(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0134338
(43) 공개일자 2021년11월09일

(51) 국제특허분류(Int. Cl.)
H04N 19/132 (2014.01) *H04N 19/105* (2014.01)
H04N 19/11 (2014.01) *H04N 19/117* (2014.01)
H04N 19/157 (2014.01) *H04N 19/159* (2014.01)
H04N 19/176 (2014.01) *H04N 19/182* (2014.01)
H04N 19/186 (2014.01) *H04N 19/54* (2014.01)
H04N 19/86 (2014.01)
(52) CPC특허분류
H04N 19/132 (2015.01)
H04N 19/105 (2015.01)
(21) 출원번호 10-2021-7028603
(22) 출원일자(국제) 2020년02월26일
심사청구일자 없음
(85) 번역문제출일자 2021년09월06일
(86) 국제출원번호 PCT/US2020/019952
(87) 국제공개번호 WO 2020/190469
국제공개일자 2020년09월24일
(30) 우선권주장
62/819,368 2019년03월15일 미국(US)
16/800,864 2020년02월25일 미국(US)

(71) 출원인
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
왕 홍타오
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
치엔 웨이-정
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(뒷면에 계속)
(74) 대리인
특허법인코리아나

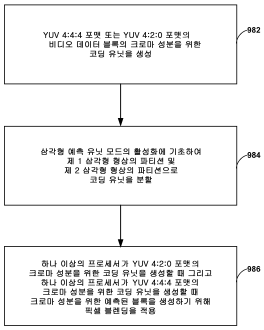
전체 청구항 수 : 총 21 항

(54) 발명의 명칭 **상이한 크로마 포맷을 사용한 필터링되지 않은 참조 샘플로의 비디오 코딩**

(57) 요 약

비디오 데이터를 프로세싱하기 위한 디바이스는 비디오 데이터를 저장하도록 구성된 메모리 및 회로에 구현되는 하나 이상의 프로세서를 포함한다. 하나 이상의 프로세서는 비디오 데이터의 화상의 영역을 위해 필터링되지 않은 참조 샘플을 획득하도록 구성된다. 하나 이상의 프로세서는 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷의 크로마 샘플들을 위해 필터링되지 않은 참조 샘플들의 인트라 참조 샘플 평활화를 비활성화하도록 구성된다. 하나 이상의 프로세서는 또한, YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위해 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성된다.

대 표 도



(52) CPC특허분류
H04N 19/11 (2015.01)
H04N 19/117 (2015.01)
H04N 19/157 (2015.01)
H04N 19/159 (2015.01)
H04N 19/176 (2015.01)
H04N 19/182 (2015.01)
H04N 19/186 (2015.01)
H04N 19/54 (2015.01)
H04N 19/86 (2015.01)
(72) 발명자
세레긴 바딤
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
카르체비츠 마르타
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

명 세 서

청구범위

청구항 1

비디오 데이터를 프로세싱하는 방법으로서,

회로에 구현된 하나 이상의 프로세서에 의해, 화상의 영역을 위해 필터링되지 않은 참조 샘플을 획득하는 단계로서, 상기 하나 이상의 프로세서는 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷의 크로마 샘플을 위해 상기 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하도록 구성되는, 상기 참조 샘플을 획득하는 단계; 및

상기 YUV 4:2:0 포맷의 상기 크로마 성분을 생성할 때 그리고 상기 YUV 4:4:4 포맷의 상기 크로마 성분을 생성할 때 상기 필터링되지 않은 참조 샘플에 기초하여 상기 화상의 블록을 위해 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 상기 하나 이상의 프로세서에 의해, 생성하는 단계

를 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 2

제 1 항에 있어서,

루마 샘플을 생성하기 위해 인트라 참조 샘플 평활화를 활성화하는 것을 결정하는 것에 응답하여:

상기 하나 이상의 프로세서에 의해, 상기 필터링되지 않은 참조 샘플에 대해 하나 이상의 필터 동작을 수행하여 필터링된 참조 샘플을 생성하는 단계; 및

상기 하나 이상의 프로세서에 의해, 상기 필터링된 참조 샘플에 기초하여 상기 화상의 블록을 위한 루마 샘플을, 인트라 예측을 사용하여, 생성하는 단계

를 더 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 3

제 2 항에 있어서,

상기 하나 이상의 필터 동작을 수행하는 단계는 상기 필터링되지 않은 참조 샘플에 대해 하나 이상의 디블로킹 동작을 수행하는 단계를 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 4

제 1 항에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 비디오 데이터 블록을 위한 잔차 블록을 디코딩하는 단계; 및

상기 하나 이상의 프로세서에 의해, 상기 예측 블록 및 상기 잔차 블록을 조합하여 상기 비디오 데이터 블록을 위해 필터링되지 않은 재구성 블록을 디코딩하는 단계를 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 5

제 4 항에 있어서,

상기 하나 이상의 프로세서에 의해, 디코딩된 화상 버퍼에서 상기 필터링되지 않은 재구성 블록을 저장하는 단계를 더 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 6

제 4 항에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 비디오 데이터 블록을 위해 필터링된 재구성 블록을 생성하는 단계로서, 상기 필터링된 재구성 블록을 생성하는 단계는 상기 필터링되지 않은 재구성 블록에 대해 하나 이상의 필터

동작을 수행하는 단계를 포함하는, 상기 필터링된 재구성 블록을 생성하는 단계; 및

상기 하나 이상의 프로세서에 의해, 디코딩된 화상 버퍼에서 상기 필터링된 재구성 블록을 저장하는 단계를 더 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 7

제 1 항에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 비디오 데이터 블록과 상기 예측 블록 간의 차이에 기초하여 상기 비디오 데이터 블록을 위한 잔차 블록을 생성하는 단계; 및

상기 하나 이상의 프로세서에 의해, 상기 잔차 블록을 인코딩하는 단계를 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 8

제 7 항에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 예측 블록 및 상기 잔차 블록을 조합하여 상기 비디오 데이터 블록을 위해 필터링되지 않은 재구성 블록을 디코딩하는 단계; 및

상기 하나 이상의 프로세서에 의해, 디코딩된 화상 버퍼에서 상기 필터링되지 않은 재구성 블록을 저장하는 단계를 더 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 9

제 7 항에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 예측 블록 및 상기 잔차 블록을 조합하여 상기 비디오 데이터 블록을 위해 필터링되지 않은 재구성 블록을 디코딩하는 단계.

상기 하나 이상의 프로세서에 의해, 상기 비디오 데이터 블록을 위해 필터링된 재구성 블록을 생성하는 단계로서, 상기 필터링된 재구성 블록을 생성하는 단계는 상기 필터링되지 않은 재구성 블록에 대해 하나 이상의 필터 동작을 수행하는 단계를 포함하는, 상기 필터링된 재구성 블록을 생성하는 단계; 및

상기 하나 이상의 프로세서에 의해, 디코딩된 화상 버퍼에서 상기 필터링된 재구성 블록을 저장하는 단계를 더 포함하는, 비디오 데이터를 프로세싱하는 방법.

청구항 10

비디오 데이터를 프로세싱하기 위한 디바이스로서,

비디오 데이터를 저장하도록 구성된 메모리; 및

회로에 구현되는 하나 이상의 프로세서를 포함하고

상기 하나 이상의 프로세서는

비디오 데이터의 화상의 영역을 위해 필터링되지 않은 참조 샘플을 획득하는 것으로서, 상기 하나 이상의 프로세서는 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷의 크로마 샘플을 위해 상기 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하도록 구성되는, 상기 참조 샘플을 획득하고;

상기 YUV 4:2:0 포맷의 상기 크로마 성분을 생성할 때 그리고 상기 YUV 4:4:4 포맷의 상기 크로마 성분을 생성할 때 상기 필터링되지 않은 참조 샘플에 기초하여 상기 화상의 블록을 위해 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 11

제 10 항에 있어서,

상기 하나 이상의 프로세서는, 또한, 루마 샘플을 생성하기 위해 인트라 참조 샘플 평활화를 활성화하는 것을 결정하는 것에 응답하여:

상기 필터링되지 않은 참조 샘플에 대해 하나 이상의 필터 동작을 수행하여 필터링된 참조 샘플을 생성하고;

상기 필터링된 참조 샘플에 기초하여 상기 화상의 블록을 위해 루마 샘플을, 인트라 예측을 사용하여, 생성하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 12

제 11 항에 있어서,

상기 하나 이상의 필터 동작을 수행하기 위하여, 상기 하나 이상의 프로세서는 상기 필터링되지 않은 참조 샘플에 대해 하나 이상의 디블로킹 동작을 수행하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 13

제 10 항에 있어서,

상기 하나 이상의 프로세서는 또한

상기 비디오 데이터 블록을 위한 잔차 블록을 디코딩하고;

상기 예측 블록 및 상기 잔차 블록을 조합하여 상기 비디오 데이터 블록을 위해 필터링되지 않은 재구성 블록을 디코딩하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 14

제 13 항에 있어서,

상기 하나 이상의 프로세서는 또한, 디코딩된 화상 버퍼에서 상기 필터링되지 않은 재구성 블록을 저장하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 15

제 13 항에 있어서,

상기 하나 이상의 프로세서는 또한

상기 비디오 데이터 블록을 위해 필터링된 재구성 블록을 생성하는 것으로서, 상기 필터링된 재구성 블록을 생성하는 것은 상기 필터링되지 않은 재구성 블록에 대해 하나 이상의 필터 동작을 수행하는 것을 포함하는, 상기 필터링된 재구성 블록을 생성하고;

디코딩된 화상 버퍼에서 상기 필터링된 재구성 블록을 저장하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 16

제 10 항에 있어서,

상기 하나 이상의 프로세서는 또한

상기 비디오 데이터 블록과 상기 예측 블록 간의 차이에 기초하여 상기 비디오 데이터 블록을 위한 잔차 블록을 생성하고;

상기 잔차 블록을 인코딩하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 17

제 16 항에 있어서,

상기 하나 이상의 프로세서는 또한

상기 예측 블록 및 상기 잔차 블록을 조합하여 상기 비디오 데이터 블록을 위해 필터링되지 않은 재구성 블록을 디코딩하고;

디코딩된 화상 버퍼에 상기 필터링되지 않은 재구성 블록을 저장하도록 구성되는, 비디오 데이터를 프로세싱하

기 위한 디바이스.

청구항 18

제 16 항에 있어서,

상기 하나 이상의 프로세서는 또한

상기 예측 블록 및 상기 잔차 블록을 조합하여 상기 비디오 데이터 블록을 위해 필터링되지 않은 재구성 블록을 디코딩하고;

상기 비디오 데이터 블록을 위해 필터링된 재구성 블록을 생성하는 것으로서, 상기 필터링된 재구성 블록을 생성하는 것은 상기 필터링되지 않은 재구성 블록에 대해 하나 이상의 필터 동작을 수행하는 것을 포함하는, 상기 필터링된 재구성 블록을 생성하고;

디코딩된 화상 버퍼에서 상기 필터링된 재구성 블록을 저장하도록 구성되는, 비디오 데이터를 프로세싱하기 위한 디바이스.

청구항 19

제 10 항에 있어서,

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

청구항 20

명령들이 저장된 컴퓨터 판독가능 저장 매체로서,

상기 명령들은 실행될 때 하나 이상의 프로세서로 하여금

비디오 데이터의 화상의 영역을 위해 필터링되지 않은 참조 샘플을 획득하게 하는 것으로서, 상기 명령들은 또한, 상기 하나 이상의 프로세서로 하여금 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷의 크로마 샘플을 위한 상기 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하게 하는, 상기 참조 샘플을 획득하게 하고;

상기 YUV 4:2:0 포맷의 상기 크로마 성분을 생성할 때 그리고 상기 YUV 4:4:4 포맷의 상기 크로마 성분을 생성할 때 상기 필터링되지 않은 참조 샘플에 기초하여 상기 화상의 블록을 위해 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하게 하는, 명령들이 저장된 컴퓨터 판독가능 저장 매체.

청구항 21

비디오 데이터를 코딩하기 위한 디바이스로서,

화상의 영역을 위해 필터링되지 않은 참조 샘플을 획득하는 수단; 및

YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 상기 필터링되지 않은 참조 샘플에 기초하여 상기 화상의 블록을 위해 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하는 수단

을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

발명의 설명

기 술 분 야

[0001] 본 출원은 2019 년 3 월 15 일 출원된 미국 특허 가출원 제 62/819,368 호의 이익을 주장하는, 2019 년 2 월 25 일자 출원된 미국 특허 출원 제 16/800,864 호에 대한 우선권을 주장하며, 이들 출원의 전체 내용은 참조로서 본 명세서에 원용된다.

[0002] 본 개시는 비디오 인코딩 및 비디오 디코딩에 관한 것이다.

배 경 기 술

[0003] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 다이렉트 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, PDA(personal digital assistant)들, 랩탑 또는 데스크탑 컴퓨터들, 태블릿 컴퓨터들, 전자책 단말기, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 소위 "스마트 폰", 원격 화상회의 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 포함될 수 있다. 디지털 비디오 디바이스들은, MPEG-2, MPEG-4, ITU-T H.263 또는 ITU-T H.264/MPEG-4, Part 10, AVC (Advanced Video Coding), ITU-T H.265/HEVC (High Efficiency Video Coding) 에 의해 정의되는 표준들, 및 그러한 표준들의 확장들에서 설명된 것들과 같은 비디오 코딩 기법들을 구현한다. 비디오 디바이스들은 그러한 비디오 코딩 기술들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0004] 비디오 코딩 기법들은 비디오 시퀀스들에 내재한 리던던시를 감소 또는 제거하기 위해 공간 (화상내) 예측 및/또는 시간 (화상간) 예측을 포함한다. 블록 기반 비디오 코딩을 위해, 비디오 슬라이스 (예를 들어, 비디오 화상 또는 비디오 화상의 일부) 는 비디오 블록들로 파티셔닝될 수도 있고, 이 비디오 블록들은 코딩 트리 유닛들 (CTU들), 코딩 유닛들 (CU들) 및/또는 코딩 노드들로도 지칭될 수도 있다. 화상의 인트라 코딩된 (I) 슬라이스에서의 비디오 블록들은 동일한 화상내의 이웃하는 블록들에 있는 참조 샘플들에 대한 공간 예측을 사용하여 인코딩된다. 화상의 인터 코딩된 (P 또는 B) 슬라이스내의 비디오 블록들은 동일한 화상내의 이웃하는 블록들에 있는 참조 샘플들에 대한 공간 예측, 또는 다른 참조 화상들 내의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 화상들은 프레임들로 지칭될 수도 있고, 참조 화상들은 참조 프레임들로 지칭될 수도 있다.

