 레벨에서는 단일 트리인 반면에 CU 레벨에서는 이중 트리)를 가질 수 있다.
- [0074] 따라서, 비-인트라 최소 크로마 인트라 예측 유닛(smallest chroma intra prediction unit, SCIPU)의 경우 크로

마가 더 분할되도록 허용되지 않지만 루마는 더 분할되도록 허용되기 때문에, 단일-트리 슬라이스의 코딩 유닛은 별개의 루마 및 크로마 트리를 가질 수 있다. 단일-트리 코딩에서, SCIPU는 크로마 블록 크기가 16개의 크로마 샘플 이상이고, 64개의 루마 샘플보다 작은 적어도 하나의 자식 루마 블록을 갖는 코딩 트리 노드로 정의된다. 앞서 설명된 바와 같이, SCIPU와 연관된 별개의 트리는 로컬 이중 트리로 불린다.

[0075] 슬라이스의 트리 유형(예컨대, 단일 트리 또는 이중 트리)에 기초하여, 슬라이스에 대해 2개의 유형의 팔레트-표("공동 팔레트" 및 "별개의 팔레트")가 사용될 수 있다. 단일 트리 슬라이스는 공동 팔레트-표를 사용하여 팔레트 코딩될 수 있다. 공동 팔레트-표의 각각의 엔트리는 Y, Cb 및 Cr 색상 성분을 포함하고, 단일 트리 슬라이스 내의 코딩 유닛(CU)의 모든 색상 성분은 (위에 설명된 로컬 이중 트리를 제외하고) 공동 팔레트-표를 사용하여 공동으로 코딩된다. 대조적으로, 이중 트리 슬라이스는 2개의 별개의 팔레트를 사용하여 팔레트 코딩된다. 이중 트리 슬라이스의 루마 및 크로마 성분은 상이한 팔레트-표를 필요로 하며, 개별적으로 코딩된다. 따라서, 이중 트리 슬라이스에 대해, 2개의 인덱스 맵(하나의 루마 성분을 위한 것이고 다른 하나는 크로마 성분을 위한 것임)은 비트 스트림에서 시그널링된다.

[0076] 도 7은 본 개시의 일부 실시예에 따른, 예시적인 균일한 최대 예측자 크기 및 최대 팔레트 크기를 도시하는 예시적인 표 1을 도시한다. 표 1에 도시된 바와 같이, 공동 팔레트와 별개의 팔레트 둘 모두의 최대 팔레트-예측자 크기는 63으로 균일하게 설정되고, 공동 및 별개의 팔레트-표 둘 모두에 대한 최대 팔레트 크기는 31로 균일하게 설정된다. 그러나, 앞서 설명된 바와 같이, 별개의 루마 및 크로마 트리가 있는 이중 트리 슬라이스/CU에는 2개의 별개의 팔레트-표가 필요한 반면에, 공동 루마-크로마 트리가 있는 단일 트리 슬라이스/CU에는 하나의 공동 팔레트-표만이 필요하다. 따라서 이중 트리 슬라이스/CU에 대한 별개의 팔레트-표를 생성하는 것의 복잡성은 단일 트리 슬라이스/CU에 대한 공동 팔레트-표를 생성하는 것의 복잡성의 대략 2배이다.

[0077] 일부 개시된 실시예와 일치하여, 팔레트 코딩 이중 트리 슬라이스/CU 대 단일 트리 슬라이스/CU에 대한 컴퓨팅 복잡성 및 시간의 불균형을 해결하기 위해, 별개의 루마 및 크로마 트리에 대한 최대 예측자 크기는 단일(예컨대, 공동) 루마-크로마 트리에 대한 것보다 낮게 설정될 수 있다. 대안적으로 또는 추가적으로, 별개의 루마 및 크로마 트리에 대한 최대 팔레트 크기(예컨대, 최대 팔레트-표 크기)는 단일(예컨대, 공동) 루마-크로마 트리에 대한 것보다 낮게 설정될 수 있다.

[0078] 일부 개시된 실시예에서, 다음 6개의 변수는 최대 예측자 크기 및 최대 팔레트 크기를 나타내도록 정의된다. 특히, 변수 "max_plt_predictor_size_joint"는 공동 팔레트의 최대 예측 크기를 나타낸다. 변수 "max_plt_predictor_size_luma"는, 별개의 팔레트가 사용되는 경우, 루마 성분의 최대 예측자 크기를 나타낸다. 변수 "max_plt_predictor_size_chroma"는, 별개의 팔레트가 사용되는 경우, 크로마 성분의 최대 예측자 크기를 나타낸다. 변수 "max_plt_size_joint"는 공동 팔레트의 최대 팔레트 크기를 나타낸다. 변수 "max_plt_size_luma"는, 별개의 팔레트가 사용되는 경우, 루마 컴포넌트의 최대 팔레트 크기를 나타낸다. 변수 "max_plt_size_chroma"는, 별개의 팔레트가 사용되는 경우, 크로마 성분의 최대 팔레트 크기를 나타낸다.

[0079] 일부 실시예에서, 최대 팔레트-예측자 크기 및 최대 팔레트 크기는 미리 정의된 세트의 고정된 값이고, 비디오 인코더로 시그널링될 필요가 없다. 도 8은 본 개시의 일부 실시예에 따른, 예시적인 최대 예측자 크기 및 최대 팔레트 크기를 도시하는 예시적인 표 2를 도시한다.

[0080] 일부 실시예에서, 이중 트리 슬라이스의 최대 팔레트-예측자 크기 및 최대 팔레트 크기는, 단일 트리 슬라이스의 최대 팔레트-예측자 크기 및 최대 팔레트 크기의 절반으로 설정된다. 표 2에 도시된 바와 같이 공동 팔레트(즉, 단일 트리 슬라이스의 경우)의 최대 팔레트-예측자 크기 및 최대 팔레트 크기는 각각 63 및 31로 정의된다. 루마 및 크로마 성분 둘 모두의 별개의 팔레트(즉, 이중 트리 슬라이스의 경우)의 최대 팔레트-예측자 크기 및 최대 팔레트 크기는 각각 31 및 15로 정의된다.

[0081] 도 9는 본 개시의 일부 실시예에 따른, 미리 정의된 최대 팔레트-예측자 크기 및 최대 팔레트 크기를 사용하기 위한 예시적인 디코딩 프로세스를 도시하는 예시적인 표 3을 도시한다. 표 3에 도시된 바와 같이 VVC 드래프트 7에서 현재 제안된 팔레트 모드 디코딩 프로세스에 대한 변경 사항은 상자(901-906)에 강조되고 이탤릭체로 강조 표시되고, VVC 드래프트 7에서 현재 제안된 팔레트 모드 디코딩 프로세스로부터 삭제될 내용은 상자(905-906)에 도시되고, 위에 줄을 그어 지워지고, 이탤릭체로 강조된다. 본 실시예에서, CU가 로컬 이중 트리(예컨대, 단일 트리 슬라이스의 별개의 루마/크로마 로컬 트리)로서 코딩되는 경우, 로컬 이중 트리를 코딩하기 위한 최대 예측자 크기는 공동 팔레트의 최대 예측자 크기로 설정된다.

[0082] 도 10은 본 개시의 일부 실시예에 따른, 미리 정의된 최대 팔레트-예측자 크기 및 최대 팔레트 크기를 사용하기

위한 예시적인 팔레트 코딩 선택스 표를 도시하는 예시적인 표 4를 도시한다. 표 1에 도시된 균일한 최대 예측자 크기 및 최대 팔레트 크기를 구현하는 데 사용된 선택스와 비교하여, 선택스에 대한 변경 사항은 표 4에서 상자(1001-1003)에 강조되고 이탤릭체로 강조 표시되며, 선택스로부터 삭제될 선택스 요소는 표 4에서 상자(1002-1003)에 도시되고 위에 줄을 그어 지워지고, 이탤릭체로 강조된다.

[0083] 일부 실시예에서, 공동 팔레트의 최대 팔레트 크기, 및 공동 팔레트의 최대 팔레트 크기와 공동 팔레트의 최대 예측자 크기 사이의 차이는 SPS 선택스를 통해 디코더에 시그널링된다. 본 실시예와 일치하는 예시적인 시맨틱(semantics)은 다음과 같이 설명된다. 선택스 요소 "sps_max_plt_size_joint_minus1"은 공동 팔레트-표의 최대 허용된 팔레트 크기 -1을 지정한다. 선택스 요소 sps_max_plt_size_joint_minus1의 값은 0에서 63(포함)까지의 범위에 있다. 선택스 요소 sps_max_plt_size_joint_minus1이 없으면, 그의 값은 0인 것으로 추론된다. 또한, 선택스 요소 "sps_delta_max_plt_predictor_size_joint"는 공동 팔레트의 최대 허용된 팔레트-예측자 크기와 최대 허용된 팔레트 크기 사이의 차이를 지정한다. 선택스 요소 sps_delta_max_plt_predictor_size_joint의 값은 0에서 63(포함)까지의 범위에 있다. 선택스 요소 sps_delta_max_plt_predictor_size_joint가 없으면, 그의 값은 0으로 추론된다.

[0084] 별개의 루마/크로마 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기는 시그널링되지 않는다. 대신에, 그들은 공동의 팔레트의 최대 팔레트 크기 및 공동의 팔레트의 최대 팔레트-예측자 크기로부터 도출된다. 도 11은 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 예시적인 도출을 도시하는 예시적인 표 5를 도시한다.

[0085] 표 5에 도시된 예에서, 선택스 요소 max_plt_size_joint는, 별개의 루마/크로마 팔레트가 사용될 때, 루마 및 크로마 성분에 균등하게 분배된다. 본 개시와 일치하여, 공동 팔레트의 최대 팔레트 크기는 또한 루마 및 크로마 성분에 불균일하게 분배될 수 있다. 도 12는 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 다른 예시적인 도출을 도시하는 예시적인 표 6을 도시한다. 표 6은 불균일한 분배의 예를 도시한다.