발명의 내용

[0005] **개요**

[0006] 일반적으로, 본 개시는 비디오 코덱에서의 상이한 크로마 포맷을 지원하기 위한 기법들을 설명한다.

[0007] 하나의 예에서, 비디오 데이터를 프로세싱하는 방법은 회로에 구현된 하나 이상의 프로세서에 의해, YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하는 단계; 하나 이상의 프로세서에 의해, 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하는 단계; 및 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때, 하나 이상의 프로세서에 의해, 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하는 단계로서, 상기 픽셀 블렌딩을 적용하는 단계는 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 (collocated) 모션 보상 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하는 단계를 포함하는, 상기 픽셀 블렌딩을 적용하는 단계를 포함한다.

[0008] 다른 예에서, 비디오 데이터를 프로세싱하기 위한 디바이스는 비디오 데이터를 저장하도록 구성된 메모리; 및 회로에 구현되는 하나 이상의 프로세서들을 포함한다. 하나 이상의 프로세서들은 다음과 같이 구성된다: YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하고; 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하고; 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하는 것으로서, 픽셀 블렌딩을 적용하기 위해, 하나 이상의 프로세서는 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하도록 구성된다.

[0009] 일례에서, 컴퓨터 판독가능 저장 매체는, 실행될 때, 하나 이상의 프로세서로 하여금 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하게 하고; 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하게 하고; 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그

리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하게 하는 것으로서, 하나 이상의 프로세서로 하여금 픽셀 블렌딩을 적용하게 하는 명령들은 하나 이상의 프로세서로 하여금 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하게 하는, 저장된 명령들을 포함한다.

[0010] 또 다른 예에서, 비디오 데이터를 코딩하기 위한 디바이스는, YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하는 수단; 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하는 수단; 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하는 수단으로서, 픽셀 블렌딩을 적용하는 수단은 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하는 수단을 포함하는, 상기 픽셀 블렌딩을 적용하는 수단을 포함한다

[0011] 하나의 예에서, 비디오 데이터를 프로세싱하는 방법은 다음을 포함한다: 회로에 구현된 하나 이상의 프로세서에 의해, 화상의 영역에 대한 필터링되지 않은 참조 샘플을 획득하는 단계로서, 상기 하나 이상의 프로세서는 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하도록 구성되는, 상기 참조 샘플을 획득하는 단계; 및 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 하나 이상의 프로세서에 의해, 생성하는 단계.

[0012] 또 다른 예에서, 비디오 데이터를 프로세싱하기 위한 디바이스는, 비디오 데이터를 저장하도록 구성된 메모리; 및 회로에 구현되는 하나 이상의 프로세서를 포함하고, 상기 하나 이상의 프로세서는 비디오 데이터의 화상의 영역에 대한 필터링되지 않은 참조 샘플을 획득하는 것으로서, 상기 하나 이상의 프로세서는 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하도록 구성되는, 상기 참조 샘플을 획득하고; YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성된다.

[0013] 일례에서, 컴퓨터 판독가능 저장 매체는, 실행될 때, 하나 이상의 프로세서로 하여금 비디오 데이터의 화상의 영역에 대한 필터링되지 않은 참조 샘플을 획득하는 것으로서, 상기 명령들은 또한, 상기 하나 이상의 프로세서로 하여금 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하게 하는, 상기 참조 샘플을 획득하고; YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하게 하는, 저장된 명령들을 포함한다.

[0014] 또 다른 예에서, 비디오 데이터를 코딩하기 위한 디바이스는 화상의 영역에 대한 필터링되지 않은 참조 샘플을 획득하는 수단; 및 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측된 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하는 수단을 포함한다.

[0015] 하나 이상의 예들의 상세는 첨부 도면 및 아래의 설명에 제시되어 있다. 다른 특징, 목적 및 이점들은 상세한 설명, 도면, 및 청구항들로부터 분명해질 것이다.

도면의 간단한 설명

[0016] 도 1 은 본 개시의 기법들을 수행할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록 도이다.

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

도 3 는 본 개시의 기법들을 수행할 수도 있는 예시적인 비디오 인코더를 예시하는 블록 도이다.

도 4 은 본 개시의 기법들을 수행할 수도 있는 예시적인 비디오 디코더를 예시하는 블록 도이다.

도 5a 는 본 개시의 기법에 따라 인터 예측에 기초하여 코딩 유닛을 제 1 삼각형 형상 파티션 및 제 2 삼각형 형상 파티션으로 분할하는 제 1 예를 예시하는 개념도이다.

도 5b 는 본 개시의 기법에 따라 인터 예측에 기초하여 코딩 유닛을 제 1 삼각형 형상 파티션 및 제 2 삼각형 형상 파티션으로 분할하는 제 2 예를 예시하는 개념도이다.

도 6 은 본 개시의 기술들에 따른, 하나의 가중치 세트를 사용한 픽셀 블렌딩을 예시하는 개념도이다.

도 7a 는 본 개시의 기술에 따른, 도 6의 픽셀 블렌딩을 위한 제 1 삼각형 형상의 파티션의 추가 세부사항을 예시하는 개념도이다.

도 7b 는 본 개시의 기술에 따른, 도 6의 픽셀 블렌딩을 위한 제 2 삼각형 형상의 파티션의 추가 세부사항을 예시하는 개념도이다.

도 7c 는 본 개시의 기술에 따른, 도 6의 픽셀 블렌딩을 사용하여 형성된 블록의 추가 세부사항을 예시하는 개념도이다.

도 8 는 본 개시의 기법들에 따른 현재 블록을 인코딩하기 위한 예시적인 방법을 예시하는 흐름도이다.

도 9 은 본 개시의 기법들에 따른 비디오 데이터의 현재 블록을 디코딩하기 위한 예시적인 방법을 예시하는 흐름도이다.

도 10 은 본 개시의 기법들에 따른, YUV 4:2:0 샘플링 포맷을 위한 가중치 세트를 사용한 비디오 데이터의 블록을 위한 크로마 성분들의 픽셀 블렌딩을 위한 예시적인 프로세스를 예시하는 흐름도이다.

도 11은 인트라 예측을 이용하여 예측 블록의 크로마 샘플들을 생성하기 위한 예시적인 프로세스를 예시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0017] **상세한 설명**

[0018] 아래에서 더 상세하게 설명되는 바와 같이, 본 개시는 상이한 포맷의 크로마 성분들의 인코딩 및/또는 디코딩을 개선하기 위한 예시적인 기술들을 설명한다. 예를 들어, 삼각형 예측 유닛 (PU) 모드를 사용할 때, 비디오 코더 (예를 들어, 비디오 인코더 또는 비디오 디코더) 는 PU를 2개의 삼각형 형상의 파티션으로 분할할 수도 있다. 이 예에서, 비디오 코더는 모션 보상된 삼각형 형상의 파티션을 생성하고 크로마 포맷에 기초한 가중치 세트를 사용하여 모션 보상된 삼각형 형상의 파티션을 결합할 수도 있다. 예를 들어, 비디오 코더는 크로마 성분이 YUV 4:4:4 포맷일 때 제 1 가중치 세트를 사용하여 픽셀 블렌딩을 적용할 수도 있고, 크로마 성분이 YUV 4:2:0 포맷일 때 제 1 가중치 세트와는 상이한, 제 2 가중치 세트를 적용할 수도 있다. 그러나, 상이한 크로마 포맷에 대해 상이한 가중치 세트를 사용하는 것은 크로마 성분을 생성하는 복잡성을 증가시킬 수도 있다.

[0019] 본 개시의 기법에 따르면, 비디오 코더 (예를 들어, 비디오 인코더 또는 비디오 디코더) 는 상이한 크로마 포맷에 대한 크로마 성분들의 처리를 통일 (unify) 하도록 구성될 수도 있다. 예를 들어, 비디오 코더는 크로마 성분이 YUV 4:4:4 포맷일 때 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하고 크로마 성분이 YUV 4:2:0 포맷일 때 YUV 4:2:0 포맷을 위한 가중치 세트를 적용할 수도 있다.

[0020] 일부 예에서, 비디오 코더 (예를 들어, 비디오 인코더 또는 비디오 디코더) 는 인트라 예측이 사용될 때 그리고 크로마 성분들이 YUV 4:2:0 포맷일 때 필터링되지 않은 참조 샘플을 사용하여 비디오 데이터의 블록을 위한 예측 블록의 크로마 샘플들을 생성하도록 구성될 수도 있다. 이 예에서, 비디오 코더는 인트라 예측이 사용될 때 그리고 크로마 성분들이 YUV 4:4:4 포맷일 때 필터링된 참조 샘플을 사용하여 비디오 데이터의 블록을 위한 예측 블록의 크로마 샘플들을 생성하도록 구성될 수도 있다. 그러나, 상이한 크로마 포맷에 대해 상이한 참조 샘플에 사용하는 것은 크로마 성분을 생성하는 복잡성을 증가시킬 수도 있다.

[0021] 본 개시의 기법에 따르면, 비디오 코더 (예를 들어, 비디오 인코더 또는 비디오 디코더) 는 상이한 크로마 포맷에 대한 크로마 성분들의 처리를 통일 (unify) 하도록 구성될 수도 있다. 예를 들어, 비디오 코더는 크로마 성분이 YUV 4:2:0 포맷이고 크로마 성분이 YUV 4:4:4 포맷일 때 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 예를 들어, 비디오 코더는 크로마 성분이 YUV 4:2:0 포맷일 때 그리고 크로마 성분이 YUV 4:4:4 포맷일 때 필터링되지 않은 참조 샘플을 사용하여 비디오 데이터의 블록을 위한 예측 블록의 크로마 샘플

들을 생성할 수도 있다.

[0022] 크로마 성분의 처리를 통일하기 위해 여기에 설명된 기술은 HEVC (High Efficiency Video Coding) 와 같은 기존 비디오 코덱 또는 VVC (Versatile Video Coding) 와 같은 개발 중인 비디오 코덱 중 어느 것에 적용되거나 또는 임의의 미래의 비디오 코딩 표준에서의 효율적인 코딩 도구일 수도 있다. 가중 예측과 관련된 기술의 예가 HEVC 및 VVC (Versatile Video Coding) 에서의 진행 중인 작업과 관련하여 설명되지만, 여기에 설명된 기술은 다른 기존 비디오 코덱 및/또는 미래의 비디오 코딩 표준에 적용될 수도 있다.

[0023] 도 1 은 본 개시의 기술을 수행할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (100) 을 예시하는 블록도이다. 본 개시의 기법들은 일반적으로 비디오 데이터를 코딩 (인코딩 및/또는 디코딩) 하는 것에 관한 것이다. 일반적으로, 비디오 데이터는 비디오를 프로세싱하기 위한 임의의 데이터를 포함한다. 따라서, 비디오 데이터는 원시, 코딩되지 않은 비디오, 인코딩된 비디오, 디코딩된 (예를 들어, 재구성된) 비디오, 및 비디오 메타데이터, 이를 테면 시그널링 데이터를 포함할 수도 있다.

[0024] 도 1 에 도시된 바와 같이, 시스템 (100) 은, 이 예에서 목적지 디바이스 (116) 에 의해 디코딩 및 디스플레이될 인코딩된 비디오 데이터를 제공하는 소스 디바이스 (102) 를 포함한다. 특히, 소스 디바이스 (102) 는 비디오 데이터를 컴퓨터 판독가능 매체 (110) 를 통해 목적지 디바이스 (116) 에 제공한다. 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋탑 박스들, 전화기 핸드셋들, 예컨대 스마트폰들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함한, 광범위한 디바이스들 중 임의의 것을 포함할 수도 있다. 일부 경우에, 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 무선 통신을 위해 갖추어질 수도 있고, 따라서 무선 통신 디바이스로서 지칭될 수도 있다.

[0025] 도 1 의 예에서, 소스 디바이스 (102) 는 비디오 소스 (104), 메모리 (106), 비디오 인코더 (200), 및 출력 인터페이스 (108) 를 포함한다. 목적지 디바이스 (116) 는 입력 인터페이스 (122), 비디오 디코더 (300), 메모리 (120), 및 디스플레이 디바이스 (118) 를 포함한다. 본 개시에 따르면, 소스 디바이스 (102) 의 비디오 인코더 (200) 및 목적지 디바이스 (116) 의 비디오 디코더 (300) 는 상이한 크로마 포맷에 대해 가중치 정보를 통일하기 위한 기법들을 적용하도록 구성될 수도 있다. 따라서, 소스 디바이스 (102) 는 비디오 인코딩 디바이스의 예를 나타내는 한편, 목적지 디바이스 (116) 는 비디오 디코딩 디바이스의 예를 나타낸다. 다른 예에서, 소스 디바이스 및 목적지 디바이스는 다른 컴포넌트들 또는 배열들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (102) 는 외부 카메라와 같은 외부 비디오 소스로부터 비디오 데이터를 수신할 수도 있다. 마찬가지로, 목적지 디바이스 (116) 는 통합된 디스플레이 디바이스를 포함하기 보다는, 외부 디스플레이 디바이스와 인터페이스로 접속할 수도 있다.

[0026] 도 1에 도시된 바와 같은 시스템 (100) 은 하나의 예일뿐이다. 일반적으로, 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스는, 제 1 크로마 포맷을 위한 가중치 정보를 결정하고, 제 1 크로마 포맷을 위한 가중치 정보에 대응하도록 제 1 크로마 포맷과 상이한 제 2 크로마 포맷을 위한 가중치 정보를 결정하고, 제 1 크로마 포맷을 위한 가중치 정보에 기초하여 예측 정보를 생성하기 위한 기술을 수행할 수도 있다. 소스 디바이스 (102) 및 목적지 디바이스 (116) 는, 소스 디바이스 (102) 가 목적지 디바이스 (116) 로의 송신을 위해 코딩된 비디오 데이터를 생성하는 그러한 코딩 디바이스들의 예들일 뿐이다. 본 개시는 "코딩" 디바이스를 데이터의 코딩 (인코딩 및/또는 디코딩) 을 수행하는 디바이스로서 지칭한다. 따라서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 코딩 디바이스들, 특히 각각 비디오 인코더 및 비디오 디코더의 예들을 나타낸다. 일부 예에서, 디바이스들 (102, 116) 은 디바이스들 (102, 116) 의 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 따라서, 시스템 (100) 은 예를 들어, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 전화를 위해, 비디오 디바이스들 (102, 116) 간의 일방향 또는 양방향 비디오 송신을 지원할 수도 있다.