[0086] 도 13은 본 개시의 일부 실시예에 따른, 예시적인 SPS(sequence parameter set) 선택스 표를 도시하는 예시적인 표 7을 도시한다. 표 1에 도시된 균일한 최대 예측자 크기 및 최대 팔레트 크기를 구현하는 데 사용되는 선택스와 비교하여, 선택스의 변경 사항은 표 7에서 상자(1301)에 강조되고 이탤릭체로 강조된다. 표 7에 도시되지 않았지만, 별개의 루마/크로마 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기는 SPS에서, 공동 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기와 함께 시그널링될 수도 있는 것으로 고려된다.

[0087] 일부 실시예에서, 최대 팔레트 크기 및 최대 팔레트-예측자 크기 관련 선택스는 픽처 헤더(PH)를 통해 전송된다. 본 실시예와 일치하는 예시적인 시맨틱은 다음과 같이 설명된다. 선택스 요소 "pic_max_plt_size_joint_minus1"은 PH와 연관된 슬라이스에 대한 공동 팔레트-표의 최대 허용된 팔레트 크기 -1을 지정한다. 선택스 요소 pic_max_plt_size_joint_minus1의 값은 0에서 63(포함)까지의 범위에 있다. 선택스 요소 "pic_delta_max_plt_predictor_size_joint"는 PH와 연관된 슬라이스에 대한 공동 팔레트의 최대 허용된 팔레트-예측자 크기와 최대 허용된 팔레트 크기 사이의 차이를 지정한다. 선택스 요소 pic_delta_max_plt_predictor_size_joint의 최대 허용 가능한 값은 63이다. 선택스 요소 pic_delta_max_plt_predictor_size_joint가 존재하지 않을 때, 그 값은 0인 것으로 추론된다.

[0088] 별개의 루마/크로마 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기는 시그널링되지 않는다. 대신에, 그들은 공동의 팔레트의 최대 팔레트 크기 및 공동의 팔레트의 최대 팔레트-예측자 크기로부터 도출된다. 도 14는 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 다른 예시적인 도출을 도시하는 예시적인 표 8을 도시한다.

[0089] 표 8에 도시된 예에서, 선택스 요소 max_plt_size_joint는, 별개의 루마/크로마 팔레트가 사용될 때, 루마 및 크로마 성분에 균등하게 분배된다. 본 개시와 일치하여, 공동 표의 최대 팔레트 크기는 또한 루마 및 크로마 성분에 불균일하게 분배될 수 있다. 도 15는 본 개시의 일부 실시예에 따른, 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 다른 예시적인 도출을 도시하는 예시적인 표 9를 도시한다. 표 9는 불균일한 분배의 예를 도시한다.

[0090] 도 16은 본 개시의 일부 실시예에 따른, 예시적인 PH 선택스를 도시하는 예시적인 표 10을 도시한다. 표 1에 도시된 균일한 최대 예측자 크기 및 최대 팔레트 크기를 구현하는 데 사용되는 선택스와 비교하여, 선택스에 대한 변경 사항은 표 10에서 상자(1601)에 강조되고 이탤릭체로 강조 표시된다. 표 10에 도시되지 않았지만, 별

개의 루마/크로마 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기는 픽처 헤더에서, 공동 팔레트의 최대 팔레트 크기 및 최대 팔레트-예측자 크기와 함께 시그널링될 수도 있는 것으로 고려된다.

- [0091] 일부 실시예에서, 최대 팔레트 크기 및 최대 팔레트-예측자 크기와 관련된 선택스는 슬라이스 헤더를 통해 각각의 슬라이스에서 시그널링된다. 본 실시예와 일치하는 예시적인 시맨틱은 다음과 같이 설명된다.
- [0092] 구체적으로, 선택스 요소 "slice_max_plt_size_joint_minus1" 및 선택스 요소 "slice_delta_max_plt_predictor_size_joint"는, 슬라이스가 단일 트리 슬라이스로서 코딩된 경우, 조건부로 시그널링된다. 선택스 요소 slice_max_plt_size_joint_minus1은 단일 트리 슬라이스에 대한 공동 팔레트-표의 최대 허용된 팔레트 크기 -1을 지정한다. 선택스 요소 slice_max_plt_size_joint의 최대값이 63인 것은 비트스트림 적합성(conformance)의 요건이다. 선택스 요소 slice_delta_max_plt_predictor_size_joint는 단일 트리 슬라이스에 대한 공동 팔레트의 최대 허용된 팔레트-예측자 크기와 최대 허용된 팔레트 크기 사이의 차이를 지정한다. 선택스 요소 slice_delta_max_plt_predictor_size_joint의 최대 허용 가능한 값은 63이다. 선택스 요소 slice_delta_max_plt_predictor_size_joint가 존재하지 않는 경우, 그 값은 0인 것으로 추론된다.
- [0093] 슬라이스가 이중 트리 슬라이스로서 코딩된 경우, 선택스 요소 "slice_max_plt_size_luma_minus1" 및 선택스 요소 "slice_delta_max_plt_predictor_size_luma"는 조건부로 시그널링된다. 선택스 요소 slice_max_plt_size_luma_minus1은 이중 트리 슬라이스에 대한 루마 팔레트-표의 최대 허용된 팔레트 크기 -1을 지정한다. 선택스 요소 slice_max_plt_size_luma가 존재하지 않는 경우, 그 값은 0인 것으로 추론된다. 선택스 요소 slice_max_plt_size_luma_minus1의 최대값이 63인 것은 비트스트림 적합성의 요건이다. 선택스 요소 slice_delta_max_plt_predictor_size_luma는 이중 트리 슬라이스에 대한 루마 팔레트의 최대 허용된 팔레트-예측자 크기와 최대 허용된 팔레트 크기 사이의 차이를 지정한다. 선택스 요소 slice_delta_max_plt_predictor_size_luma의 최대 허용 가능한 값은 63이다. 선택스 요소 slice_delta_max_plt_predictor_size_luma가 존재하지 않을 때, 그 값은 0인 것으로 추론된다.
- [0094] 도 17은 본 개시의 일부 실시예에 따른, I, P 및 B 슬라이스에 대한 최대 팔레트 크기 및 최대 팔레트-예측자 크기의 예시적인 도출을 도시하는 예시적인 표 11을 도시한다.
- [0095] 도 18은 본 개시의 일부 실시예에 따른, 예시적인 SH 선택스를 도시하는 예시적인 표 12를 도시한다. 표 1에 도시된 균일한 최대 예측자 크기 및 최대 팔레트 크기를 구현하는 데 사용되는 선택스와 비교하여, 선택스에 대한 변경 사항은 표 12에서 상자(1801)에 강조되고 이탤릭체로 강조 표시된다. 본 실시예와 일치하는 예측 업데이트 절차는 표 3에 도시된 것과 동일하고, 본 실시예와 일치하는 팔레트 코딩 선택스는 표 4에 도시된 것과 동일하다.
- [0096] 도 19는 본 개시의 일부 실시예에 따른 예시적인 팔레트 코딩 방법(1900)의 흐름도를 도시한다. 방법(1900)은 인코더(예컨대, 도 2a의 프로세스(200A) 또는 도 2b의 프로세스(200B)), 디코더(예컨대, 도 3a의 프로세스(300A) 또는 도 3b의 프로세스(300B))에 의해 수행되거나 또는 장치(예컨대, 도 4의 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소에 의해 수행될 수 있다. 예컨대, 프로세서(예컨대, 도 4의 프로세서(402))는 방법(1900)을 수행할 수 있다. 일부 실시예에서, 방법(1900)은, 컴퓨터(예컨대, 도 4의 장치(400))에 의해 실행되는 프로그램 코드와 같은 컴퓨터 실행 가능 명령어를 포함하는, 컴퓨터 판독가능 매체에 구현된 컴퓨터 프로그램 제품에 의해 구현될 수 있다.
- [0097] 단계(1901)에서, CU의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지에 대한 결정이 이루어질 수 있다. 예컨대, 변수 트리 유형은, (예컨대, 도 9의 표 3 또는 도 10의 표 4에 도시된 바와 같이) CU의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 나타내는 데 활용될 수 있다.
- [0098] 단계(1903)에서, 팔레트 모드에서 루마 성분 및 크로마 성분이 공동으로 코딩되는 것에 응답하여, CU에 대한 제1 최대 팔레트-표 크기가 결정될 수 있다. 일부 실시예에서, CU에 대한 제1 최대 팔레트-표 크기는 비디오 비트스트림에서 시그널링된 제1 선택스 요소(예컨대, 도 13의 표 7에 도시된 선택스 요소 sps_max_plt_size_joint_minus1 또는 도 16의 표 10에 도시된 선택스 요소 pic_max_plt_size_joint_minus1)의 값에 기초하여 결정될 수 있다.
- [0099] 단계(1905)에서, 팔레트 모드에서 루마 성분 및 크로마 성분이 공동으로 코딩되는 것에 응답하여, CU에 대한 제1 최대 팔레트-예측자 크기가 결정될 수 있다. 일부 실시예에서, CU에 대한 제1 최대 팔레트-예측자 크기는 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값 및 제2 선택스 요소의 값(예컨대, 도 13의 표 7에 도시

된 선택스 요소 `sps_delta_max_plt_predictor_size_joint` 또는 도 16의 표 10에 도시된 선택스 요소 `pic_delta_max_plt_predictor_size_joint`)에 기초하여 결정될 수 있다. 예컨대, CU에 대한 제1 최대 팔레트-예측자 크기는 (예컨대, 도 11의 표 5, 도 12의 표 6, 도 14의 표 8, 또는 도 15의 표 9에 도시된 바와 같이) 제1 선택스 요소의 값과 제2 선택스 요소의 값의 합인 것으로 결정될 수 있다. 일부 실시예에서, 제1 선택스 요소 및 제2 선택스 요소는 (예컨대, 도 13의 표 7에 도시된 바와 같이) CU와 연관된 SPS 또는 (예컨대, 도 16의 표 10에 도시된 바와 같이) CU와 관련된 PH에서 시그널링된다.

[0100] 단계(1907)에서, 팔레트 모드에서 루마 성분 및 크로마 성분이 공동으로 코딩되는 것에 응답하여, CU는 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 예측될 수 있다. 예컨대, CU는 도 9의 표 3에 도시된 바와 같이 예측될 수 있다.

[0101] 일부 실시예에서, 방법(1900)은: 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여, 제1 최대 팔레트-표 크기에 기초하여 CU에 대한 제2 최대 팔레트-표 크기를 결정하는 단계, 제1 최대 팔레트-예측자 크기에 기초하여 CU에 대한 제2 최대 팔레트-예측자 크기를 결정하는 단계, 및 제2 최대 팔레트-표 크기 및 제2 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 포함할 수 있다. 제2 최대 팔레트-표 크기 또는 제2 최대 팔레트-예측자 크기는 루마 성분 또는 크로마 성분에 대한 것이다. 예컨대, 루마 성분 또는 크로마 성분에 대한 최대 팔레트-표 크기 또는 최대 팔레트-예측자 크기는 도 11의 표 5, 도 12의 표 6, 도 14의 표 8, 또는 도 15의 표 9에 기초하여 결정될 수 있다.

[0102] 일부 실시예에서, 방법(1900)은, CU에 대한 제1 최대 팔레트-표 크기를 제1 미리 결정된 값인 것으로 결정하는 단계를 포함할 수 있다. 방법(1900)은 또한 CU에 대한 제1 최대 팔레트-예측자 크기를 제2 미리 결정된 값인 것으로 결정하는 단계를 포함할 수 있다. 예컨대, 도 8의 표 2에 도시된 바와 같이, 공동 팔레트에 대한 최대 팔레트-표 크기는 31일 수 있고, 공동 팔레트에 대한 최대 팔레트-예측자 크기는 63일 수 있다. 일부 실시예에서, 방법(1900)은: 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여, CU에 대한 제3 최대 팔레트-표 크기를 제3 미리 결정된 값인 것으로 결정하는 단계, 및 제3 최대 팔레트-표 크기에 기초하여, CU를 예측하는 단계를 포함할 수 있다. 제3 미리 결정된 값은 제1 미리 결정된 값보다 작다. 예컨대, 도 8의 표 2에 도시된 바와 같이, 루마 또는 크로마 팔레트에 대한 최대 팔레트-표 크기는 15일 수 있다.

[0103] 일부 실시예에서, 방법(1900)은, 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부인 것에 응답하여, (예컨대, 도 9의 표 3에 도시된 바와 같이) 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 포함할 수 있다.

[0104] 일부 실시예에서, 방법(1900)은, 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부가 아닌 것에 응답하여, CU에 대한 제3 최대 팔레트-예측자 크기를 제4 미리 결정된 값인 것으로 결정하는 단계, 및 제3 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 포함할 수 있다. 제4 미리 결정된 값은 제2 미리 결정된 값보다 작다. 예컨대, 도 8의 표 2에 도시된 바와 같이, 별개의 팔레트에 대한 최대 팔레트-예측자 크기는 31일 수 있다. CU는 도 9의 표 3에 도시된 바와 같이 예측될 수 있다.

[0105] 일부 실시예에서, 방법(1900)은: CU를 포함하는 픽처 슬라이스가 단일 트리 슬라이스 또는 이중 트리 슬라이스 인지를 결정하는 단계, 픽처 슬라이스가 단일 트리 슬라이스인 것에 응답하여, 픽처 슬라이스의 슬라이스 헤더에서 시그널링된 제3 선택스 요소의 값에 기초하여, 픽처 슬라이스 내의 CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계, 및 슬라이스 헤더에서 시그널링된 제3 선택스 요소의 값 및 제4 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계를 포함할 수 있다. CU에 대한 제1 최대 팔레트-예측자 크기는 제3 선택스 요소의 값과 제4 선택스 요소의 값의 합인 것으로 결정될 수 있다. 예컨대, 도 17의 표 11에 도시된 바와 같이, 픽처 슬라이스가 단일 트리 슬라이스인 것에 응답하여(예컨대, `slice_type != I || qtbt_dual_tree_intra_flag == 0`), 공동 팔레트에 대한 최대 팔레트-표 크기는 슬라이스 헤더(예컨대, 도 18의 표 12에 도시된 바와 같은 SH)에서 시그널링된 선택스 요소 `slice_max_plt_size_joint_minus1`의 값에 기초하여 결정될 수 있고, 공동 팔레트에 대한 최대 팔레트-예측자 크기는 슬라이스 헤더(예컨대, 도 18의 표 12에 도시된 바와 같은 SH)에서 시그널링된 선택스 요소 `slice_max_plt_size_joint_minus1`의 값 및 선택스 요소 `slice_delta_max_plt_predictor_size_joint`의 값의 합인 것으로 결정될 수 있다.

[0106] 일부 실시예에서, 방법(1900)은: 픽처 슬라이스가 이중 트리 슬라이스인 것에 응답하여, 슬라이스 헤더에서 시그널링된 제5 선택스 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-표 크기를 결정하는 단계, 슬라이스 헤더에서 시그널링된 제5 선택스 요소의 값 및 제6 선택스 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-예측자 크기를 결정하는 단계, 및 제4 최대 팔레트-표 크기 및 제4 최대 팔레트-예측자 크기에 기초하여 CU를 예

측하는 단계를 포함할 수 있다. CU에 대한 제4 최대 팔레트-예측자 크기는 제5 선택스 요소의 값과 제6 선택스 요소의 값의 합인 것으로 결정될 수 있다. 예컨대, 도 17의 표 11에 도시된 바와 같이, 픽처 슬라이스가 이중 트리 슬라이스인 것에 응답하여, 루마 또는 크로마 팔레트에 대한 최대 팔레트-표 크기는 슬라이스 헤더(예컨대, 도 18의 표 12에 도시된 바와 같은 SH)에서 시그널링되는 선택스 요소 slice_max_plt_size_luma_minus1의 값에 기초하여 결정될 수 있고, 루마 또는 크로마 팔레트에 대한 최대 팔레트-예측자 크기는 슬라이스 헤더(예컨대, 도 18의 표 12에 도시된 바와 같은 SH)에서 시그널링되는 선택스 요소 slice_delta_max_plt_predictor_size_luma의 값과 선택스 요소 slice_max_plt_size_luma_minus1의 값의 합인 것으로 결정될 수 있다.

- [0107] 실시예는 다음의 조항을 사용하여 추가로 설명될 수 있다.
- [0108] 1. 팔레트 코딩 방법으로서,
- [0109] 코딩 유닛(CU)의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 단계; 및
- [0110] 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여,
- [0111] CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계;
- [0112] CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계; 및
- [0113] 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 포함한다.
- [0114] 2. 제1 조항의 방법에 있어서, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계는:
- [0115] 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계를 포함한다.
- [0116] 3. 제2 조항의 방법에 있어서, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계는:
- [0117] 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값 및 제2 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계를 포함한다.
- [0118] 4. 제3 조항의 방법에 있어서, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계는:
- [0119] CU에 대한 제1 최대 팔레트-예측자 크기를 제1 선택스 요소의 값과 제2 선택스 요소의 값의 합인 것으로 결정하는 단계를 포함한다.
- [0120] 5. 제3 조항 및 제4 조항 중 어느 하나의 방법에 있어서, 제1 선택스 요소 및 제2 선택스 요소는 CU와 연관된 시퀀스 파라미터 세트(SPS)에서 시그널링된다.
- [0121] 6. 제3 조항 및 제4 조항 중 어느 하나의 방법에 있어서, 제1 선택스 요소 및 제2 선택스 요소는 CU와 연관된 픽처 헤더(PH)에서 시그널링된다.
- [0122] 7. 제1 조항 내지 제6 조항 중 어느 하나의 방법에 있어서,
- [0123] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
- [0124] 제1 최대 팔레트-표 크기에 기초하여 CU에 대한 제2 최대 팔레트-표 크기를 결정하는 단계,
- [0125] 제1 최대 팔레트-예측자 크기에 기초하여 CU에 대한 제2 최대 팔레트-예측자 크기를 결정하는 단계, 및
- [0126] 제2 최대 팔레트-표 크기 및 제2 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 더 포함한다.
- [0127] 8. 제7 조항의 방법에 있어서, 제2 최대 팔레트-표 크기 또는 제2 최대 팔레트-예측자 크기는 루마 성분 또는 크로마 성분에 대한 것이다.
- [0128] 9. 제1 조항의 방법에 있어서, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계는:
- [0129] CU에 대한 제1 최대 팔레트-표 크기를 제1 미리 결정된 값인 것으로 결정하는 단계를 포함한다.
- [0130] 10. 제1 조항 및 제9 조항 중 어느 하나의 방법에 있어서, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계는:

- [0131] CU에 대한 제1 최대 팔레트-예측자 크기를 제2 미리 결정된 값인 것으로 결정하는 단계를 포함한다.
- [0132] 11. 제9 조항 및 제10 조항 중 어느 하나의 방법에 있어서,
- [0133] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
- [0134] CU에 대한 제3 최대 팔레트-표 크기를 제3 미리 결정된 값인 것으로 결정하는 단계; 및
- [0135] 제3 최대 팔레트-표 크기에 기초하여, CU를 예측하는 단계를 더 포함하고,
- [0136] 제3 미리 결정된 값은 제1 미리 결정된 값보다 작다.
- [0137] 12. 제11 조항의 방법에 있어서,
- [0138] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부인 것에 응답하여, 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 더 포함한다.
- [0139] 13. 제10 조항 내지 제12 조항 중 어느 하나의 방법에 있어서,
- [0140] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부가 아닌 것에 응답하여,
- [0141] CU에 대한 제3 최대 팔레트-예측자 크기를 제4 미리 결정된 값인 것으로 결정하는 단계; 및
- [0142] 제3 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 더 포함하고,
- [0143] 제4 미리 결정된 값은 제2 미리 결정된 값보다 작다.
- [0144] 14. 제1 조항의 방법에 있어서,
- [0145] CU를 포함하는 픽처 슬라이스가 단일 트리 슬라이스 또는 이중 트리 슬라이스인지를 결정하는 단계; 및
- [0146] 픽처 슬라이스가 단일 트리 슬라이스인 것에 응답하여,
- [0147] 픽처 슬라이스의 슬라이스 헤더에서 시그널링된 제3 선택 요소의 값에 기초하여, 픽처 슬라이스 내의 CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계; 및
- [0148] 슬라이스 헤더에서 시그널링된 제3 선택 요소의 값 및 제4 선택 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계를 더 포함한다.
- [0149] 15. 제14 조항의 방법에 있어서, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계는:
- [0150] CU에 대한 제1 최대 팔레트-예측자 크기를 제3 선택 요소의 값과 제4 선택 요소의 값의 합인 것으로 결정하는 단계를 포함한다.
- [0151] 16. 제14 조항 및 제15 조항 중 어느 하나의 방법에 있어서,
- [0152] 픽처 슬라이스가 이중 트리 슬라이스인 것에 응답하여,
- [0153] 슬라이스 헤더에서 시그널링된 제5 선택 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-표 크기를 결정하는 단계,
- [0154] 슬라이스 헤더에서 시그널링된 제5 선택 요소의 값 및 제6 선택 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-예측자 크기를 결정하는 단계, 및
- [0155] 제4 최대 팔레트-표 크기 및 제4 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 더 포함한다.
- [0156] 17. 제16 조항의 방법에 있어서, CU에 대한 제4 최대 팔레트-예측자 크기를 결정하는 단계는:
- [0157] CU에 대한 제4 최대 팔레트-예측자 크기를 제5 선택 요소의 값과 제6 선택 요소의 값의 합인 것으로 결정하는 단계를 포함한다.
- [0158] 18. 비디오 처리 장치로서,
- [0159] 명령어를 저장하기 위한 적어도 하나의 메모리; 및
- [0160] 명령어를 실행하도록 구성된 적어도 하나의 프로세서를 포함하고,