[0027] 일반적으로, 비디오 소스 (104) 는 비디오 데이터 (즉, 원시, 코딩되지 않은 비디오 데이터) 의 소스를 나타내며, 화상들을 위한 데이터를 인코딩하는 비디오 인코더 (200) 에 비디오 데이터의 순차적인 일련의 화상들 (또한 "프레임들" 로도 지칭됨) 을 제공한다. 소스 디바이스 (102) 의 비디오 소스 (104) 는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 원시 비디오를 포함하는 비디오 아카이브, 및/또는 비디오 컨텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스를 포함할 수도 있다. 추가의 대안으로서, 비디오 소스 (104) 는 컴퓨터 그래픽 기반 데이터를 소스 비디오로서, 또는 라이브 비디오, 아카이브된 비디오, 및 컴퓨터 생성된 비디오의 조합으로서 생성할 수도 있다. 각각의 경우에 있어서, 비디오 인코더 (200) 는

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

[0028] 소스 디바이스 (102) 의 메모리 (106) 및 목적지 디바이스 (116) 의 메모리 (120) 는 범용 메모리들을 나타낸다. 일부 예에서, 메모리들 (106, 120) 은 원시 비디오 데이터, 예컨대, 비디오 소스 (104) 로부터의 원시 비디오 및 비디오 디코더 (300) 로부터의 원시, 디코딩된 비디오 데이터를 저장할 수도 있다. 추가적으로, 또는 대안적으로, 메모리들 (106, 120) 은 예를 들어, 각각 비디오 인코더 (200) 및 비디오 디코더 (300) 에 의해 실행가능한 소프트웨어 명령들을 저장할 수도 있다. 이 예에서는 비디오 인코더 (200) 및 비디오 디코더 (300) 와 따로 도시되지만, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 또한 기능적으로 유사하거나 또는 동등한 목적들을 위한 내부 메모리들을 포함할 수도 있음을 이해해야 한다. 또한, 메모리들 (106, 120) 은 예를 들어, 비디오 인코더 (200) 로부터 출력되고 비디오 디코더 (300) 에 입력되는 인코딩된 비디오 데이터를 저장할 수도 있다. 일부 예들에서, 메모리들 (106, 120) 의 부분들은 예를 들어, 원시, 디코딩된, 및/또는 인코딩된 비디오 데이터를 저장하기 위해 하나 이상의 비디오 버퍼들로서 할당될 수도 있다.

[0029] 컴퓨터 판독 가능 매체 (110) 는 인코딩된 비디오 데이터를 소스 디바이스 (102) 로부터 목적지 디바이스 (116) 로 전송할 수 있는 임의의 유형의 매체 또는 디바이스를 나타낼 수도 있다. 일례에서, 컴퓨터 판독 가능 매체 (110) 는, 소스 디바이스 (102) 로 하여금, 인코딩된 비디오 데이터를 직접 목적지 디바이스 (116) 에 실시간으로, 예를 들어 무선 주파수 네트워크 또는 컴퓨터 기반 네트워크를 통해 송신할 수 있게 하기 위한 통신 매체를 나타낸다. 무선 통신 프로토콜과 같은 통신 표준에 따라, 출력 인터페이스 (108) 는 인코딩된 비디오 데이터를 포함하는 송신 신호를 변조할 수도 있고, 입력 인터페이스 (122) 는 수신된 송신 신호를 변조할 수도 있다. 통신 매체는 무선 주파수 (radio frequency; RF) 스펙트럼 또는 하나 이상의 물리 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷 기반 네트워크의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (102) 로부터 목적지 디바이스 (116) 로의 통신을 가능하게 하는 데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0030] 일부 예에서, 소스 디바이스 (102) 는 출력 인터페이스 (108) 로부터 저장 디바이스 (116) 로 인코딩된 데이터를 출력할 수도 있다. 유사하게, 목적지 디바이스 (116) 는 입력 인터페이스 (122) 를 통해 저장 디바이스 (116) 로부터 인코딩된 데이터에 액세스할 수도 있다. 저장 디바이스 (116) 는 하드 드라이브, 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 휘발성 또는 비휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 로컬 액세스 데이터 저장 매체들 중 임의의 것을 포함할 수도 있다.

[0031] 일부 예에서, 소스 디바이스 (102) 는 소스 디바이스 (102) 에 의해 생성된 인코딩된 비디오를 저장할 수도 있는 파일 서버 (114) 또는 다른 중간 저장 디바이스에 인코딩된 비디오 데이터를 출력할 수도 있다. 목적지 디바이스 (116) 는 스트리밍 또는 다운로드를 통해 파일 서버 (114) 로부터의 저장된 비디오 데이터에 액세스할 수도 있다. 파일 서버 (114) 는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (116) 에 송신할 수 있는 임의의 유형의 서버 디바이스일 수도 있다. 파일 서버 (114) 는 (예를 들어, 웹사이트를 위한) 웹 서버, 파일 전송 프로토콜 (FTP) 서버, 콘텐츠 전달 네트워크 디바이스, 또는 NAS (network attached storage) 디바이스를 나타낼 수도 있다. 목적지 디바이스 (116) 는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해 파일 서버 (114) 로부터 인코딩된 비디오 데이터에 액세스할 수도 있다. 이것은 파일 서버 (114) 상에 저장된 인코딩된 비디오 데이터에 액세스하기에 적합한, 무선 채널 (예를 들어, Wi-Fi 접속), 유선 접속 (예를 들어, DSL, 케이블 모뎀 등), 또는 양자의 조합을 포함할 수도 있다. 파일 서버 (114) 및 입력 인터페이스 (122) 는 스트리밍 송신 프로토콜, 다운로드 송신 프로토콜, 또는 이들의 조합에 따라 동작하도록 구성될 수도 있다.

[0032] 출력 인터페이스 (108) 및 입력 인터페이스 (122) 는 무선 송신기/수신기, 모뎀들, 유선 네트워킹 컴포넌트들 (예를 들어, 이더넷 카드들), 다양한 IEEE 802.11 표준들 중 임의의 것에 따라 동작하는 무선 통신 컴포넌트들, 또는 다른 물리적 컴포넌트들을 나타낼 수도 있다. 출력 인터페이스 (108) 및 입력 인터페이스 (122) 가 무선 컴포넌트를 포함하는 예들에 있어서, 출력 인터페이스 (108) 및 입력 인터페이스 (122) 는 4G, 4G-LTE

(Long-Term Evolution), LTE 어드밴스드, 5G 등과 같은 셀룰러 통신 표준에 따라, 인코딩된 비디오 데이터와 같은 데이터를 전송하도록 구성될 수도 있다. 출력 인터페이스 (108) 가 무선 송신기를 포함하는 일부 예들에 있어서, 출력 인터페이스 (108) 및 입력 인터페이스 (122) 는 IEEE 802.11 사양, IEEE 802.15 사양 (예를 들어, ZigBee™), Bluetooth™ 표준 등과 같은 다른 무선 표준들에 따라, 인코딩된 비디오 데이터와 같은 데이터를 전송하도록 구성될 수도 있다. 일부 예들에서, 소스 디바이스 (102) 및/또는 목적지 디바이스 (116) 는 각각의 시스템-온-칩 (SoC) 디바이스들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (102) 는 비디오 인코더 (200) 및/또는 출력 인터페이스 (108) 에 기인한 기능성을 수행하기 위한 SoC 디바이스를 포함할 수도 있고, 목적지 디바이스 (116) 는 비디오 디코더 (300) 및/또는 입력 인터페이스 (122) 에 기인한 기능성을 수행하기 위한 SoC 디바이스를 포함할 수도 있다.

[0033] 본 개시의 기술은 OTA (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 인터넷 스트리밍 비디오 송신들, 예컨대 DASH (dynamic adaptive streaming over HTTP), 데이터 저장 매체 상으로 인코딩되는 디지털 비디오, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은 다양한 멀티미디어 애플리케이션들 중 임의의 것을 지원하여 비디오 코딩에 적용될 수도 있다.

[0034] 목적지 디바이스 (116) 의 입력 인터페이스 (122) 는 컴퓨터 판독 가능 매체 (110) (예를 들어, 저장 디바이스 (112), 파일 서버 (114) 등) 로부터 인코딩된 비디오 비트스트림을 수신한다. 컴퓨터 판독가능 매체 (110) 로부터의 인코딩된 비디오 비트스트림은 비디오 블록들 또는 다른 코딩된 유닛들 (예를 들어, 슬라이스들, 화상들, 화상들의 그룹들, 시퀀스들 등) 의 프로세싱 및/또는 특성들을 기술하는 값들을 갖는 구문 요소와 같은, 비디오 디코더 (300) 에 의해 또한 사용되는 비디오 인코더 (200) 에 의해 정의된 시그널링 정보를 포함할 수도 있다. 디스플레이 디바이스 (118) 는 디코딩된 비디오 데이터의 디코딩된 화상들을 사용자에게 디스플레이한다. 디스플레이 디바이스 (118) 는 음극선관 (CRT), 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 어느 것을 나타낼 수도 있다.

[0035] 도 1 에 도시되지는 않았지만, 일부 예에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 각각 오디오 인코더 및/또는 오디오 디코더와 통합될 수도 있고, 공통 데이터 스트림에서 오디오 및 비디오 양자 모두를 포함하는 멀티플렉싱된 스트림들을 핸들링하기 위해, 적절한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및/또는 소프트웨어를 포함할 수도 있다. 적용가능하면, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜 또는 다른 프로토콜들 이를테면 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.

[0036] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 각각 다양한 적합한 인코더 및/또는 디코더 회로, 예컨대 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적 회로들 (ASIC들), 필드 프로그래밍 가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들 중 어느 것으로서 구현될 수도 있다. 기술이 부분적으로 소프트웨어에서 구현되는 경우, 디바이스는 적합한, 비일시적 컴퓨터 판독 가능 매체에 소프트웨어를 위한 명령들을 저장하고, 본 개시의 기술을 수행하기 위해 하나 이상의 프로세서들을 사용하여 하드웨어에서 그 명령들을 실행할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 의 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 쪽도 각각의 디바이스에서 결합된 인코더/디코더 (CODEC) 의 부분으로서 통합될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 를 포함하는 디바이스는 집적 회로, 마이크로프로세서, 및/또는 무선 통신 디바이스, 예컨대 셀룰러 전화기를 포함할 수도 있다.

[0037] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 고 효율 비디오 코딩 (HEVC) 으로 또한 지칭되는 ITU-T H.265 와 같은 비디오 코딩 표준 또는 그에 대한 확장들, 예컨대 멀티-뷰 및/또는 스케일러블 비디오 코딩 확장들에 따라 동작할 수도 있다. 대안적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 JEM (Joint Exploration Test Model) 또는 VVC (Versatile Video Coding) 로도 또한 지칭되는 ITU-T H.266 과 같은 다른 독점 또는 산업 표준들에 따라 동작할 수도 있다. VVC 표준의 최근 드래프트는 Bross 등의, "Versatile Video Coding (Draft 7)," Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 16th Meeting: Geneva, CH, 1-11 October 2019, JVET-P2001-v9 (hereinafter "VVC Draft 7") 에 기재되어 있다. 하지만, 본 개시의 기법들은, 어느 특정 코딩 표준에 한정되지 않는다.

[0038] 일반적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 화상들의 블록 기반 코딩을 수행할 수도 있다. 용어 "블록" 은 일반적으로 프로세싱될 (예컨대, 인코딩될, 디코딩될, 또는 다르게는 인코딩 및/또는 디코딩 프로세스에서 사용될) 데이터를 포함하는 구조를 지칭한다. 예를 들어, 블록은 루미넌스 및/또는 크로미넌

스 데이터의 샘플들의 2차원 매트릭스를 포함할 수도 있다. 일반적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 YUV (예를 들어, Y, Cb, Cr) 포맷으로 표현된 비디오 데이터를 코딩할 수도 있다. 즉, 화상의 샘플들을 위한 적색, 녹색, 및 청색 (RGB) 데이터를 코딩하기보다는, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 및 크로미넌스 성분들을 코딩할 수도 있고, 여기서 크로미넌스 성분은 적색 색조 및 청색 색조 크로미넌스 성분 양자 모두를 포함할 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 인코딩 이전에 수신된 RGB 포맷화된 데이터를 YUV 표현으로 변환하고, 비디오 디코더 (300) 는 YUV 표현을 RGB 포맷으로 변환한다. 대안적으로, 전-프로세싱 유닛 및 후-프로세싱 유닛들 (도시되지 않음) 이 이들 변환들을 수행할 수도 있다.

[0039] YUV 4:4:4는 3개의 성분 (예를 들어, Y, Cb, Cr) 각각이 동일한 샘플링 레이트를 갖고 크로마 서브샘플링이 없는 YUV 표현을 나타낼 수도 있다. 예를 들어, YUV 4:4:4 의 루마 (예를 들어, Y) 성분은 제 1 해상도를 가질 수도 있다. 이 예에서, YUV 4:4:4 의 제 1 크로마 성분 (예를 들어, Cb) 성분은 루마 성분의 제 1 해상도를 가질 수도 있고 YUV 4:4:4 의 제 2 크로마 성분 (예를 들어, Cr) 성분은 루마 성분의 제 1 해상도를 가질 수도 있다. 대조적으로, YUV 4:2:0 는 각각의 크로마 성분 (예를 들어, Cb, Cr) 이 수직 및 수평 방향 모두에서 루마 성분의 절반의 샘플링 레이트를 갖고 따라서 YUV 4:2:0 이 크로마 서브샘플링을 가질 수도 있는 YUV 표현을 나타낼 수도 있다. 예를 들어, YUV 4:4:4 의 루마 (예를 들어, Y) 성분은 제 1 해상도를 가질 수도 있다. 이 예에서, YUV 4:4:4의 제 1 크로마 성분 (예를 들어, Cb) 성분은 루마 성분의 제 1 해상도의 절반인 제 2 해상도 (예를 들어, 수직 및 수평) 를 가질 수도 있고 YUV 4:4:4의 제 2 크로마 성분 (예를 들어, Cr) 성분은 루마 성분의 제 1 해상도의 절반이고 제 1 크로마 성분의 제 2 해상도와 동일한 제 2 해상도 (예를 들어, 수직 및 수평) 를 가질 수도 있다. 본 개시는 크로마 포맷의 예로서 YUV 4:4:4 및 YUV 4:2:0을 언급하지만, 다른 예는 다른 포맷 (예를 들어, YUV 4:2:2 등) 을 사용할 수도 있다.