- [0161] 명령어는 장치로 하여금:
- [0162] 코딩 유닛(CU)의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 것; 및
- [0163] 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여,
- [0164] CU에 대한 제1 최대 팔레트-표 크기를 결정하는 것;
- [0165] CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 것; 및
- [0166] 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 한다.
- [0167] 19. 제18 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0168] 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0169] 20. 제19 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0170] 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값 및 제2 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0171] 21. 제20 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0172] CU에 대한 제1 최대 팔레트-예측자 크기를 제1 선택스 요소의 값과 제2 선택스 요소의 값의 합인 것으로 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0173] 22. 제20 조항 및 제21 조항 중 어느 하나의 장치에 있어서, 제1 선택스 요소 및 제2 선택스 요소는 CU와 연관된 시퀀스 파라미터 세트(SPS)에서 시그널링된다.
- [0174] 23. 제20 조항 및 제21 조항 중 어느 하나의 장치에 있어서, 제1 선택스 요소 및 제2 선택스 요소는 CU와 연관된 픽처 헤더(PH)에서 시그널링된다.
- [0175] 24. 제18 조항 내지 제23 조항 중 어느 하나의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0176] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
- [0177] 제1 최대 팔레트-표 크기에 기초하여 CU에 대한 제2 최대 팔레트-표 크기를 결정하는 것,
- [0178] 제1 최대 팔레트-예측자 크기에 기초하여 CU에 대한 제2 최대 팔레트-예측자 크기를 결정하는 것, 및
- [0179] 제2 최대 팔레트-표 크기 및 제2 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0180] 25. 제24 조항의 장치에 있어서, 제2 최대 팔레트-표 크기 또는 제2 최대 팔레트-예측자 크기는 루마 성분 또는 크로마 성분에 대한 것이다.
- [0181] 26. 제18 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0182] CU에 대한 제1 최대 팔레트-표 크기를 제1 미리 결정된 값인 것으로 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0183] 27. 제18 조항 및 제26 조항 중 어느 하나의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0184] CU에 대한 제1 최대 팔레트-예측자 크기를 제2 미리 결정된 값인 것으로 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0185] 28. 제26 조항 및 제27 조항 중 어느 하나의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0186] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
- [0187] CU에 대한 제3 최대 팔레트-표 크기를 제3 미리 결정된 값인 것으로 결정하는 것; 및
- [0188] 제3 최대 팔레트-표 크기에 기초하여, CU를 예측하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성되고,
- [0189] 제3 미리 결정된 값은 제1 미리 결정된 값보다 작다.

- [0190] 29. 제28 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0191] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부인 것에 응답하여, 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0192] 30. 제27 조항 내지 제29 조항 중 어느 하나의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0193] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부가 아닌 것에 응답하여,
- [0194] CU에 대한 제3 최대 팔레트-예측자 크기를 제4 미리 결정된 값인 것으로 결정하는 것; 및
- [0195] 제3 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성되고,
- [0196] 제4 미리 결정된 값은 제2 미리 결정된 값보다 작다.
- [0197] 31. 제18 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0198] CU를 포함하는 픽처 슬라이스가 단일 트리 슬라이스 또는 이중 트리 슬라이스인지를 결정하는 것; 및
- [0199] 픽처 슬라이스가 단일 트리 슬라이스인 것에 응답하여,
- [0200] 픽처 슬라이스의 슬라이스 헤더에서 시그널링된 제3 선택스 요소의 값에 기초하여, 픽처 슬라이스 내의 CU에 대한 제1 최대 팔레트-표 크기를 결정하는 것; 및
- [0201] 슬라이스 헤더에서 시그널링된 제3 선택스 요소의 값 및 제4 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0202] 32. 제31 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0203] CU에 대한 제1 최대 팔레트-예측자 크기를 제3 선택스 요소의 값과 제4 선택스 요소의 값의 합인 것으로 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0204] 33. 제31 조항 및 제32 조항 중 어느 하나의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0205] 픽처 슬라이스가 이중 트리 슬라이스인 것에 응답하여,
- [0206] 슬라이스 헤더에서 시그널링된 제5 선택스 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-표 크기를 결정하는 것,
- [0207] 슬라이스 헤더에서 시그널링된 제5 선택스 요소의 값 및 제6 선택스 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-예측자 크기를 결정하는 것, 및
- [0208] 제4 최대 팔레트-표 크기 및 제4 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0209] 34. 제33 조항의 장치에 있어서, 적어도 하나의 프로세서는 장치로 하여금:
- [0210] CU에 대한 제4 최대 팔레트-예측자 크기를 제5 선택스 요소의 값과 제6 선택스 요소의 값의 합인 것으로 결정하는 것을 수행하게 하기 위한 명령어를 실행하도록 구성된다.
- [0211] 35. 명령어의 세트를 저장하는 비일시적 컴퓨터 관독 가능 저장 매체로서, 명령어의 세트는 비디오 처리 장치로 하여금 방법을 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하고, 방법은:
- [0212] 코딩 유닛(CU)의 루마 성분 및 CU의 크로마 성분이 팔레트 모드에서 공동으로 또는 개별적으로 코딩되는지를 결정하는 단계; 및
- [0213] 루마 성분 및 크로마 성분이 팔레트 모드에서 공동으로 코딩되는 것에 응답하여,
- [0214] CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계;
- [0215] CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계; 및

- [0216] 제1 최대 팔레트-표 크기 및 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 포함한다.
- [0217] 36. 제35 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0218] 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0219] 37. 제36 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0220] 비디오 비트스트림에서 시그널링된 제1 선택스 요소의 값 및 제2 선택스 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0221] 38. 제37 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0222] CU에 대한 제1 최대 팔레트-예측자 크기를 제1 선택스 요소의 값과 제2 선택스 요소의 값의 합인 것으로 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0223] 39. 제37 조항 및 제38 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 제1 선택스 요소 및 제2 선택스 요소는 CU와 연관된 시퀀스 파라미터 세트(SPS)에서 시그널링된다.
- [0224] 40. 제37 조항 및 제38 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 제1 선택스 요소 및 제2 선택스 요소는 CU와 연관된 픽처 헤더(PH)에서 시그널링된다.
- [0225] 41. 제35 조항 내지 제40 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0226] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
- [0227] 제1 최대 팔레트-표 크기에 기초하여 CU에 대한 제2 최대 팔레트-표 크기를 결정하는 단계,
- [0228] 제1 최대 팔레트-예측자 크기에 기초하여 CU에 대한 제2 최대 팔레트-예측자 크기를 결정하는 단계, 및
- [0229] 제2 최대 팔레트-표 크기 및 제2 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0230] 42. 제41 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 제2 최대 팔레트-표 크기 또는 제2 최대 팔레트-예측자 크기는 루마 성분 또는 크로마 성분에 대한 것이다.
- [0231] 43. 제35 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0232] CU에 대한 제1 최대 팔레트-표 크기를 제1 미리 결정된 값인 것으로 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0233] 44. 제35 조항 및 제43 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0234] CU에 대한 제1 최대 팔레트-예측자 크기를 제2 미리 결정된 값인 것으로 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0235] 45. 제43 조항 및 제44 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0236] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되는 것에 응답하여,
- [0237] CU에 대한 제3 최대 팔레트-표 크기를 제3 미리 결정된 값인 것으로 결정하는 단계; 및
- [0238] 제3 최대 팔레트-표 크기에 기초하여, CU를 예측하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하고,
- [0239] 제3 미리 결정된 값은 제1 미리 결정된 값보다 작다.
- [0240] 46. 제45 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0241] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부인 것에 응

답하여, 제1 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.