[0040] 본 개시는 일반적으로 화상의 데이터를 인코딩 또는 디코딩하는 프로세스를 포함하는 화상들의 코딩 (예를 들어, 인코딩 및 디코딩) 을 언급할 수도 있다. 유사하게, 본 개시는, 예를 들어, 예측 및/또는 잔차 코딩과 같은, 블록들을 위한 데이터를 인코딩 또는 디코딩하는 프로세스를 포함하는 화상의 블록들의 코딩을 언급할 수도 있다. 인코딩된 비디오 비트스트림은 일반적으로 코딩 결정들 (예를 들어, 코딩 모드들) 및 화상들의 블록들로의 파티셔닝을 나타내는 구문 요소를 위한 일련의 값들을 포함한다. 따라서, 화상 또는 블록을 코딩하는 것에 대한 언급들은 일반적으로 화상 또는 블록을 형성하는 구문 요소를 위한 코딩 값들로서 이해되어야 한다.

[0041] HEVC 는 코딩 유닛들 (CU들), 예측 유닛들 (PU들), 및 변환 유닛들 (TU들) 을 포함한 다양한 블록들을 정의한다. HEVC 에 따르면, (비디오 인코더 (200) 와 같은) 비디오 코더는 쿼드트리 구조에 따라 코딩 트리 유닛 (CTU) 을 CU들로 파티셔닝한다. 즉, 비디오 코더는 CTU들 및 CU들을 4개의 동일한 비-중첩 정사각형들로 파티셔닝하고, 쿼드트리의 각각의 노드는 0개 또는 4개의 자식 노드들 중 어느 하나를 갖는다. 자식 노드들이 없는 노드들은 "리프 노드들" 로서 지칭될 수도 있고, 그러한 리프 노드들의 CU들은 하나 이상의 PU들 및/또는 하나 이상의 TU들을 포함할 수도 있다. 비디오 코더는 PU들 및 TU들을 추가로 파티셔닝할 수도 있다. 예를 들어, HEVC 에서, 잔차 쿼드트리 (RQT) 는 TU들의 파티셔닝을 나타낸다. HEVC 에서, PU들은 인터 예측 데이터를 나타내는 한편, TU들은 잔차 데이터를 나타낸다. 인트라 예측되는 CU들은 인트라 모드 표시와 같은 인트라 예측 정보를 포함한다.

[0042] 다른 예로서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 JEM 또는 VVC 에 따라 동작하도록 구성될 수도 있다. JEM 또는 VVC 에 따르면, 비디오 코더 (예컨대 비디오 인코더 (200)) 는 화상을 복수의 코딩 트리 유닛들 (CTU들) 로 파티셔닝한다. 비디오 인코더 (200) 는 쿼드트리 바이너리 트리 (quadtree-binary tree; QTBT) 구조 또는 멀티 타입 트리 (Multi-Type Tree; MTT) 구조와 같은 트리 구조에 따라 CTU 를 파티셔닝할 수도 있다. QTBT 구조는 HEVC 의 CU들, PU들, 및 TU들 간의 분리와 같은 다수의 파티션 유형들의 개념들을 제거한다. QTBT 구조는 2 개의 레벨들: 쿼드트리 파티셔닝에 따라 파티셔닝된 제 1 레벨, 및 바이너리 트리 파티셔닝에 따라 파티셔닝된 제 2 레벨을 포함한다. QTBT 구조의 루트 노드는 CTU 에 대응한다. 바이너리 트리들의 리프 노드들은 코딩 유닛들 (CU들) 에 대응한다.

[0043] MTT 파티셔닝 구조에서, 블록들은 쿼드트리 (QT) 파티션, 바이너리 트리 (BT) 파티션, 및 하나 이상의 유형의 트리플 트리 (TT) 파티션들을 사용하여 파티셔닝될 수도 있다. 트리플 트리 파티션은 블록이 3 개의 서브-블록들로 분할되는 파티션이다. 일부 예에서, 트리플 트리 파티션은 중심을 통해 원래의 블록을 나누지 않고 블록을 3 개의 서브-블록들로 나눈다. MTT 에서의 파티셔닝 유형 (예를 들어, QT, BT, 및 TT) 은 대칭적

또는 비대칭적일 수도 있다.

[0044] 일부 예에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 및 크로미넌스 성분들의 각각을 나타내기 위해 단일 QTBT 또는 MTT 구조를 사용할 수도 있는 한편, 다른 예에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 2 개 이상의 QTBT 또는 MTT 구조들, 이를 테면 루미넌스 성분을 위한 하나의 QTBT/MTT 구조 및 양자의 크로미넌스 성분을 위한 다른 QTBT/MTT 구조 (또는 각각의 크로미넌스 성분을 위한 2 개의 QTBT/MTT 구조들) 를 사용할 수도 있다.

[0045] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 HEVC 당 쿼드트리 파티셔닝, QTBT 파티셔닝, MTT 파티셔닝, 또는 다른 파티셔닝 구조들을 사용하도록 구성될 수도 있다. 설명의 목적을 위해, 본 개시의 기술의 설명은 QTBT 파티셔닝에 대하여 제시된다. 그러나, 본 개시의 기술은 또한, 쿼드트리 파티셔닝, 또는 다른 유형의 파티셔닝도 역시 사용하도록 구성된 비디오 코더들에 적용될 수도 있음을 이해해야 한다.

[0046] 본 개시는 수직 및 수평 차원들의 측면에서 (CU 또는 다른 비디오 블록과 같은) 블록의 샘플 차원을 지칭하기 위해 "NxN"및 "N 바이 (by) N", 예를 들어 16x16 샘플 또는 16 바이 16 샘플을 상호 교환적으로 사용할 수도 있다. 일반적으로, 16x16 CU 는, 수직 방향에서 16개 샘플들 (y = 16) 그리고 수평 방향에서 16개 샘플들 (x = 16) 을 가질 것이다. 마찬가지로, NxN CU 는 일반적으로 수직 방향에서 N개 샘플들 및 수평 방향에서 N개 샘플들을 갖고, 여기서 N 은 음이 아닌 정수 값을 나타낸다. CU 에서의 샘플들은 행들 (rows) 및 열들 (columns) 로 배열될 수도 있다. 더욱이, CU들은 수직 방향에서와 동일한 수의 샘플들을 수평 방향에서 반드시 가질 필요는 없다. 예를 들면, CU들은 N×M 샘플들을 포함할 수도 있고, 여기서 M 은 N 과 반드시 동일한 것은 아니다.

[0047] 비디오 인코더 (200) 는 예측 및/또는 잔차 정보를 나타내는 CU들에 대한 비디오 데이터, 및 다른 정보를 인코딩한다. 예측 정보는 CU 를 위한 예측 블록을 형성하기 위하여 CU 가 어떻게 예측될지를 표시한다. 잔차 정보는 일반적으로 인코딩 이전의 CU 의 샘플들과 예측 블록 사이의 샘플 바이 샘플 (sample-by-sample) 차이들을 나타낸다.

[0048] CU 를 예측하기 위해, 비디오 인코더 (200) 는 일반적으로 인터 예측 또는 인트라 예측을 통해 CU 를 위한 예측 블록을 형성할 수도 있다. 인터 예측은 일반적으로 이전에 코딩된 화상의 데이터로부터 CU 를 예측하는 것을 지칭하는 반면, 인트라 예측은 일반적으로 동일한 화상의 이전에 코딩된 데이터로부터 CU 를 예측하는 것을 지칭한다. 인터 예측을 수행하기 위해, 비디오 인코더 (200) 는 하나 이상의 모션 벡터들을 사용하여 예측 블록을 생성할 수도 있다. 비디오 인코더 (200) 는 일반적으로, 예를 들어, CU 와 참조 블록 사이의 차이들의 측면에서, CU 에 근접하게 매칭하는 참조 블록을 식별하기 위해 모션 탐색을 수행할 수도 있다. 비디오 인코더 (200) 는 절대차의 합 (sum of absolute difference; SAD), 제곱차의 합 (sum of squared differences; SSD), 평균 절대차 (mean absolute difference; MAD), 평균 제곱차 (mean squared differences; MSD), 또는 참조 블록이 현재 CU 에 근접하게 매칭하는지 여부를 결정하기 위한 다른 그러한 차이 계산들을 사용하여 차이 메트릭을 계산할 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 단방향 예측 또는 양방향 예측을 사용하여 현재 CU 를 예측할 수도 있다.

[0049] JEM 및 VVC 의 일부 예들은 또한, 인터 예측 모드로 고려될 수도 있는 아핀 모션 보상 모드 (affine motion compensation mode) 를 제공한다. 아핀 모션 보상 모드에서, 비디오 인코더 (200) 는 줌 인 또는 아웃, 회전, 원근 모션 (perspective motion), 또는 다른 불규칙한 모션 타입들과 같은 비-병진 모션을 나타내는 2 개 이상의 모션 벡터들을 결정할 수도 있다.

[0050] 인트라 예측을 수행하기 위해, 비디오 인코더 (200) 는 예측 블록을 생성하기 위해 인트라 예측 모드를 선택할 수도 있다. JEM 및 VVC 의 일부 예들은 다양한 방향 모드들 뿐만 아니라 평면 모드 및 DC 모드를 포함하여 67 개의 인트라 예측 모드들을 제공한다. 일반적으로, 비디오 인코더 (200) 는 현재 블록의 샘플들을 예측할 현재 블록 (예를 들어, CU 의 블록) 에 대한 이웃하는 샘플들을 기술하는 인트라 예측 모드를 선택한다. 그러한 샘플들은 일반적으로, 비디오 인코더 (200) 가 래스터 스캔 순서로 (왼쪽에서 오른쪽으로, 상단에서 하단으로) CTU들 및 CU들을 코딩하는 것을 가정하여, 현재 블록과 동일한 화상에서 현재 블록의 상측, 좌상측, 또는 좌측에 있을 수도 있다.

[0051] 비디오 인코더 (200) 는 현재 블록을 위한 예측 모드를 나타내는 데이터를 인코딩한다. 예를 들어, 인터 예측 모드의 경우, 비디오 인코더 (200) 는 다양한 이용가능한 인터 예측 모드들 중 어느 것이 사용되는지를 나타내는 데이터 뿐만 아니라, 대응하는 모드를 위한 모션 정보를 인코딩할 수도 있다. 단방향 또는 양방향 인

터 예측을 위해, 예를 들어, 비디오 인코더 (200) 는 어드밴스드 모션 벡터 예측 (AMVP) 또는 병합 모드를 사용하여 모션 벡터들을 인코딩할 수도 있다. 비디오 인코더 (200) 는 유사한 모드들을 사용하여 아핀 모션 보상 모드에 대한 모션 벡터들을 인코딩할 수도 있다.

[0052] 블록의 인트라 예측 또는 인터 예측과 같은 예측에 후속하여, 비디오 인코더 (200) 는 블록에 대한 잔차 데이터를 계산할 수도 있다. 잔차 블록과 같은 잔차 데이터는 대응하는 예측 모드를 사용하여 형성되는, 블록과 블록을 위한 예측 블록 사이의 샘플 바이 샘플 차이들을 나타낸다. 비디오 인코더 (200) 는 샘플 도메인 대신에 변환 도메인에서 변환된 데이터를 생성하기 위해, 잔차 블록에 하나 이상의 변환들을 적용할 수도 있다. 예를 들어, 비디오 인코더 (200) 는 이산 코사인 변환 (DCT), 정수 변환, 웨이브릿 변환, 또는 개념적으로 유사한 변환을 잔차 비디오 데이터에 적용할 수도 있다. 부가적으로, 비디오 인코더 (200) 는 모드 의존적 비-분리형 이차 변환 (MDNSST), 신호 의존적 변환, Karhunen-Loeve 변환 (KLT) 등과 같은 제 1 변환에 후속하여 이차 변환을 적용할 수도 있다. 비디오 인코더 (200) 는 하나 이상의 변환들의 적용에 후속하여 변환 계수들을 생성한다.

[0053] 위에 언급된 바와 같이, 변환 계수들을 생성하기 위한 임의의 변환들에 후속하여, 비디오 인코더 (200) 는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 일반적으로, 변환 계수들이 그 계수들을 나타내는 데 사용되는 데이터의 양을 가능하게는 감소시키기 위해 양자화되어, 추가 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스를 수행함으로써, 비디오 인코더 (200) 는 계수들의 일부 또는 전부와 연관된 비트 심도를 감소시킬 수도 있다. 예를 들어, 비디오 인코더 (200) 는 양자화 동안 n-비트 값을 m-비트 값으로 라운딩 다운할 수도 있고, 여기서 n 은 m 보다 크다. 일부 예들에서, 양자화를 수행하기 위해, 비디오 인코더 (200) 는 양자화될 값의 비트단위 우측-시프트를 수행할 수도 있다.

[0054] 양자화에 이어, 비디오 인코더 (200) 는 변환 계수들을 스캔하여, 양자화된 변환 계수들을 포함한 2 차원 행렬로부터 1 차원 벡터를 생성할 수도 있다. 스캔은 벡터의 전방에 더 높은 에너지 (및 따라서 더 낮은 주파수) 계수들을 배치하고 벡터의 후방에 더 낮은 에너지 (및 따라서 더 높은 주파수) 변환 계수들을 배치하도록 설계될 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 사전 정의된 스캔 순서를 활용하여 양자화된 변환 계수들을 스캔함으로써 직렬화된 벡터를 생성하고, 그 후 벡터의 양자화된 변환 계수들을 엔트로피 인코딩할 수도 있다. 다른 예들에서, 비디오 인코더 (200) 는 적응적 스캔을 수행할 수도 있다. 1차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캔한 후, 비디오 인코더 (200) 는, 예를 들어, 컨텍스트-적응적 이진 산술 코딩 (CABAC) 에 따라, 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (200) 는 또한, 비디오 데이터를 디코딩하는 데 있어서 비디오 디코더 (300) 에 의한 사용을 위해 인코딩된 비디오 데이터와 연관된 메타데이터를 기술하는 구문 요소를 위한 값들을 엔트로피 인코딩할 수도 있다.

[0055] CABAC 을 수행하기 위해, 비디오 인코더 (200) 는 송신될 심볼에 컨텍스트 모델 내의 컨텍스트를 할당할 수도 있다. 컨텍스트는 예를 들어, 심볼의 이웃하는 값들이 제로 값인지 여부와 관련될 수도 있다. 확률 결정은 심볼에 할당된 컨텍스트에 기초할 수도 있다.

[0056] 비디오 인코더 (200) 는 구문 데이터, 예컨대 블록 기반 구문 데이터, 화상 기반 구문 데이터, 및 시퀀스 기반 구문 데이터를, 비디오 디코더 (300) 에, 예를 들어, 화상 헤더, 블록 헤더, 슬라이스 헤더에서, 또는 다른 구문 데이터, 예컨대 시퀀스 파라미터 세트 (SPS), 화상 파라미터 세트 (PPS), 또는 비디오 파라미터 세트 (VPS) 에서 추가로 생성할 수도 있다. 비디오 디코더 (300) 는 마찬가지로 대응하는 비디오 데이터를 디코딩하는 방법을 결정하기 위해 그러한 구문 데이터를 디코딩할 수도 있다.