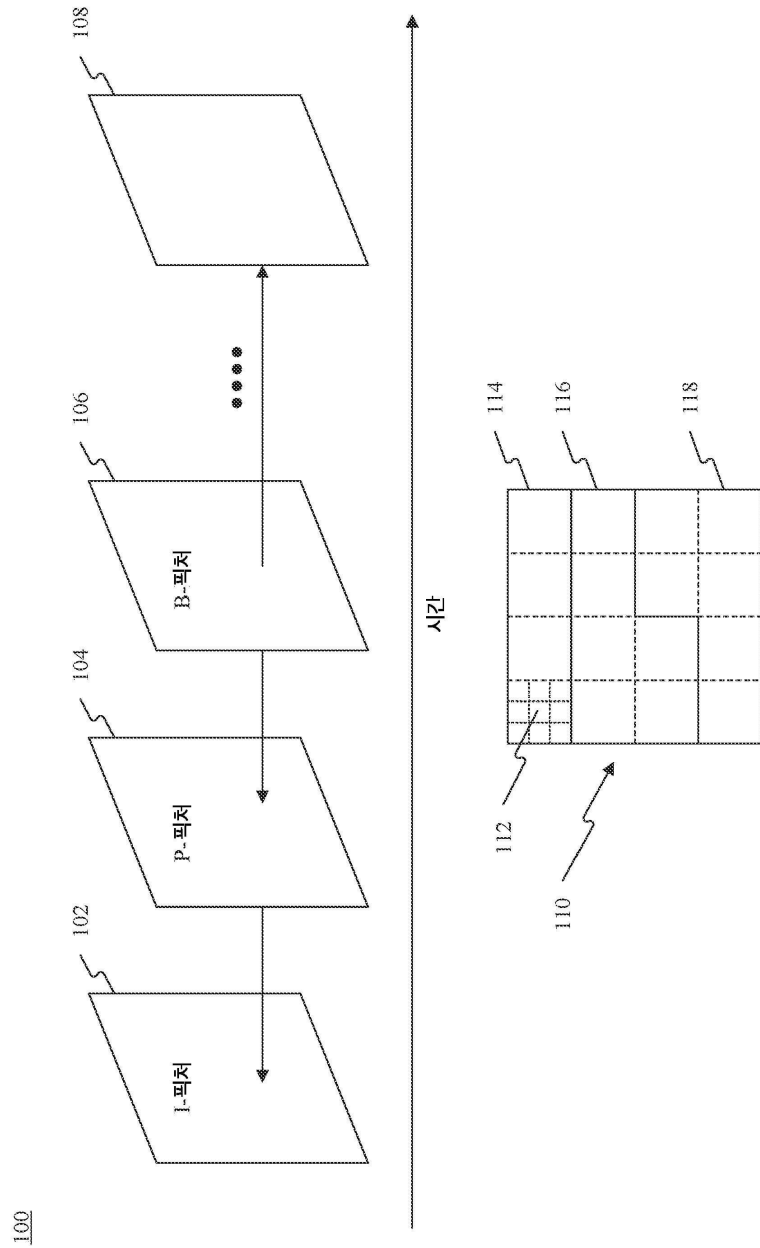
- [0242] 47. 제44 조항 내지 제46 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0243] 루마 성분 및 크로마 성분이 팔레트 모드에서 개별적으로 코딩되고 CU가 단일 트리 슬라이스의 일부가 아닌 것에 응답하여,
- [0244] CU에 대한 제3 최대 팔레트-예측자 크기를 제4 미리 결정된 값인 것으로 결정하는 단계; 및
- [0245] 제3 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하고,
- [0246] 제4 미리 결정된 값은 제2 미리 결정된 값보다 작다.
- [0247] 48. 제35 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0248] CU를 포함하는 픽처 슬라이스가 단일 트리 슬라이스 또는 이중 트리 슬라이스인지를 결정하는 단계; 및
- [0249] 픽처 슬라이스가 단일 트리 슬라이스인 것에 응답하여,
- [0250] 픽처 슬라이스의 슬라이스 헤더에서 시그널링된 제3 선택 요소의 값에 기초하여, 픽처 슬라이스 내의 CU에 대한 제1 최대 팔레트-표 크기를 결정하는 단계; 및
- [0251] 슬라이스 헤더에서 시그널링된 제3 선택 요소의 값 및 제4 선택 요소의 값에 기초하여, CU에 대한 제1 최대 팔레트-예측자 크기를 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0252] 49. 제48 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0253] CU에 대한 제1 최대 팔레트-예측자 크기를 제3 선택 요소의 값과 제4 선택 요소의 값의 합인 것으로 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0254] 50. 제48 조항 및 제49 조항 중 어느 하나의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0255] 픽처 슬라이스가 이중 트리 슬라이스인 것에 응답하여,
- [0256] 슬라이스 헤더에서 시그널링된 제5 선택 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-표 크기를 결정하는 단계,
- [0257] 슬라이스 헤더에서 시그널링된 제5 선택 요소의 값 및 제6 선택 요소의 값에 기초하여, CU에 대한 제4 최대 팔레트-예측자 크기를 결정하는 단계, 및
- [0258] 제4 최대 팔레트-표 크기 및 제4 최대 팔레트-예측자 크기에 기초하여 CU를 예측하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0259] 51. 제50 조항의 비밀시적 컴퓨터 판독 가능 저장 매체에 있어서, 명령어의 세트는 비디오 처리 장치로 하여금:
- [0260] CU에 대한 제4 최대 팔레트-예측자 크기를 제5 선택 요소의 값과 제6 선택 요소의 값의 합인 것으로 결정하는 단계를 수행하게 하도록 하나 이상의 처리 디바이스에 의해 실행 가능하다.
- [0261] 일부 실시예에서, 명령어를 포함하는 비밀시적 컴퓨터 판독 가능 저장 매체가 또한 제공되고, 명령어는 전술한 방법을 수행하게 하도록 디바이스(이를테면, 개시된 인코더 및 디코더)에 의해 실행될 수 있다. 비밀시적 매체의 일반적인 형태는, 예컨대, 플로피 디스크, 플래터블 디스크, 하드 디스크, 솔리드 스테이트 드라이브, 자기 테이프 또는 임의의 다른 자기 데이터 저장 매체, CD-ROM, 임의의 다른 광학 데이터 저장 매체, 홀의 패턴을 갖는 임의의 물리적 매체, RAM, PROM 및 EPROM, FLASH-EPROM 또는 다른 임의의 플래시 메모리, NVRAM, 캐시, 레지스터, 임의의 다른 메모리 칩 또는 카트리지, 및 이들의 네트워크 버전을 포함한다. 디바이스는 하나 이상의 프로세서(CPU), 입력/출력 인터페이스, 네트워크 인터페이스 및/또는 메모리를 포함할 수 있다.
- [0262] 본원에서 "제1" 및 "제2"와 같은 관계 용어는 엔티티 또는 동작을 다른 엔티티 또는 동작과 구분하기 위해서만 사용되며, 이러한 엔티티 또는 동작 간의 임의의 실제 관계 또는 시퀀스를 요구하거나 암시하지 않는다는 것이 유의되어야 한다. 더욱이, 포함하는(comprising), "갖는(having)", "포함하는(containing)" 및 "포함하는

(including)"이라는 단어 및 다른 유사한 형태는 그 의미가 동등하고, 이러한 단어 중 어느 하나를 따르는 항목 또는 항목들이 이러한 항목 또는 항목들의 철저한 열거(exhaustive listing)를 의미하지 않거나 또는 열거된 항목 또는 항목만으로 제한되는 것을 의미하지 않는다는 점에서, 제약을 두지 않는 것으로(open-ended) 의도된다.

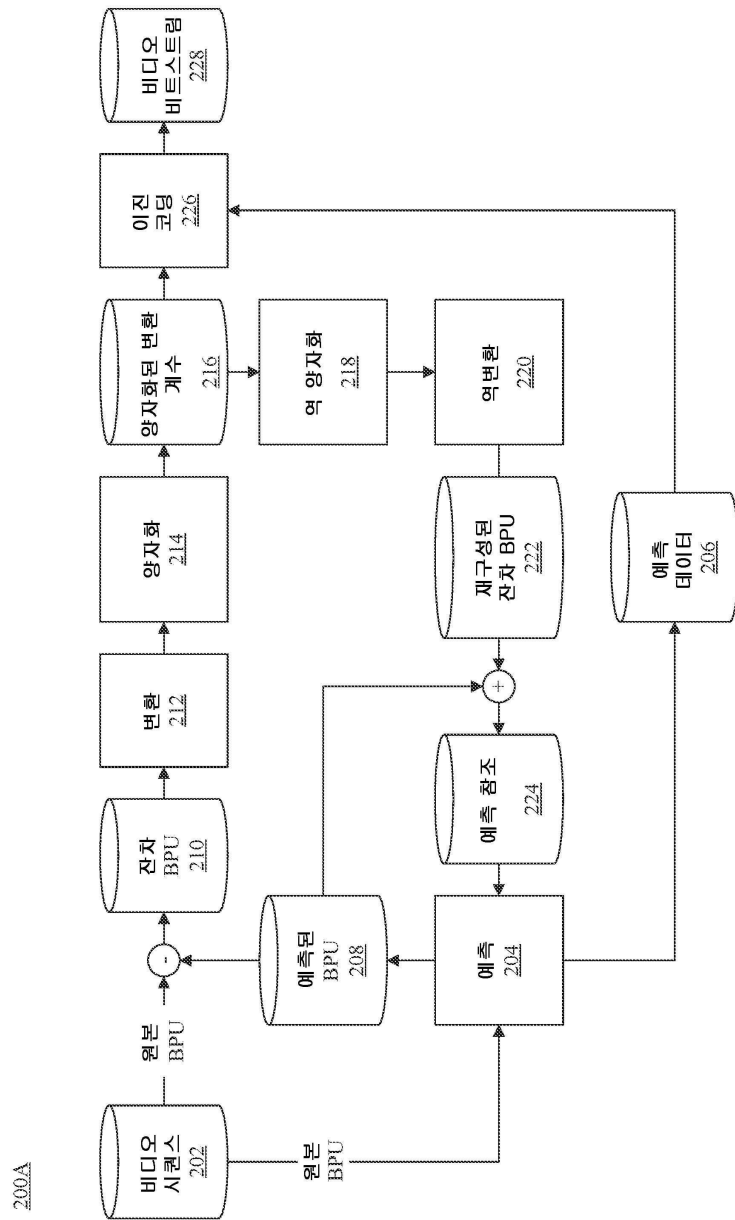
- [0263] 본원에서 사용된 바와 같이, 달리 구체적으로 언급되지 않는 한, "또는"이란 용어는 실행 불가능한 경우를 제외하고 모든 가능한 조합을 포함한다. 예컨대, 데이터베이스가 A 또는 B가 포함될 수 있다고 명시되어 있으면, 달리 구체적으로 명시되지 않거나 실행 불가능한 경우를 제외하고, 데이터베이스는 A 또는 B, 또는 A 및 B를 포함할 수 있다. 제2 예시로서, 데이터베이스가 A, B 또는 C를 포함할 수 있다고 명시되어 있으면, 달리 구체적으로 명시되지 않거나 실행 불가능한 경우를 제외하고, 데이터베이스는 A 또는 B 또는 C, 또는 A 및 B, 또는 A 및 C, 또는 B 및 C, 또는 A 및 B 및 C를 포함할 수 있다.
- [0264] 상술한 실시예는 하드웨어 또는 소프트웨어(프로그램 코드), 또는 하드웨어 및 소프트웨어의 조합에 의해 구현될 수 있다는 것이 인식된다. 소프트웨어로 구현되는 경우, 이는 상술한 컴퓨터 판독 가능 매체에 저장될 수 있다. 소프트웨어는 프로세서에 의해 실행될 때 개시된 방법을 수행할 수 있다. 본 개시에서 설명된 컴퓨팅 유닛 및 다른 기능 유닛은 하드웨어 또는 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합에 의해 구현될 수 있다. 통상의 기술자는 또한 위에서 설명된 모듈/유닛 중 다수가 하나의 모듈/유닛으로 결합될 수 있고, 위에서 설명된 모듈/유닛 각각이 복수의 서브-모듈/서브-유닛으로 더 분할될 수 있음을 이해할 것이다.
- [0265] 전술한 명세서에서, 실시예는 구현마다 다를 수 있는 수많은 특정 세부사항을 참조로 설명되었다. 설명된 실시예의 특정 적응 및 수정이 이루어질 수 있다. 다른 실시예는 본원에 개시된 본 발명의 명세서 및 실시를 고려하여 통상의 기술자에게 명백할 수 있다. 명세서 및 예시는 단지 예시적인 것으로 간주되어야 하는 것으로 의도되며, 본 발명의 진정한 범주 및 사상은 다음 청구범위에 의해 나타난다. 또한, 도면에 도시된 단계의 시퀀스는 단지 예시의 목적이며, 단계의 임의의 특정 시퀀스로 제한되도록 의도되지 않는다. 이와 같이, 통상의 기술자는 이러한 단계가 동일한 방법을 구현하면서 상이한 순서로 수행될 수 있음을 인식할 수 있다.
- [0266] 도면 및 명세서에서, 예시적인 실시예가 개시된다. 하지만, 이들 실시예에 대해 다수의 변형 및 수정이 이루어질 수 있다. 따라서, 특정한 용어가 이용되더라도, 제한의 목적이 아닌 일반적이고 설명적인 의미로 사용된다.