[0057] 이러한 방식으로, 비디오 인코더 (200) 는 인코딩된 비디오 데이터, 예컨대, 화상의 블록들 (예컨대, CU들) 로의 파티셔닝을 기술하는 구문 요소 및 블록을 위한 예측 및/또는 잔차 정보를 포함하는 비트스트림을 생성할 수도 있다. 궁극적으로, 비디오 디코더 (300) 는 비트스트림을 수신하고 인코딩된 비디오 데이터를 디코딩할 수도 있다.

[0058] 일반적으로, 비디오 디코더 (300) 는 비트스트림의 인코딩된 비디오 데이터를 디코딩하기 위해 비디오 인코더 (200) 에 의해 수행되는 것과 상반되는 프로세스를 수행한다. 예를 들어, 비디오 디코더 (300) 는 비디오 인코더 (200) 의 CABAC 인코딩 프로세스와 실질적으로 유사하지만, 상반되는 방식으로 CABAC 을 사용하여 비트스트림의 구문 요소를 위한 값들을 디코딩할 수도 있다. 구문 요소는 화상의 CTU들로의 파티셔닝 정보, 및 QTBT 구조와 같은 대응하는 파티션 구조에 따른 각각의 CTU 의 파티셔닝을 정의하여, CTU 의 CU들을 정의할 수도 있다. 구문 요소는 비디오 데이터의 블록들 (예를 들어, CU들) 에 대한 예측 및 잔차 정보를 추가로 정의할 수도 있다.

[0059] 잔차 정보는 예를 들어 양자화된 변환 계수들에 의해 표현될 수도 있다. 비디오 디코더 (300) 는 블록을 위한 잔차 블록을 재생성하기 위해 블록의 양자화된 변환 계수들을 역 양자화 및 역 변환할 수도 있다. 비디오 디코더 (300) 는 시그널링된 예측 모드 (인트라 또는 인터 예측) 및 관련된 예측 정보 (예를 들어, 인터 예측을 위한 모션 정보) 를 사용하여 블록을 위한 예측 블록을 형성한다. 비디오 디코더 (300) 는 그 후 예측 블록과 잔차 블록을 (샘플 바이 샘플 기반으로) 결합하여 원래 블록을 재생성할 수도 있다. 즉, 예를 들어, 비디오 디코더 (300) 는 비디오 데이터의 블록을 위한 필터링되지 않은 재구성된 블록을 디코딩하기 위해 예측된 블록과 잔차 블록을 조합할 수도 있다. 일부 예에서, 비디오 디코더 (300) 는 DPB (314) 에서 필터링되지 않은 재구성된 블록을 저장할 수도 있다. 비디오 디코더 (300) 는 블록의 경계들을 따라 시각적 아티팩트들을 감소시키기 위해 디블로킹 프로세스를 수행하는 것과 같은 추가적인 프로세싱을 수행할 수도 있다. 즉, 예를 들어, 비디오 디코더 (300) 는 비디오 데이터의 블록을 위해 필터링된 재구성된 블록을 생성할 수도 있고, 여기서 필터링된 재구성된 블록을 생성하는 것은 필터링되지 않은 재구성된 블록에 대해 하나 이상의 필터 동작을 수행하는 것을 포함한다. 이 예에서, 비디오 디코더 (300) 는 DPB (314) 에서 필터링된 재구성된 블록을 저장할 수도 있다.

[0060] 본 개시의 기법들에 따르면, 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하도록 구성될 수도 있다. 이 예에서, 비디오 인코더 (200) 또는 비디오 디코더 (300) 는 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하도록 구성될 수도 있다. 비디오 인코더 (200) 또는 비디오 디코더 (300) 는 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하도록 구성될 수도 있다. YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성하기 위한 단일 가중치 세트를 사용하는 것은 비디오 데이터 디코딩의 복잡성을 감소시킬 수도 있고, 이는 비디오 인코더 (200) 및 비디오 디코더 (300) 가 비디오 데이터를 디코딩하는 시간의 양을 감소시키거나 및/또는 비디오 인코더 (200) 및 비디오 디코더 (300) 의 전력 소비를 감소시킬 수 있다.

[0061] 픽셀 블렌딩을 적용하기 위해, 비디오 인코더 (200) 또는 비디오 디코더 (300) 는, 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하도록 구성될 수도 있다.

[0062] 일부 예에서, 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 화상의 영역을 위한 필터링되지 않은 참조 샘플들을 획득하도록 구성될 수도 있다. 이 예에서, 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플들을 위한 필터링되지 않은 참조 샘플들의 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다. YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하는 것은, 비디오 데이터를 디코딩하는 복잡성을 감소시킬 수도 있고, 이는 비디오 인코더 (200) 및 비디오 디코더 (300) 가 비디오 데이터를 디코딩하는 시간의 양을 감소시키거나 및/또는 비디오 인코더 (200) 및 비디오 디코더 (300) 의 전력 소비를 감소시킬 수 있다.

[0063] 본 개시는 일반적으로 구문 요소들과 같은, 소정의 정보를 "시그널링" 하는 것을 언급할 수도 있다. 용어 “시그널링” 은 일반적으로, 인코딩된 비디오 데이터를 디코딩하는 데 사용되는 구문 요소 및/또는 다른 데이터의 값들의 통신을 지칭할 수도 있다. 즉, 비디오 인코더 (200) 는 비트스트림에서 구문 요소를 위한 값들을 시그널링할 수도 있다. 일반적으로, 시그널링은 비트스트림에서 값을 생성하는 것을 지칭한다. 상기 언급된 바와 같이, 소스 디바이스 (102) 는 목적지 디바이스 (116) 에 의한 추후 취출을 위해 저장 디바이스 (112) 에 구문 요소를 저장할 때 발생할 수도 있는 바와 같이, 비실시간으로 또는 실질적으로 실시간으로 비트스트림을 목적지 디바이스 (116) 로 전송할 수도 있다.

[0064] 도 2a 및 도 2b 는 예시적인 쿼드트리 이진 트리 (QTBT) 구조 (130), 및 대응하는 코딩 트리 유닛 (CTU) (132) 을 예시하는 개념도이다. 실선들은 쿼드트리 분할을 나타내고, 점선들은 이진 트리 분할을 나타낸다. 이진 트리의 각각의 분할된 (즉, 비-리프) 노드에서, 어느 분할 타입 (즉, 수평 또는 수직) 이 사용되는지를 나타내기 위해 하나의 플래그가 시그널링되며, 여기서 0 은 수평 분할을 표시하고 1 은 수직 분할을 표시한다.

쿼드트리 분할의 경우, 분할 타입을 표시할 필요가 없는데, 쿼드트리 노드들은 블록을 동일한 크기를 갖는 4 개의 서브-블록들로 수평으로 그리고 수직으로 분할하기 때문이다. 이에 따라, QTBT 구조 (130) 의 영역 트리 레벨 (즉, 실선들) 을 위한 구문 요소 (예컨대 분할 정보) 및 QTBT 구조 (130) 의 예측 트리 레벨 (즉, 점선들) 을 위한 구문 요소 (예컨대 분할 정보) 를, 비디오 인코더 (200) 가 인코딩할 수도 있고, 비디오 디코더 (300) 가 디코딩할 수도 있다. QTBT 구조 (130) 의 종단 리프 노드들에 의해 표현된 CU들에 대한, 예측 및 변환 데이터와 같은, 비디오 데이터를, 비디오 인코더 (200) 가 인코딩할 수도 있고, 비디오 디코더 (300) 가 디코딩할 수도 있다.

[0065] 일반적으로, 도 2b 의 CTU (132) 는 제 1 및 제 2 레벨들에서 QTBT 구조 (130) 의 노드들에 대응하는 블록들의 크기를 정의하는 파라미터들과 연관될 수도 있다. 이들 파라미터들은 CTU 크기 (샘플들에서 CTU (132) 의 크기를 나타냄), 최소 쿼드트리 크기 (MinQTSize, 최소 허용된 쿼드트리 리프 노드 크기를 나타냄), 최대 바이너리 트리 크기 (MaxBTSize, 최대 허용된 바이너리 트리 루트 노드 크기를 나타냄), 최대 바이너리 트리 심도 (MaxBTDepth, 최대 허용된 바이너리 트리 심도를 나타냄), 및 최소 바이너리 트리 크기 (MinBTSize, 최소 허용된 바이너리 트리 리프 노드 크기를 나타냄) 를 포함할 수도 있다.

[0066] CTU 에 대응하는 QTBT 구조의 루트 노드는 QTBT 구조의 제 1 레벨에서 4 개의 자식 노드들을 가질 수도 있고, 이들의 각각은 쿼드트리 파티셔닝에 따라 파티셔닝될 수도 있다. 즉, 제 1 레벨의 노드들은 리프 노드들 (자식 노드들이 없음) 이거나 또는 4 개의 자식 노드들을 갖는다. QTBT 구조 (130) 의 예는 그러한 노드들을 브랜치들에 대한 실선들을 갖는 자식 노드들 및 부모 노드를 포함하는 것으로서 나타낸다. 제 1 레벨의 노드들이 최대 허용된 바이너리 트리 루트 노드 크기 (MaxBTSize) 보다 더 크지 않으면, 그들은 개개의 바이너리 트리들에 의해 추가로 파티셔닝될 수 있다. 하나의 노드의 바이너리 트리 분할은 분할로부터 발생하는 노드들이 최소 허용된 바이너리 트리 리프 노드 크기 (MinBTSize) 또는 최대 허용된 바이너리 트리 깊이 (MaxBTDepth) 에 도달할 때까지 반복될 수 있다. QTBT 구조 (130) 의 예는 그러한 노드들을 브랜치들을 위한 점선을 갖는 것으로서 나타낸다. 이진 트리 리프 노드는 어떠한 추가의 파티셔닝도 없이, 예측 (예를 들어, 화상내 또는 화상간 예측) 및 변환을 위해 사용되는 코딩 유닛 (CU) 으로 지칭된다. 위에 논의된 바와 같이, CU들은 또한, "비디오 블록들" 또는 "블록들" 로 지칭될 수도 있다.

[0067] QTBT 파티셔닝 구조의 하나의 예에서, CTU 크기는 128x128 (루마 샘플들 및 2 개의 대응하는 64x64 크로마 샘플들) 로서 설정되고, MinQTSize 는 16x16 으로서 설정되고, MaxBTSize 는 64x64 로서 설정되고, (폭 및 높이 양자 모두에 대한) MinBTSize 는 4 로서 설정되고, 그리고 MaxBTDepth 는 4 로서 설정된다. 쿼드트리 파티셔닝은 쿼드-트리 리프 노드들을 생성하기 위해 먼저 CTU 에 적용된다. 쿼드트리 리프 노드들은 16x16 (즉, MinQTSize) 으로부터 128x128 (즉, CTU 크기) 까지의 크기를 가질 수도 있다. 리프 쿼드트리 노드가 128x128 인 경우, 사이즈가 MaxBTSize (즉, 이 예에서는 64x64) 를 초과하기 때문에 그것은 바이너리 트리에 의해 추가로 스플릿팅되지 않을 것이다. 그렇지 않으면, 리프 쿼드트리 노드는 바이너리 트리에 의해 추가로 파티셔닝될 것이다. 따라서, 쿼드트리 리프 노드는 또한 바이너리 트리를 위한 루트 노드이고 바이너리 트리 심도를 0 으로서 갖는다. 바이너리 트리 심도가 MaxBTDepth (이 예에서는 4) 에 도달할 때, 추가의 분할은 허용되지 않는다. 바이너리 트리 노드가 MinBTSize (이 예에서는 4) 와 동일한 폭을 가질 때, 그것은 추가의 수평 분할이 허용되지 않음을 의미한다. 유사하게, 높이가 MinBTSize 와 동일한 바이너리 트리 노드는 그 바이너리 트리 노드에 대해 추가의 수직 분할이 허용되지 않음을 암시한다. 위에 언급된 바와 같이, 이진 트리의 리프 노드들은 CU들로 지칭되고, 추가의 파티셔닝 없이 예측 및 변환에 따라 추가로 프로세싱된다.

[0068] 도 3 는 본 개시의 기술을 수행할 수도 있는 예시적인 비디오 인코더 (200) 를 예시하는 블록 도이다. 도 3 는 설명의 목적으로 제공되며, 본 개시에서 대체로 예시화되고 설명된 바와 같은 기술의 한정으로서 고려되지 않아야 한다. 설명의 목적으로, 본 개시는 HEVC 비디오 코딩 표준 및 개발 중인 JEM, VVC (ITU-T H.266) 에 비디오 코딩 표준과 같은 비디오 코딩 표준들의 맥락에서 비디오 인코더 (200) 를 설명한다. 그러나, 본 개시의 기술은 이들 비디오 코딩 표준들에 한정되지 않으며, 일반적으로 비디오 인코딩 및 디코딩에 적용가능하다.

[0069] 도 3 의 예에서, 비디오 인코더 (200) 는 비디오 데이터 메모리 (230), 모드 선택 유닛 (202), 잔차 생성 유닛 (204), 변환 프로세싱 유닛 (206), 양자화 유닛 (208), 역 양자화 유닛 (210), 역 변환 프로세싱 유닛 (212), 재구성 유닛 (214), 필터 유닛 (216), 디코딩된 화상 버퍼 (DPB) (218), 및 엔트로피 인코딩 유닛 (220) 을 포함한다. 비디오 데이터 메모리 (230), 모드 선택 유닛 (202), 잔차 생성 유닛 (204), 변환 프로세싱 유닛 (206), 양자화 유닛 (208), 역 양자화 유닛 (210), 역 변환 프로세싱 유닛 (212), 재구성 유닛 (214), 필터 유닛 (216), DPB (218), 및 엔트로피 인코딩 유닛 (220) 중 임의의 것 또는 전부는 하나 이상의 프로세서에서 또

는 프로세싱 회로에서 구현될 수도 있다. 예를 들어, 비디오 인코더 (200) 의 유닛은 프로세서, ASIC 또는 FPGA의 부분을 형성하는 하나 이상의 회로 또는 논리 요소로서 구현될 수도 있다. 더욱이, 비디오 인코더 (200) 는 이들 및 다른 기능들을 수행하기 위해 부가 또는 대안의 프로세서들 또는 프로세싱 회로를 포함할 수도 있다.