도면

도면1

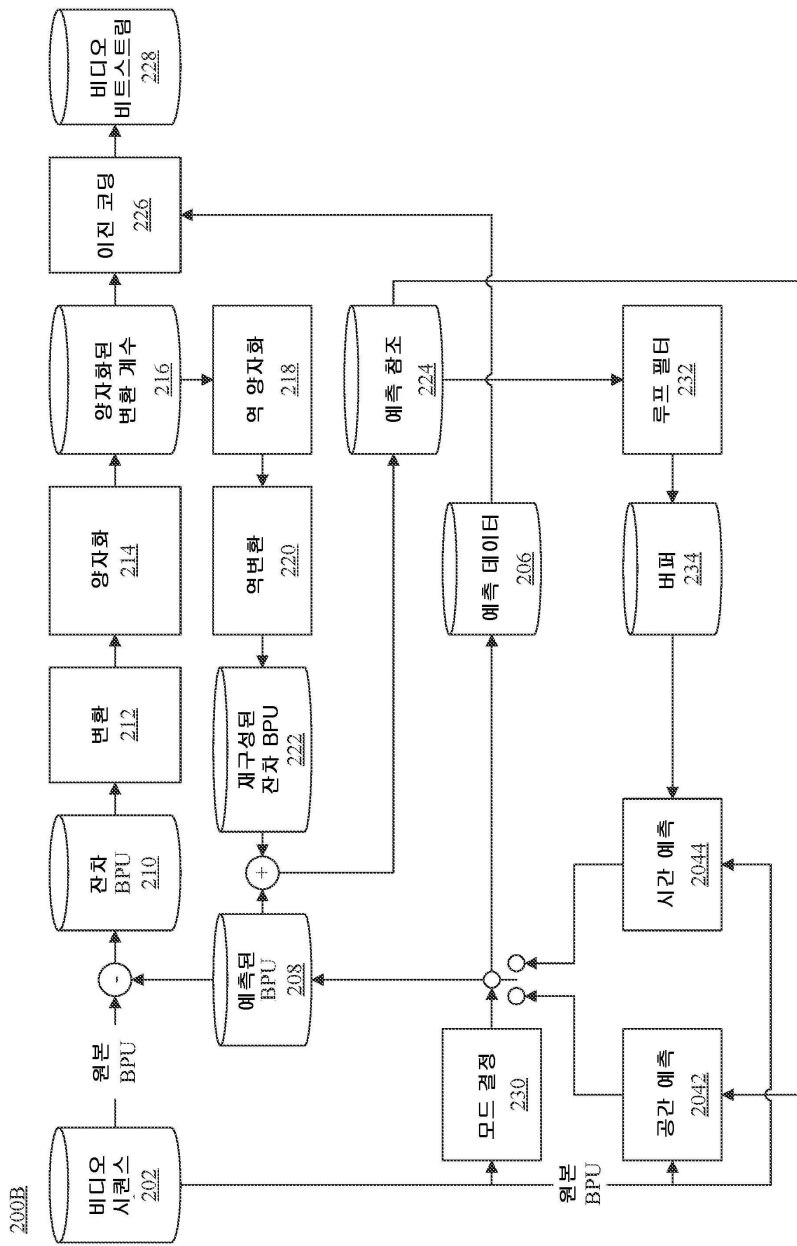


도면2a



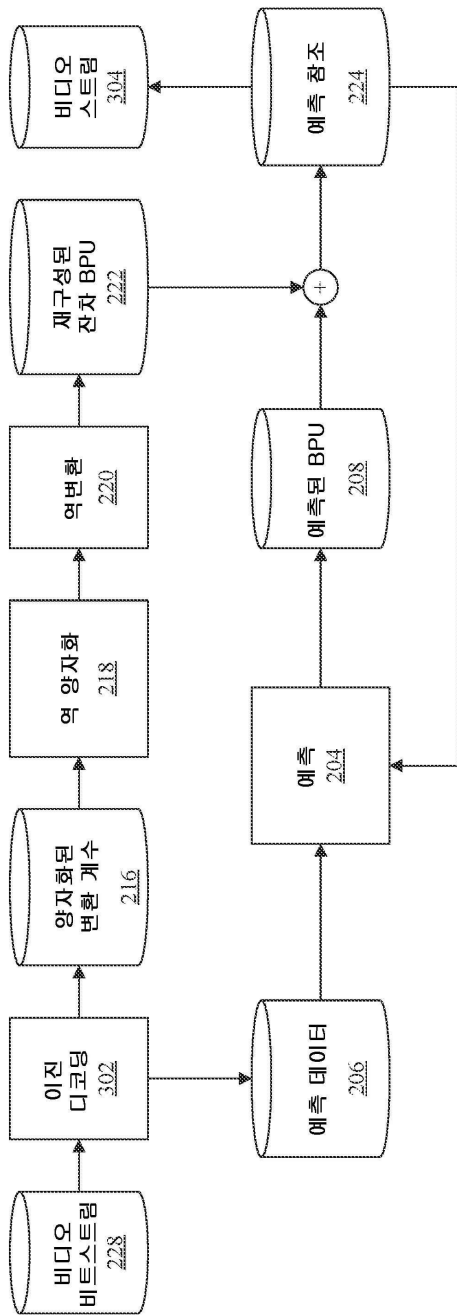
200A

도면2b



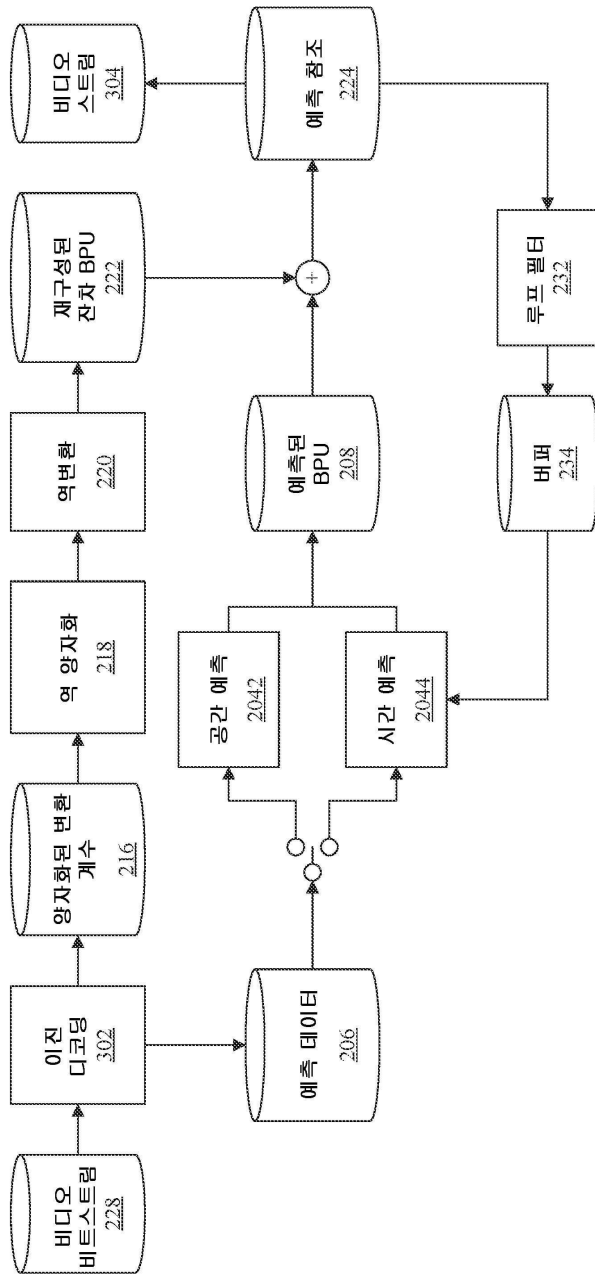
도면3a

300A

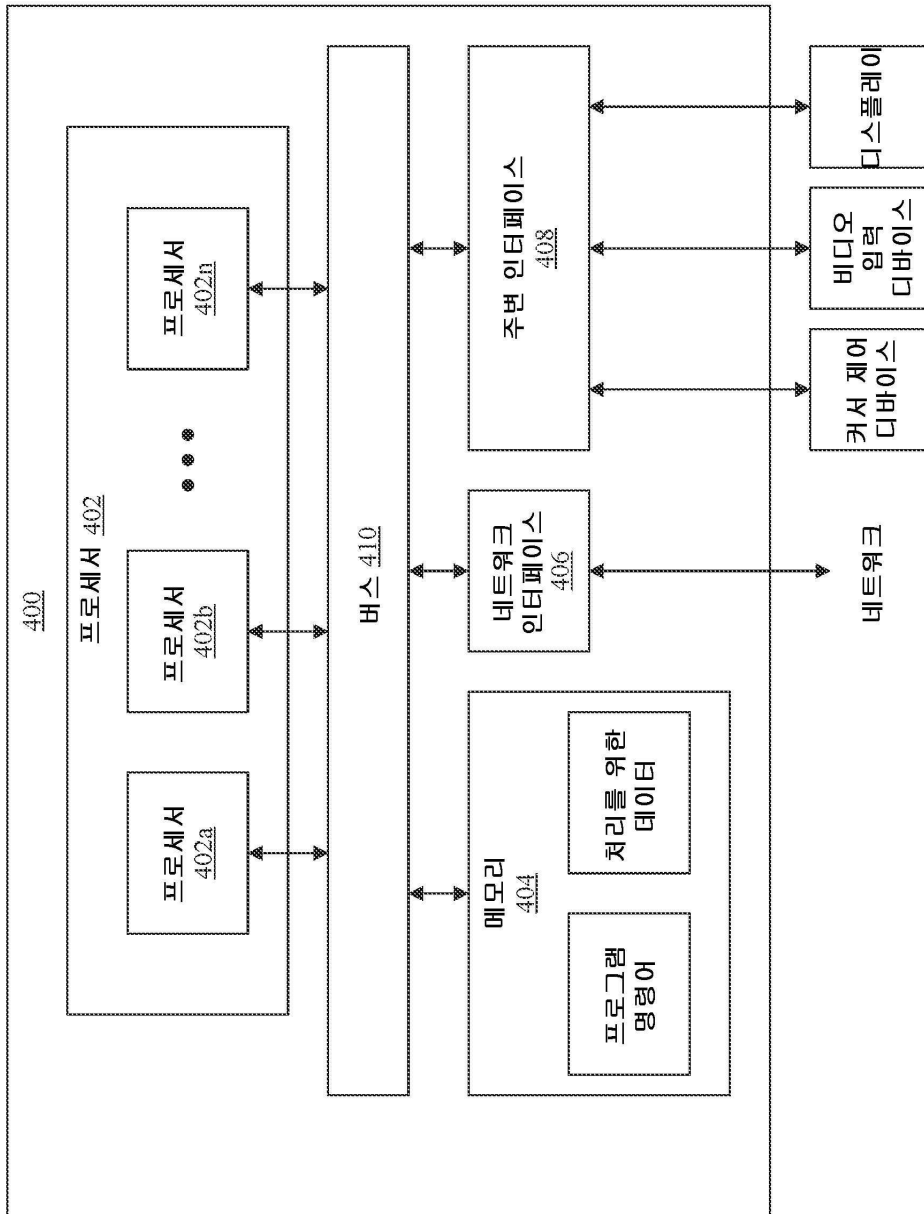


도면3b

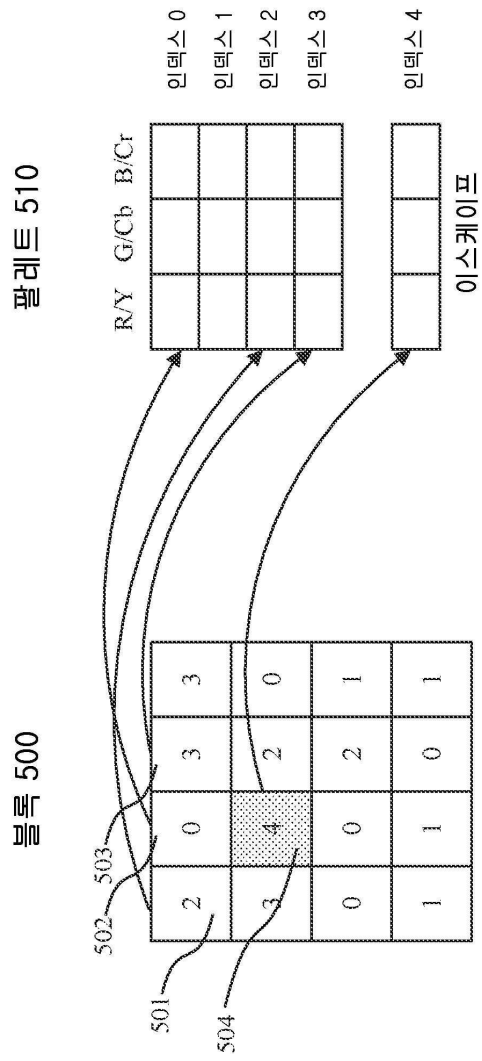
300B



도면4

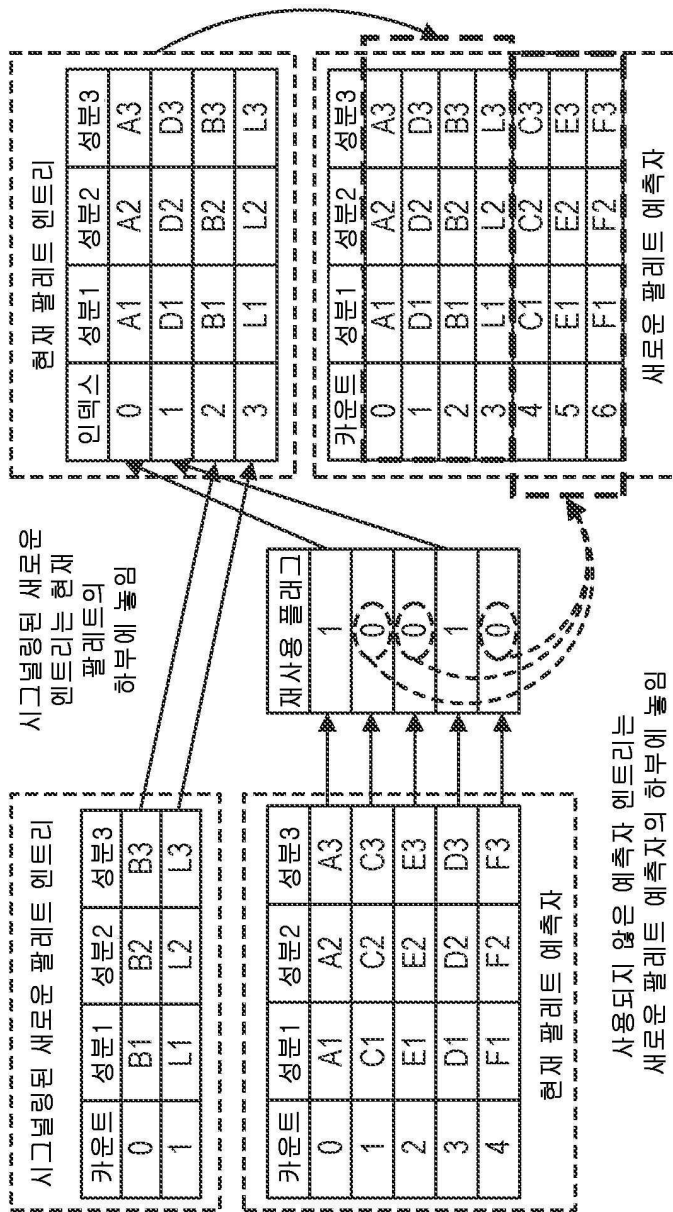


도면5



팔레트 모드에서 코딩된 블록의 예

도면6



팔레트 예측자를 업데이트하기 위한 예시적인 프로세스

도면7

표 1: 예시적인 균일한 최대 예측자 크기 및 최대 팔레트 크기

공에 및 블록의 팔레트 들 모두에 대한 최대 팔레트 예측자 크기: 63
공에 및 블록의 팔레트 표 들 모두에 대한 최대 팔레트 크기: 31

도면8

표 2: 예시적인 최대 예측자 크기 및 최대 팔레트 크기

최대 팔레트 예측자 크기	max_plt_predictor_size_joint: 63
	max_plt_predictor_size_luma: 31
	max_plt_predictor_size_chroma: 31
최대 팔레트 크기	max_plt_size_joint: 31
	max_plt_size_luma: 15
	max_plt_size_chroma: 15

도면9

표 3: 미리 정의된 최대 팔레트 예측자 크기 및 최대 팔레트 크기를 사용하기 위한 예시적인 디코딩 프로세스

```

8.4.5.3 Decoding process for palette mode
The variable localDualtree is derived as follows:
901 localDualtree = treeType != SINGLE_TREE && ( slice_type != I || ( slice_type == I &&
qibit_dual_tree_intra_flag == 0 ) ) ? 1 : 0
Depending on the value of treeType, the variables startComp, numComps and maxPltPredSize are derived as
follows:
-- If treeType is equal to SINGLE_TREE:
startComp = 0
902 numComps = 3
maxPltPredSize = max_plt_predictor_size_joint
-- Otherwise, treeType is equal to DUAL_TREE_LUMA:
startComp = 0
903 numComps = 1
maxPltPredSize = localDualtree ? max_plt_predictor_size_joint : max_plt_predictor_size_joint_luma
-- Otherwise, treeType is equal to DUAL_TREE_CHROMA:
startComp = 1
904 numComps = 2
maxPltPredSize = localDualtree ? max_plt_predictor_size_joint : max_plt_predictor_size_chroma

the value PredictorPaletteSize[ startComp ] and the array PredictorPaletteEntries are derived or modified as
follows:
for( i = 0; i < CurrentPaletteSize[ startComp ]; i++ )
for( compIdx = startComp; compIdx < ( startComp + numComps ); compIdx++ )
newPredictorPaletteEntries[ compIdx ][ i ] = CurrentPaletteEntries[ compIdx ][ i ]
905 newPredictorPaletteSize = CurrentPaletteSize[ startComp ]
for( i = 0; i < PredictorPaletteSize[ startComp ] && newPredictorPaletteSize < 63 * maxPltPredSize; i++ )
if( !PalettePredictorEntryReuseFlags[ i ] ) {
for( compIdx = startComp; compIdx < ( startComp + numComps ); compIdx++ ) (460)
newPredictorPaletteEntries[ compIdx ][ newPredictorPaletteSize ] =
PredictorPaletteEntries[ compIdx ][ i ]
newPredictorPaletteSize++
}
for( compIdx = startComp; compIdx < ( startComp + numComps ); compIdx++ )
for( i = 0; i < newPredictorPaletteSize; i++ )
PredictorPaletteEntries[ compIdx ][ i ] = newPredictorPaletteEntries[ compIdx ][ i ]
PredictorPaletteSize[ startComp ] = newPredictorPaletteSize

906 It is a requirement of bitstream conformance that the value of PredictorPaletteSize[ startComp ] shall be in the
range of 0 to 63 * maxPltPredSize, inclusive.
    
```

도면10

표 4: 미리 정의된 최대 팔레트 예측자 크기 및 최대 팔레트 크기를 사용하기 위한 예시적인 팔레트 코딩 선택스 표

		Descriptor
	palette_coding(x0, y0, cbWidth, cbHeight, treeType) {	
	startComp = (treeType == DUAL_TREE_CHROMA) ? 1 : 0	
	numComps = (treeType == SINGLE_TREE) ? 3 :	
	(treeType == DUAL_TREE_CHROMA) ? 2 : 1	
1001	maxNumPltEntries = (treeType == SINGLE_TREE) ? max_plt_size_joint :	
	(treeType == DUAL_TREE_CHROMA) ? max_plt_size_chroma :	
	max_plt_size_luma	
	palettePredictionFinished = 0	
	NumPredictedPaletteEntries = 0	
	for(predictorEntryIdx = 0; predictorEntryIdx < PredictorPaletteSize[startComp] &&	
	!palettePredictionFinished &&	
1002	NumPredictedPaletteEntries < 34 maxNumPltEntries; predictorEntryIdx++) {	
	palette_predictor_run	ae(v)
	if(palette_predictor_run != 1) {	
	if(palette_predictor_run > 1)	
	predictorEntryIdx += palette_predictor_run - 1	
	PalettePredictorEntryReuseFlags[predictorEntryIdx] = 1	
	NumPredictedPaletteEntries++	
	} else	
	palettePredictionFinished = 1	
	}	
1003	if(NumPredictedPaletteEntries < 34 maxNumPltEntries)	
	num_signalled_palette_entries	ae(v)

도면11

표 5: 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트 예측자 크기의 예시적인 도출

최대 팔레트 크기	max_plt_size_joint = sps_max_plt_size_joint - 1 + 1
	max_plt_size_luma = max_plt_size_joint >> 1
	max_plt_size_chroma = max_plt_size_joint >> 1
최대 팔레트 예측자 크기	max_plt_predictor_size_joint = max_plt_size_joint + sps_delta_max_plt_predictor_size_joint
	max_plt_predictor_size_luma = max_plt_predictor_size_joint >> 1
	max_plt_predictor_size_chroma = max_plt_predictor_size_joint >> 1

도면12

표 6: 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트 예측자 크기의 예시적인 도출

최대 팔레트 크기	$\text{max_plt_size_joint} = \text{sps_max_plt_size_joint_minus1} + 1$
	$\text{max_plt_size_chroma} = \text{max_plt_size_joint} \gg 2$
	$\text{max_plt_size_luma} = \text{max_plt_size_joint} - \text{max_plt_size_chroma}$
최대 팔레트 예측자 크기	$\text{max_plt_predictor_size_joint} = \text{max_plt_size_joint} + \text{sps_delta_max_plt_predictor_size_joint}$
	$\text{max_plt_predictor_size_chroma} = \text{max_plt_predictor_size_joint} \gg 2$
	$\text{max_plt_predictor_size_luma} = \text{max_plt_predictor_size_joint} - \text{max_plt_predictor_size_chroma}$

도면13

표 7: 예시적인 시퀀스 파라미터 세트(SPS) 선택스 표

	Descriptor
<code>seq_parameter_set_rbsp() {</code>	
<code> if(ChromaArrayType != 0)</code>	
<code> qtbtt_dual_tree_intra_flag</code>	u(1)
<code> if(chroma_format_idc == 3) {</code>	
<code> sps_palette_enabled_flag</code>	u(1)
<code> sps_act_enabled_flag</code>	u(1)
<code> }</code>	
<code> if(<i>sps_palette_enabled_flag</i>)</code>	
1301 <code> <i>sps_max_plt_size_joint_minus1</i></code>	u(6)
<code> <i>sps_delta_max_plt_predictor_size_joint</i></code>	u(6)
<code> }</code>	

도면14

표 8: 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트 예측자 크기의 예시적인 도출

최대 팔레트 크기	$\text{max_plt_size_joint} = \text{pic_max_plt_size_joint_minus1} + 1$
	$\text{max_plt_size_luma} = \text{max_plt_size_joint} \gg 1$
	$\text{max_plt_size_chroma} = \text{max_plt_size_joint} \gg 1$
최대 팔레트 예측자 크기	$\text{max_plt_predictor_size_joint} = \text{max_plt_size_joint} + \text{sps_delta_max_plt_predictor_size_joint}$
	$\text{max_plt_predictor_size_luma} = \text{max_plt_predictor_size_joint} \gg 1$
	$\text{max_plt_predictor_size_chroma} = \text{max_plt_predictor_size_joint} \gg 1$

도면15

표 9: 별개의 팔레트의 최대 팔레트 크기 및 최대 팔레트 예측자 크기의 예시적인 도출

최대 팔레트 크기	$\text{max_plt_size_joint} = \text{pic_max_plt_size_joint_minus1} + 1$
	$\text{max_plt_size_chroma} = \text{max_plt_size_joint} \gg 2$
	$\text{max_plt_size_luma} = \text{max_plt_size_joint} - \text{max_plt_size_chroma}$
최대 팔레트 예측자 크기	$\text{max_plt_predictor_size_joint} = \text{max_plt_size_joint} + \text{sps_delta_max_plt_predictor_size_joint}$
	$\text{max_plt_predictor_size_chroma} = \text{max_plt_predictor_size_joint} \gg 2$
	$\text{max_plt_predictor_size_luma} = \text{max_plt_predictor_size_joint} - \text{max_plt_predictor_size_chroma}$

도면16

표 10: 예시적인 픽처 헤더(PH) 선택스

	picture_header_rbsp() {	Descriptor
	non_reference_picture_flag	u(1)
	gdr_pic_flag	u(1)
	if(sps_scaling_list_enabled_flag) {	
	pic_scaling_list_present_flag	u(1)
	if(pic_scaling_list_present_flag)	
	pic_scaling_list_aps_id	u(3)
	}	
	if(sps_palette_enabled_flag) {	
1601	pic_max_plt_size_joint_minus1	u(6)
	pic_delta_max_plt_predictor_size_joint	u(6)
	}	
	if(picture_header_extension_present_flag) {	
	ph_extension_length	uc(v)
	for(i = 0; i < ph_extension_length ; i++)	
	ph_extension_data_byte[i]	u(8)
	}	
	rbsp_trailing_bits()	
	}	

도면17

표 11: I, P 및 B 슬라이스에 대한 최대 팔레트 크기 및 최대 팔레트 예측자 크기의 예시적인 도출

```

if (slice_type != I || qtbt_dual_tree_intra_flag == 0) // single tree case
{
    max_plt_size_joint = slice_max_plt_size_joint_minus1 + 1
    max_plt_size_luma = max_plt_size_joint >> 1 // in case of local luma tree
    max_plt_size_chroma = max_plt_size_joint >> 1 // in case of local chroma tree
    max_plt_predictor_size_joint = max_plt_size_joint + slice_delta_max_plt_predictor_size_joint
    max_plt_predictor_size_luma = 0 // never use in this case
    max_plt_predictor_size_chroma = 0 // never use in this case
}
else
{
    max_plt_size_luma = slice_max_plt_size_luma_minus1 + 1
    max_plt_size_chroma = max_plt_size_luma
    max_plt_predictor_size_luma = max_plt_size_luma + slice_delta_max_plt_predictor_size_luma
    max_plt_predictor_size_chroma = max_plt_predictor_size_luma
    max_plt_size_joint = 0 // never used for this slice
    max_plt_predictor_size_joint = 0 // never used for this slice
}
    
```

도면18

표 12 : 예시적인 슬라이스 헤더(SH) 선택스

	Descriptor
slice_header {	
slice_pic_order_cnt_lsb	u(v)
if (subpics_present_flag)	
slice_subpic_id	u(v)
if (rect_slice_flag NumTilesInPic > 1)	
slice_address	u(v)
if (rect_slice_flag && NumTilesInPic > 1)	
num_tiles_in_slice_minus1	ue(v)
slice_type	ue(v)
if (sps_palette_enabled_flag)	
if (slice_type != I qtbt_dual_tree_intra_flag == 0) {	
slice_max_plt_size_joint_minus1	u(6)
slice_delta_max_plt_predictor_size_joint	u(6)
}	
if (slice_type == 1 && qtbt_dual_tree_intra_flag) {	
slice_max_plt_size_luma_minus1	u(6)
slice_delta_max_plt_predictor_size_luma	u(6)
}	
}	
if (slice_header_extension_present_flag) {	
slice_header_extension_length	ue(v)
for (i = 0; i < slice_header_extension_length; i++)	
slice_header_extension_data_byte[i]	u(8)
}	
byte_alignment()	
}	

1801

도면19

1900