[0070] 비디오 데이터 메모리 (230) 는 비디오 인코더 (200) 의 컴포넌트들에 의해 인코딩될 비디오 데이터를 저장할 수도 있다. 비디오 인코더 (200) 는 예를 들어, 비디오 소스 (104) (도 1) 로부터 비디오 데이터 메모리 (230) 에 저장된 비디오 데이터를 수신할 수도 있다. DPB (218) 는 비디오 인코더 (200) 에 의한 후속 비디오 데이터의 예측에 사용하기 위해 참조 비디오 데이터를 저장하는 참조 화상 메모리로서 작용할 수도 있다. 비디오 데이터 메모리 (230) 및 DPB (218) 는 다양한 메모리 디바이스들, 예컨대 동기식 랜덤 액세스 메모리 (synchronous dynamic random access memory; SDRAM) 을 포함하는 DRAM, 자기저항 RAM (magnetoresistive RAM; MRAM), 저항 RAM (resistive RAM; RRAM), 또는 다른 타입들의 메모리 디바이스들 중 임의의 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (230) 및 DPB (218) 는 동일한 메모리 디바이스 또는 개별 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (230) 는 예시된 바와 같이 비디오 인코더 (200) 의 다른 컴포넌트들과 온-칩되거나, 또는 그 컴포넌트들에 대해 오프-칩될 수도 있다.

[0071] 본 개시에서, 비디오 데이터 메모리 (230) 에 대한 언급은 이처럼 구체적으로 설명되지 않으면 비디오 인코더 (200) 내부의 메모리, 또는 이처럼 구체적으로 설명되지 않으면 비디오 인코더 (200) 외부의 메모리에 한정되는 것으로 해석되지 않아야 한다. 오히려, 비디오 데이터 메모리 (230) 에 대한 언급은, 비디오 인코더 (200) 가 인코딩을 위해 수신하는 비디오 데이터 (예컨대, 인코딩될 현재 블록을 위한 비디오 데이터) 를 저장하는 레퍼런스 메모리로서 이해되어야 한다. 도 1 의 메모리 (106) 는 또한 비디오 인코더 (200) 의 다양한 유닛들로부터의 출력들의 일시적 저장을 제공할 수도 있다.

[0072] 도 3 의 다양한 유닛들은 비디오 인코더 (200) 에 의해 수행되는 동작들의 이해를 돕기 위해 예시된다. 그 유닛들은 고정 기능 회로들, 프로그래밍가능 회로들, 또는 이들의 조합으로서 구현될 수도 있다. 고정 기능 회로들은 특정 기능을 제공하는 회로들을 지칭하고, 수행될 수 있는 동작들에 대해 미리설정된다. 프로그램가능 회로들은 다양한 태스크들을 수행하도록 프로그램될 수 있는 회로들을 지칭하며, 수행될 동작들에서 유연한 기능성을 제공한다. 예를 들어, 프로그래밍가능 회로들은, 프로그래밍가능 회로들로 하여금, 소프트웨어 또는 펌웨어의 명령들에 의해 정의된 방식으로 동작하게 하는 소프트웨어 또는 펌웨어를 실행할 수도 있다. 고정 기능 회로들은 (예를 들어, 파라미터들을 수신하거나 또는 파라미터들을 출력하기 위해) 소프트웨어 명령들을 실행할 수도 있지만, 고정 기능 회로들이 수행하는 동작들의 타입들은 일반적으로 불변이다. 일부 예들에서, 유닛들 중 하나 이상은 별개의 회로 블록들 (고정 기능 또는 프로그래밍가능) 일 수도 있고, 일부 예들에서, 하나 이상의 유닛들은 집적 회로들일 수도 있다.

[0073] 비디오 인코더 (200) 는 프로그래밍가능 회로들로부터 형성된, 산술 로직 유닛들 (arithmetic logic unit; ALU 들), 기본 함수 유닛들 (elementary function unit; EFU들), 디지털 회로들, 아날로그 회로들, 및/또는 프로그래밍가능 코어들을 포함할 수도 있다. 비디오 인코더 (200) 의 동작이 프로그램 가능 회로에 의해 실행되는 소프트웨어를 사용하여 수행되는 예에서, 메모리 (106) (도 1) 는 비디오 인코더 (200) 가 수신하고 실행하는 소프트웨어의 오브젝트 코드를 저장할 수도 있거나, 또는 비디오 인코더 (200) 내의 다른 메모리 (미도시) 가 그러한 명령들을 저장할 수도 있다.

[0074] 비디오 데이터 메모리 (230) 는 수신된 비디오 데이터를 저장하도록 구성된다. 비디오 인코더 (200) 는 비디오 데이터 메모리 (230) 로부터 비디오 데이터의 화상을 취출하고, 비디오 데이터를 잔차 생성 유닛 (204) 및 모드 선택 유닛 (202) 에 제공할 수도 있다. 비디오 데이터 메모리 (230) 에서의 비디오 데이터는 인코딩될 원시 비디오 데이터일 수도 있다.

[0075] 모드 선택 유닛 (202) 은 모션 추정 유닛 (222), 모션 보상 유닛 (224), 및 인트라 예측 유닛 (226) 을 포함한다. 모드 선택 유닛 (202) 은 다른 예측 모드들에 따라 비디오 예측을 수행하기 위해 추가적인 기능 유닛들을 포함할 수도 있다. 예들로서, 모드 선택 유닛 (202) 은 팔레트 유닛, 인트라 블록 카피 유닛 (이는 모션 추정 유닛 (222) 및/또는 모션 보상 유닛 (224) 의 부분일 수도 있음), 아핀 유닛, 선형 모델 (LM) 유닛 등을 포함할 수도 있다.

[0076] 모드 선택 유닛 (202) 은 일반적으로 인코딩 파라미터들의 조합들 및 그러한 조합들에 대한 결과적인 레이트-왜곡 값들을 테스트하기 위해 다중의 인코딩 패스들을 조정한다. 인코딩 파라미터들은 CTU들의 CU들로의 파티셔닝, CU들을 위한 예측 모드들, CU들의 잔차 데이터를 위한 변환 타입들, CU들의 잔차 데이터를 위한 양자화

파라미터들 등을 포함할 수도 있다. 모드 선택 유닛 (202) 은 궁극적으로 다른 테스트된 조합들보다 더 나은 레이트-왜곡 값들을 갖는 인코딩 파라미터들의 조합을 선택할 수도 있다.

[0077] 비디오 인코더 (200) 는 비디오 데이터 메모리 (230) 로부터 취출된 화상을 일련의 CTU들로 파티셔닝하고, 슬라이스 내에 하나 이상의 CTU들을 캡슐화할 수도 있다. 모드 선택 유닛 (202) 은 위에 설명된 HEVC 의 쿼드트리 구조 또는 QTBT 구조와 같은, 트리 구조에 따라 화상의 CTU 를 파티셔닝할 수도 있다. 위에 설명된 바와 같이, 비디오 인코더 (200) 는 트리 구조에 따라 CTU 를 파티셔닝하는 것으로부터 하나 이상의 CU들을 형성할 수도 있다. 그러한 CU 는 일반적으로 "비디오 블록" 또는 "블록" 으로도 또한 지칭될 수도 있다.

[0078] 일부 예에서, 모드 선택 유닛 (202) 은 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하도록 구성될 수도 있다. 이 예에서, 모드 선택 유닛 (202) 은 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하도록 구성될 수도 있다. 모드 선택 유닛 (202) 은 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하도록 구성될 수도 있다. 이러한 방식으로, 모드 선택 유닛(202) 은 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷 양자 모두를 위한 가중치 세트를 사용할 수도 있으며, 이는 상이한 크로마 포맷의 처리를 통일하는 데 도움이 될 수 있다. 상이한 크로마 포맷의 처리를 통일하면 코덱 프로세싱 오버헤드를 줄이는 데 도움이 될 수 있다.

[0079] 픽셀 블렌딩을 적용하기 위해, 모드 선택 유닛 (202) 은, 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하도록 구성될 수도 있다. 가중 평균을 결정하는 예시적인 프로세스는 아래에서 더 자세히 설명된다 (예를 들어, 도 7a 내지 도 7c 참조).

[0080] 일반적으로, 일부 예에서, 모드 선택 유닛 (202) 은 또한 그의 컴포넌트들 (예를 들어, 모션 추정 유닛 (222), 모션 보상 유닛 (224), 및 인트라 예측 유닛 (226)) 을 제어하여 현재 블록 (예를 들어, 현재 CU, 또는 HEVC 에서, PU 및 TU 의 오버랩하는 부분) 을 위한 예측 블록을 생성한다. 현재 블록의 인터 예측을 위해, 모션 추정 유닛 (222) 은 하나 이상의 참조 화상들 (예를 들어, DPB (218) 에 저장된 하나 이상의 이전에 코딩된 화상들) 에서 하나 이상의 밀접하게 매칭하는 참조 블록들을 식별하기 위해 모션 탐색을 수행할 수도 있다. 특히, 모션 추정 유닛 (222) 은, 예를 들어 절대차의 합 (sum of absolute difference; SAD), 제곱차의 합 (sum of squared differences; SSD), 평균 절대차 (mean absolute difference; MAD), 평균 제곱차의 합 (mean squared differences; MSD) 등에 따라, 잠재적 참조 블록이 현재 블록에 얼마나 유사한지를 나타내는 값을 계산할 수도 있다. 모션 추정 유닛 (222) 은 일반적으로 고려되는 참조 블록과 현재 블록 사이의 샘플 바이 샘플 (sample-by-sample) 차이들을 사용하여 이러한 계산들을 수행할 수도 있다. 모션 추정 유닛 (222) 은 현재 블록과 가장 밀접하게 매칭하는 참조 블록을 표시하는, 이러한 계산들로부터 비롯되는 최저 값을 갖는 참조 블록을 식별할 수도 있다.

[0081] 모션 추정 유닛 (222) 은 현재 화상에서의 현재 블록의 위치에 대한 참조 화상들에서의 참조 블록들의 위치들을 정의하는 하나 이상의 모션 벡터 (MV) 들을 형성할 수도 있다. 모션 추정 유닛 (222) 은 그 후 모션 벡터들을 모션 보상 유닛 (224) 에 제공할 수도 있다. 예를 들어, 단방향 인터 예측에 대해, 모션 추정 유닛 (222) 은 단일 모션 벡터를 제공할 수도 있는 반면, 양방향 인터 예측에 대해, 모션 추정 유닛 (222) 은 2개의 모션 벡터들을 제공할 수도 있다. 그 다음, 모션 보상 유닛 (224) 은 모션 벡터들을 사용하여 예측 블록을 생성할 수도 있다. 예를 들어, 모션 보상 유닛 (224) 은 모션 벡터를 사용하여 참조 블록의 데이터를 취출할 수도 있다. 다른 예로서, 모션 벡터가 분수 샘플 정밀도를 갖는다면, 모션 보상 유닛 (224) 은 하나 이상의 보간 필터들에 따라 예측 블록을 위한 값들을 보간할 수도 있다. 더욱이, 양방향 인터 예측에 대해, 모션 보상 유닛 (224) 은 개별 모션 벡터들에 의해 식별된 2개의 참조 블록들을 위한 데이터를 취출하고, 예컨대, 샘플 바이 샘플 평균화 또는 가중된 평균화를 통해 취출된 데이터를 결합할 수도 있다.

[0082] 다른 예로서, 인트라 예측, 또는 인트라 예측 코딩에 대해, 인트라 예측 유닛 (226) 은 현재 블록에 이웃하는 샘플들로부터 예측 블록을 생성할 수도 있다. 예를 들어, 방향성 모드들에 대해, 인트라 예측 유닛 (226) 은 일반적으로 이웃 샘플들의 값들을 수학적으로 결합하고 현재 블록에 걸쳐 정의된 방향에서 이들 계산된 값들을 파퓰레이트 (populate) 하여 예측 블록을 생성할 수도 있다. 또 다른 예로서, DC 모드에 대해, 인트라 예측 유닛 (226) 은 현재 블록에 대한 이웃하는 샘플들의 평균을 계산하고 예측 블록을 생성하여 예측 블록의

각각의 샘플에 대해 이러한 결과적인 평균을 포함할 수도 있다.

[0083] 인트라 예측 유닛 (226) 은 화상의 영역을 위해 필터링되지 않은 참조 샘플을 획득하도록 구성될 수도 있다. 인트라 예측 유닛 (226) 은 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플들을 위한 필터링되지 않은 참조 샘플들의 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 인트라 참조 샘플 평활화가 비활성화될 때, 인트라 예측 유닛 (226) 은 크로마 샘플을 위해 필터링되지 않은 참조 샘플만을 사용할 수도 있다. 인트라 예측 유닛 (226) 은 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다. 즉, 예를 들어, YUV 4:4:4 포맷 및 YUV 4:4:4 포맷 모두의 크로마 성분에 대해, 인트라 예측 유닛 (226) 은 크로마 샘플을 위해 필터링되지 않은 참조 샘플만을 사용할 수도 있다. 이러한 방식으로, 비디오 데이터를 인코딩 및 디코딩하는 복잡성이 감소될 수 있고, 이는 인트라 예측 유닛 (226) 이 비디오 데이터를 코딩 (예를 들어, 인코딩 또는 디코딩) 하는 시간의 양을 감소시키거나 및/또는 인트라 예측 유닛 (226) 의 전력 소비를 감소시킬 수 있다.

[0084] 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화를 위해 여기에 설명된 기술은 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성하기 위해 여기서 설명된 기술과 함께 또는 별도로 사용될 수도 있다. 예를 들어, 인트라 예측 유닛 (226) 은 모드 선택 유닛 (202) 이 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하는 것과 조합하여 화상의 영역을 위해 필터링되지 않은 참조 샘플들을 획득하도록 구성될 수 있다. 일부 예에서, 인트라 예측 유닛 (226) 은 조합하여 화상의 영역을 위해 필터링되지 않은 참조 샘플들을 획득하도록 구성될 수 있고 모드 선택 유닛 (202) 은 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하는 것을 억제한다.

[0085] 모드 선택 유닛 (202) 은 예측 블록을 잔차 생성 유닛 (204) 에 제공한다. 잔차 생성 유닛 (204) 은 비디오 데이터 메모리 (230) 로부터의 현재 블록의 원시, 코딩되지 않은 버전 및 모드 선택 유닛 (202) 으로부터의 예측 블록을 수신한다. 잔차 생성 유닛 (204) 은 현재 블록과 예측 블록 사이의 샘플 바이 샘플 차이들을 계산한다. 결과적인 샘플 바이 샘플 차이들은 현재 블록을 위한 잔차 블록을 정의한다. 일부 예들에서, 잔차 생성 유닛 (204) 은 또한 잔차 차분 펄스 코드 변조 (residual differential pulse code modulation; RDPCM) 를 사용하여 잔차 블록을 생성하기 위해 잔차 블록에서의 샘플 값들 사이의 차이들을 결정할 수도 있다. 일부 예들에서, 잔차 생성 유닛 (204) 은 이진 감산 (binary subtraction) 을 수행하는 하나 이상의 감산기 회로들을 사용하여 형성될 수도 있다.

[0086] 모드 선택 유닛 (202) 이 CU들을 PU들로 파티셔닝하는 예들에서, 각각의 PU 는 루마 예측 유닛 및 대응하는 크로마 예측 유닛들과 연관될 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 다양한 크기를 갖는 PU들을 지원할 수도 있다. 위에 나타낸 바와 같이, CU 의 크기는 CU 의 루마 코딩 블록의 크기를 지칭할 수도 있고 PU 의 크기는 PU 의 루마 예측 유닛의 크기를 지칭할 수도 있다. 특정 CU 의 크기가 2Nx2N 이라고 가정하면, 비디오 인코더 (200) 는 인트라 예측을 위해 2Nx2N 또는 NxN 의 PU 크기들을 지원하고, 인터 예측을 위해 2Nx2N, 2NxN, Nx2N, NxN, 또는 유사한 것의 대칭적인 PU 크기들을 지원할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 또한, 인터 예측을 위해 2NxnU, 2NxnD, nLx2N, 및 nRx2N 의 PU 크기들에 대한 비대칭적인 파티셔닝을 지원할 수도 있다.

[0087] 모드 선택 유닛이 CU 를 PU들로 추가로 파티셔닝하지 않는 예들에서, 각각의 CU 는 루마 코딩 블록 및 대응하는 크로마 코딩 블록들과 연관될 수도 있다. 위와 같이, CU 의 크기는 CU 의 루마 코딩 블록의 크기를 지칭할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (120) 는 2Nx2N, 2NxN, 또는 Nx2N 의 CU 크기들을 지원할 수도 있다.

[0088] 몇개의 예로서, 인트라 블록 카피 모드 코딩, 아핀-모드 코딩, 및 선형 모델 (LM) 모드 코딩과 같은 다른 비디오 코딩 기술에 대해, 모드 선택 유닛 (202) 은 코딩 기술과 연관된 각각의 유닛들을 통해, 인코딩되는 현재 블록을 위한 예측 블록을 생성한다. 팔레트 모드 코딩과 같은 일부 예들에서, 모드 선택 유닛 (202) 은 예측 블록을 생성하지 않을 수도 있고, 대신 선택된 팔레트에 기초하여 블록을 재구성하는 방식을 표시하는 구문 요소들을 생성할 수도 있다. 그러한 모드들에서, 모드 선택 유닛 (202) 은 이들 구문 요소들을 인코딩될 엔트로피 인코딩 유닛 (220) 에 제공할 수도 있다.

[0089] 위에 설명된 바와 같이, 잔차 생성 유닛 (204) 은 현재 블록 및 대응하는 예측 블록을 위한 비디오 데이터를 수신한다. 잔차 생성 유닛 (204) 은 그 후 현재 블록을 위한 잔차 블록을 생성한다. 잔차 블록을 생성하

기 위해, 잔차 생성 유닛 (204) 은 예측 블록과 현재 블록 사이의 샘플 바이 샘플 차이들을 계산한다.

[0090] 변환 프로세싱 유닛 (206) 은 잔차 블록에 하나 이상의 변환들을 적용하여 변환 계수들의 블록 (본 명세서에서는 "변환 계수 블록" 으로서 지칭됨) 을 생성한다. 변환 프로세싱 유닛 (206) 은 다양한 변환들을 잔차 블록에 적용하여 변환 계수 블록을 형성할 수도 있다. 예를 들어, 변환 프로세싱 유닛 (206) 은 이산 코사인 변환 (DCT), 방향성 변환, Karhunen-Loeve 변환 (KLT), 또는 개념적으로 유사한 변환을 잔차 블록에 적용할 수도 있다. 일부 예들에 있어서, 변환 프로세싱 유닛 (206) 은 잔차 블록에 대한 다중의 변환들, 예컨대, 1 차 변환 및 2 차 변환, 예컨대 회전 변환을 수행할 수도 있다. 일부 예들에 있어서, 변환 프로세싱 유닛 (206) 은 잔차 블록에 변환들을 적용하지 않는다.

[0091] 양자화 유닛 (208) 은 양자화된 변환 계수 블록을 생성하기 위해 변환 계수 블록에서의 변환 계수들을 양자화할 수도 있다. 양자화 유닛 (208) 은 현재 블록과 연관된 양자화 파라미터 (QP) 값에 따라 변환 계수 블록의 변환 계수들을 양자화할 수도 있다. 비디오 인코더 (202) 는 (예를 들어, 모드 선택 유닛 (202) 을 통해) CU 와 연관된 QP 값을 조정함으로써 현재 블록과 연관된 계수 블록들에 적용된 양자화도를 조정할 수도 있다. 양자화는 정보의 손실을 도입할 수도 있으며, 따라서 양자화된 변환 계수들은 변환 프로세싱 유닛 (206) 에 의해 생성된 원래의 변환 계수들보다 더 낮은 정밀도를 가질 수도 있다.

[0092] 역 양자화 유닛 (210) 및 역 변환 프로세싱 유닛 (212) 은 각각 양자화된 변환 계수 블록에 역 양자화 및 역 변환들을 적용하여, 변환 계수 블록으로부터 잔차 블록을 재구성할 수도 있다. 재구성 유닛 (214) 은 모드 선택 유닛 (202) 에 의해 생성된 예측 블록 및 재구성된 잔차 블록에 기초하여 (잠재적으로 어느 정도의 왜곡을 가짐에도 불구하고) 현재 블록에 대응하는 재구성된 블록을 생성할 수도 있다. 예를 들어, 재구성 유닛 (214) 은 재구성된 잔차 블록의 샘플들을, 모드 선택 유닛 (202) 에 의해 생성된 예측 블록으로부터의 대응하는 샘플들에 가산하여 재구성된 블록을 생성할 수도 있다. 즉, 예를 들어, 재구성 유닛 (214) 은 비디오 데이터의 블록을 위한 필터링되지 않은 재구성된 블록을 디코딩하기 위해 예측된 블록과 잔차 블록을 조합할 수도 있다. 일부 예에서, 재구성 유닛 (214) 은 DPB (218) 에서 필터링되지 않은 재구성된 블록을 저장할 수도 있다.

[0093] 필터 유닛 (216) 은 복원된 블록에 대해 하나 이상의 필터 동작들을 수행할 수도 있다. 예를 들어, 필터 유닛 (216) 은 CU들의 에지들을 따라 블록키니스 아티팩트 (blockiness artifacts) 를 감소시키기 위해 디블록킹 동작들을 수행할 수도 있다. 필터 유닛 (216) 의 동작들은 일부 예들에서 스킵될 수도 있다. 즉, 예를 들어, 필터 유닛 (216) 은 비디오 데이터의 블록을 위해 필터링된 재구성된 블록을 생성할 수도 있고, 여기서 필터링된 재구성된 블록을 생성하는 것은 필터링되지 않은 재구성된 블록에 대해 하나 이상의 필터 동작을 수행하는 것을 포함한다. 이 예에서, 필터 유닛 (216) 은 DPB (218) 에서 필터링된 재구성된 블록을 저장할 수도 있다.

[0094] 비디오 인코더 (200) 는 DPB (218) 에 재구성된 블록들을 저장한다. 예를 들어, 필터 유닛 (216) 의 동작들이 필요하지 않은 예들에서, 재구성 유닛 (214) 은 재구성된 블록들을 DPB (218) 에 저장할 수도 있다. 필터 유닛 (216) 의 동작들이 필요한 예들에서, 필터 유닛 (216) 은 필터링된 재구성된 블록들을 DPB (218) 에 저장할 수도 있다. 모션 추정 유닛 (222) 및 모션 보상 유닛 (224) 은 재구성된 (및 잠재적으로 필터링된) 블록들로부터 형성된 DPB (218) 로부터 참조 화상을 취출하여, 후속 인코딩된 화상들의 블록들을 인터 예측할 수도 있다. 또한, 인트라 예측 유닛 (226) 은 현재 화상에서의 다른 블록들을 인트라 예측하기 위해 현재 화상의 DPB (218) 에서 재구성된 블록들을 사용할 수도 있다.

[0095] 일반적으로, 엔트로피 인코딩 유닛 (220) 은 비디오 인코더 (200) 의 다른 기능 컴포넌트들로부터 수신된 구문 요소들을 엔트로피 인코딩할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (220) 은 양자화 유닛 (208) 으로부터 양자화된 변환 계수 블록들을 엔트로피 인코딩할 수도 있다. 다른 예로서, 엔트로피 인코딩 유닛 (220) 은 모드 선택 유닛 (202) 으로부터 예측 구문 요소들 (예를 들어, 인터 예측을 위한 모션 정보 또는 인트라 예측을 위한 인트라 모드 정보) 을 엔트로피 인코딩할 수도 있다. 엔트로피 인코딩 유닛 (220) 은 엔트로피-인코딩된 데이터를 생성하기 위해, 비디오 데이터의 다른 예인, 구문 요소들에 대해 하나 이상의 엔트로피 인코딩 동작들을 수행할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (220) 은 컨텍스트-적응적 가변 길이 코딩 (CAVLC) 동작, CABAC 동작, V2V (variable-to-variable) 길이 코딩 동작, 구문 기반 컨텍스트-적응적 이진 산술 코딩 (SBAC) 동작, 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩 동작, 지수-골롬 인코딩 동작, 또는 다른 유형의 엔트로피 인코딩 동작을 데이터에 대해 수행할 수도 있다. 일부 예들에서, 엔트로피 인코딩 유닛 (220) 은 구문 요소들이 엔트로피 인코딩되지 않는 바이패스 모드에서 동작할 수도 있다.

[0096] 비디오 인코더 (200) 는 화상 또는 슬라이스의 블록들을 재구성하는 데 필요한 엔트로피 인코딩된 구문 요소들을 포함하는 비트스트림을 출력할 수도 있다. 특히, 엔트로피 인코딩 유닛 (220) 은 비트스트림을 출력할 수도 있다.

[0097] 위에 설명된 동작들은 블록에 대하여 설명된다. 그러한 설명은 루마 코딩 블록 및/또는 크로마 코딩 블록들을 위한 동작들인 것으로 이해되어야 한다. 위에 설명된 바와 같이, 일부 예들에서, 루마 코딩 블록 및 크로마 코딩 블록들은 CU 의 루마 및 크로마 성분들이다. 일부 예들에서, 루마 코딩 블록 및 크로마 코딩 블록들은 PU 의 루마 및 크로마 성분들이다.

[0098] 일부 예들에 있어서, 루마 코딩 블록에 대해 수행된 동작들은 크로마 코딩 블록들에 대해 반복될 필요가 없다. 하나의 예로서, 루마 코딩 블록을 위한 모션 벡터 (MV) 및 참조 화상을 식별하기 위한 동작들이 크로마 블록들을 위한 MV 및 참조 화상을 식별하기 위해 반복될 필요는 없다. 오히려, 루마 코딩 블록을 위한 MV 는 크로마 블록들을 위한 MV 를 결정하기 위해 스케일링될 수도 있고, 참조 화상은 동일할 수도 있다. 다른 예로서, 인트라 예측 프로세스는 루마 코딩 블록들 및 크로마 코딩 블록들에 대해 동일할 수도 있다.

[0099] 비디오 인코더 (200) 는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로에 구현되고 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하도록 구성된 하나 이상의 프로세서를 포함하는 비디오 데이터를 인코딩하도록 구성된 디바이스의 예를 나타낸다. 이 예에서, 비디오 인코더 (200) 는 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하도록 구성될 수도 있다. 비디오 인코더 (200) 는 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하도록 구성될 수도 있다. 이런 이유로, 비디오 인코더 (200) 는 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷 양자 모두를 위한 가중치 세트를 사용할 수도 있으며, 이는 상이한 크로마 포맷의 처리를 통일하는 데 도움이 될 수 있다. 상이한 크로마 포맷의 처리를 통일하면 코덱 프로세싱 오버헤드를 줄이는 데 도움이 될 수 있다. 픽셀 블렌딩을 적용하기 위해, 비디오 인코더 (200) 는, 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하도록 구성될 수도 있다.

[0100] 일부 예에서, 비디오 인코더 (200) 는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로에서 구현되고 화상의 영역을 위한 필터링되지 않은 참조 샘플을 획득하도록 구성된 하나 이상의 프로세서를 포함하는 비디오 데이터를 인코딩하도록 구성된 디바이스의 예를 나타낸다. 이 예에서, 비디오 인코더 (200) 는 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플들을 위한 필터링되지 않은 참조 샘플들의 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 비디오 인코더 (200) 는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다. 즉, 예를 들어, YUV 4:4:4 포맷 및 YUV 4:4:4 포맷 모두의 크로마 성분에 대해, 비디오 인코더 (200) 는 크로마 샘플을 위해 필터링되지 않은 참조 샘플만을 사용할 수도 있다. 이러한 방식으로, 비디오 데이터를 인코딩 및 디코딩하는 복잡성이 감소될 수 있고, 이는 비디오 인코더 (200) 가 비디오 데이터를 코딩 (예를 들어, 인코딩 또는 디코딩) 하는 시간의 양을 감소시키거나 및/또는 비디오 인코더 (200) 의 전력 소비를 감소시킬 수 있다. 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화를 위해 여기에 설명된 기술은 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성하기 위해 여기서 설명된 기술과 함께 또는 별도로 사용될 수도 있다.

[0101] 도 4 은 본 개시의 기법들을 수행할 수도 있는 예시적인 비디오 디코더 (300) 를 예시하는 블록 도이다. 도 4는 설명의 목적으로 제공되고 본 개시에 폭넓게 예시되고 설명된 바와 같은 기술에 대해 한정하지 않는다. 설명의 목적들을 위해, 본 개시는 비디오 디코더 (300) 가 JEM, VVC (개발중인 ITU-T H.266), 및 HEVC 의 기법들에 따라 기술되는 것을 설명한다. 그러나, 본 개시의 기술은 다른 비디오 코딩 표준들로 구성되는 비디오 코딩 디바이스들에 의해 수행될 수도 있다.

[0102] 도 4 의 예에서, 비디오 디코더 (300) 는, 코딩된 화상 버퍼 (CPB) 메모리 (320), 엔트로피 디코딩 유닛 (302), 예측 프로세싱 유닛 (304), 역 양자화 유닛 (306), 역 변환 프로세싱 유닛 (308), 재구성 유닛 (310), 필터 유닛 (312), 및 디코딩된 화상 버퍼 (DPB) (314) 를 포함한다. CPB 메모리 (320), 엔트로피 디코딩 유닛

(302), 예측 프로세싱 유닛 (304), 역 양자화 유닛 (306), 역 변환 프로세싱 유닛 (308), 재구성 유닛 (310), 필터 유닛 (312), 및 DPB (314) 의 어느 것 또는 전부는 하나 이상의 프로세서들에서 또는 프로세싱 회로에서 구현될 수도 있다. 더욱이, 비디오 디코더 (300) 는 이들 및 다른 기능들을 수행하기 위해 추가적인 또는 대안적인 프로세서들 또는 프로세싱 회로를 포함할 수도 있다.

[0103] 예측 프로세싱 유닛 (304) 은 모션 보상 유닛 (316) 및 인트라 예측 프로세싱 유닛 (318) 을 포함한다. 예측 프로세싱 유닛 (304) 은 다른 예측 모드들에 따라 예측을 수행하기 위해 부가적인 유닛들을 포함할 수도 있다. 예를 들어, 예측 프로세싱 유닛 (304) 은 팔레트 유닛, 인트라 블록 카피 유닛 (모션 보상 유닛 (318) 의 일부를 형성할 수도 있음), 아핀 유닛, 선형 모델 (LM) 유닛 등을 포함할 수도 있다. 다른 예들에서, 비디오 디코더 (300) 는 더 많거나, 더 적거나, 또는 상이한 기능 컴포넌트들을 포함할 수도 있다.

[0104] CPB 메모리 (320) 는, 비디오 디코더 (300) 의 컴포넌트들에 의해 디코딩될 인코딩된 비디오 비트스트림과 같은 비디오 데이터를 저장할 수도 있다. CPB 메모리 (320) 에 저장된 비디오 데이터는, 예를 들어, 컴퓨터 판독가능 매체 (110) (도 1) 로부터 획득될 수도 있다. CPB 메모리 (320) 는 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터 (예를 들어, 구문 요소) 를 저장하는 CPB 를 포함할 수도 있다. 또한, CPB 메모리 (320) 는 비디오 디코더 (300) 의 다양한 유닛들로부터의 출력들을 나타내는 일시적 데이터와 같은, 코딩된 화상의 구문 요소외의 비디오 데이터를 저장할 수도 있다. DPB (314) 는 일반적으로, 인코딩된 비디오 비트스트림의 후속 데이터 또는 화상들을 디코딩할 때 참조 비디오 데이터로서 비디오 디코더 (300) 가 출력 및/또는 사용할 수도 있는 디코딩된 화상들을 저장한다. CPB 메모리 (320) 및 DPB (314) 는 다양한 메모리 디바이스들, 이를 테면, SDRAM (synchronous DRAM) 을 포함하는 DRAM (Dynamic random access memory), MRAM (magnetoresistive RAM), RRAM (resistive RAM) , 또는 다른 유형들의 메모리 디바이스들 중 임의의 것에 의해 형성될 수도 있다. CPB 메모리 (320) 및 DPB (314) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, CPB 메모리 (320) 는 비디오 디코더 (300) 의 다른 컴포넌트들과 온-칩되거나, 또는 그 컴포넌트들에 대해 오프-칩될 수도 있다.

[0105] 추가적으로 또는 대안적으로, 일부 예들에서, 비디오 디코더 (300) 는 메모리 (120) (도 1) 로부터 코딩된 비디오 데이터를 취출할 수도 있다. 즉, 메모리 (120) 는 CPB 메모리 (320) 로 위에서 논의된 바와 같이 데이터를 저장할 수도 있다. 마찬가지로, 메모리 (120) 는 비디오 디코더 (300) 의 기능성의 일부 또는 전부가 비디오 디코더 (300) 의 프로세싱 회로에 의해 실행되도록 소프트웨어에서 구현될 때, 비디오 디코더 (300) 에 의해 실행될 명령들을 저장할 수도 있다.

[0106] 도 4 에 도시된 다양한 유닛들은 비디오 디코더 (300) 에 의해 수행되는 동작들의 이해를 돕기 위해 예시된다. 그 유닛들은 고정 기능 회로들, 프로그래밍가능 회로들, 또는 이들의 조합으로서 구현될 수도 있다. 도 3 와 유사하게, 고정 기능 회로들은 특정 기능성을 제공하는 회로들을 지칭하고, 수행될 수 있는 동작들에 대해 미리설정된다. 프로그램가능 회로들은 다양한 태스크들을 수행하도록 프로그램될 수 있는 회로들을 지칭하며, 수행될 수 있는 동작들에서 유연한 기능성을 제공한다. 예를 들어, 프로그래밍가능 회로들은, 프로그래밍가능 회로들로 하여금, 소프트웨어 또는 펌웨어의 명령들에 의해 정의된 방식으로 동작하게 하는 소프트웨어 또는 펌웨어를 실행할 수도 있다. 고정 기능 회로들은 (예를 들어, 파라미터들을 수신하거나 또는 파라미터들을 출력하기 위해) 소프트웨어 명령들을 실행할 수도 있지만, 고정 기능 회로들이 수행하는 동작들의 타입들은 일반적으로 불변이다. 일부 예들에서, 유닛들 중 하나 이상은 별개의 회로 블록들 (고정 기능 또는 프로그래밍가능) 일 수도 있고, 일부 예들에서, 하나 이상의 유닛들은 집적 회로들일 수도 있다.

[0107] 비디오 디코더 (300) 는 프로그래밍가능 회로들로부터 형성된, ALU들, EFU들, 디지털 회로들, 아날로그 회로들, 및/또는 프로그래밍가능 코어들을 포함할 수도 있다. 비디오 디코더 (300) 의 동작들이 프로그래밍가능 회로들 상에서 실행하는 소프트웨어에 의해 수행되는 예들에서, 온-칩 또는 오프-칩 메모리는 비디오 디코더 (300) 가 수신하고 실행하는 소프트웨어의 명령들 (예를 들어, 오브젝트 코드) 을 저장할 수도 있다.

[0108] 엔트로피 디코딩 유닛 (302) 은 인코딩된 비디오 데이터를 CPB 로부터 수신하고, 비디오 데이터를 엔트로피 디코딩하여 구문 요소들을 재생성할 수도 있다. 예측 프로세싱 유닛 (304), 역 양자화 유닛 (306), 역 변환 프로세싱 유닛 (308), 재구성 유닛 (310), 및 필터 유닛 (312) 은 비트스트림으로부터 추출된 구문 요소들에 기초하여 디코딩된 비디오 데이터를 생성할 수도 있다.

[0109] 일반적으로, 비디오 디코더 (300) 는 블록 바이 블록 (block-by-block) 기반으로 화상을 재구성한다. 비디오 디코더 (300) 는 개별적으로 각각의 블록에 대해 재구성 동작을 수행할 수도 있다 (여기서 현재 재구성되는, 즉 디코딩되는 블록은 "현재 블록" 으로 지칭될 수도 있음).

[0110] 엔트로피 디코딩 유닛 (302) 은 양자화 파라미터 (QP) 및/또는 변환 모드 표시(들)와 같은 변환 정보 뿐만 아니라, 양자화된 변환 계수 블록의 양자화된 변환 계수들을 정의하는 구문 요소들을 엔트로피 디코딩할 수도 있다.

역 양자화 유닛 (306) 은 양자화된 변환 계수 블록과 연관된 QP 를 사용하여, 양자화도 그리고, 마찬가지로, 역 양자화 유닛 (306) 이 적용할 역 양자화도를 결정할 수도 있다. 역 양자화 유닛 (306) 은 예를 들어, 양자화된 변환 계수들을 역 양자화하기 위해 비트단위 좌측-시프트 동작을 수행할 수도 있다. 이에 의해, 역 양자화 유닛 (306) 은 변환 계수들을 포함하는 변환 계수 블록을 형성할 수도 있다.

[0111] 역 양자화 유닛 (306) 이 변환 계수 블록을 형성한 후, 역 변환 프로세싱 유닛 (308) 은 현재 블록과 연관된 잔차 블록을 생성하기 위해 변환 계수 블록에 하나 이상의 역 변환들을 적용할 수도 있다. 예를 들어, 역 변환 프로세싱 유닛 (308) 은 역 DCT, 역 정수 변환, 역 Karhunen-Loeve 변환 (KLT), 역 회전 변환, 역 방향성 변환, 또는 다른 역 변환을 계수 블록에 적용할 수도 있다.

[0112] 또한, 예측 프로세싱 유닛 (304) 은 엔트로피 디코딩 유닛 (302) 에 의해 엔트로피 디코딩된 예측 정보 구문 요소들에 따라 예측 블록을 생성한다. 예를 들어, 예측 정보 구문 요소들이 현재 블록이 인터 예측됨을 표시하면, 모션 보상 유닛 (316) 은 예측 블록을 생성할 수도 있다. 이 경우에, 예측 정보 구문 요소들은 참조 블록을 취출할 DPB (314) 에서의 참조 화상 뿐만 아니라 현재 화상에서의 현재 블록의 위치에 대한 참조 화상에서의 참조 블록의 위치를 식별하는 모션 벡터를 표시할 수도 있다. 모션 보상 유닛 (316) 은 일반적으로 모션 보상 유닛 (224) (도 3) 에 대하여 설명된 것과 실질적으로 유사한 방식으로 인터 예측 프로세스를 수행할 수도 있다.

[0113] 일부 예에서, 예측 프로세싱 유닛 (304) 은 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하도록 구성될 수도 있다. 이 예에서, 예측 프로세싱 유닛 (304) 은 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하도록 구성될 수도 있다. 예측 프로세싱 유닛 (304) 은 예측 프로세싱 유닛 (304) 이 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 예측 프로세싱 유닛 (304) 이 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하도록 구성될 수도 있다. 이러한 방식으로, 예측 프로세싱 유닛 (304) 은 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷 양자 모두를 위한 가중치 세트를 사용할 수도 있으며, 이는 상이한 크로마 포맷의 처리를 통일하는 데 도움이 될 수 있다. 상이한 크로마 포맷의 처리를 통일하면 코덱 프로세싱 오버헤드를 줄이는 데 도움이 될 수 있다.

[0114] 픽셀 블렌딩을 적용하기 위해, 예측 프로세싱 유닛 (304) 은 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하도록 구성될 수도 있다.

가중 평균을 결정하는 예시적인 프로세스는 아래에서 더 자세히 설명된다 (예를 들어, 도 7a 내지 도 7c 참조).

[0115] 예측 정보 구문 요소가 현재 블록이 인트라 예측됨을 표시하면, 인트라 예측 유닛 (318) 은 예측 정보 구문 요소에 의해 표시된 인트라 예측 모드에 따라 예측 블록을 생성할 수도 있다. 다시, 인트라 예측 유닛 (318) 은 일반적으로 인트라 예측 유닛 (226) (도 3) 에 대하여 설명된 것과 실질적으로 유사한 방식으로 인트라 예측 프로세스를 수행할 수도 있다. 인트라 예측 유닛 (318) 은 DPB (314) 로부터 현재 블록을 위한 이웃하는 샘플들의 데이터를 취출할 수도 있다.

[0116] 일부 예에서, 인트라 예측 유닛 (318) 은 화상의 영역에 대해 필터링되지 않은 참조 샘플을 획득하도록 구성될 수도 있다. 이 예에서, 인트라 예측 유닛 (318) 는 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플들을 위한 필터링되지 않은 참조 샘플들의 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 인트라 참조 샘플 평활화가 비활성화될 때, 인트라 예측 유닛 (318) 은 크로마 샘플을 위해 필터링되지 않은 참조 샘플만을 사용할 수도 있다. 인트라 예측 유닛 (318) 은 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다. 즉, 예를 들어, YUV 4:4:4 포맷 및 YUV 4:4:4 포맷 모두의 크로마 성분에 대해, 인트라 예측 유닛 (318) 은 크로마 샘플을 위해 필터링되지 않은 참조 샘플만을 사용할 수도 있다. 이러한 방식으로, 비디오 데이터를 인코딩 및 디코딩하는 복잡성이 감소될 수 있고, 이는 인트라 예측 유닛 (318) 이 비디오 데이터를 코딩 (예를 들어, 인코딩 또는 디코딩) 하는 시간의 양을 감소시키거나 및/또는 인트라 예측 유닛 (318) 의 전력 소비를 감소시킬 수 있다.

[0117] 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화를 위해 여기에 설명된 기술은 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성하기 위해 여기서 설명된 기술과 함께 또는 별도로 사용될 수도 있다. 예를 들어, 인트라 예측 유닛 (318) 은 예측 프로세싱 유닛 (304) 이 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하는 것과 조합하여 화상의 영역을 위해 필터링되지 않은 참조 샘플들을 획득하도록 구성될 수 있다. 일부 예에서, 인트라 예측 유닛 (318) 은 조합해서 화상의 영역을 위해 필터링되지 않은 참조 샘플들을 획득하도록 구성될 수 있고 예측 프로세싱 유닛 (304) 은 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하는 것을 억제한다.

[0118] 재구성 유닛 (310) 은 예측 블록 및 잔차 블록을 사용하여 현재 블록을 재구성한다. 예를 들어, 재구성 유닛 (310) 은 잔차 블록의 샘플들을 예측 블록의 대응하는 샘플들에 가산하여 현재 블록을 재구성할 수도 있다.

[0119] 필터 유닛 (312) 은 재구성된 블록들에 대해 하나 이상의 필터 동작들을 수행할 수도 있다. 예를 들어, 필터 유닛 (312) 은 재구성된 블록들의 에지들을 따라 블록키니스 아티팩트를 감소시키기 위해 디블록킹 동작들을 수행할 수도 있다. 필터 유닛 (312) 의 동작들이 모든 예들에서 반드시 수행되는 것은 아니다.

[0120] 비디오 디코더 (300) 는 DPB (314) 에 재구성된 블록들을 저장할 수도 있다. 위에 논의된 바와 같이, DPB (314) 는 예측 프로세싱 유닛 (304) 에, 인트라 예측을 위한 현재 화상 및 후속 모션 보상을 위한 이전에 디코딩된 화상들의 샘플들과 같은 참조 정보를 제공할 수도 있다. 또한, 비디오 디코더 (300) 는 도 1 의 디스플레이 디바이스 (118) 와 같은 디스플레이 디바이스 상에의 후속 프리젠테이션을 위해 DPB 로부터 디코딩된 화상들을 출력할 수도 있다.

[0121] 일부 예에서, 비디오 디코더 (300) 는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로에 구현되고 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하도록 구성된 하나 이상의 프로세서를 포함하는 비디오 데이터를 인코딩하도록 구성된 디바이스의 예를 나타낸다. 이 예에서, 비디오 디코더 (300) 는 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할하도록 구성될 수도 있다. 비디오 디코더 (300) 는 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용하도록 구성될 수도 있다. 이런 이유로, 비디오 디코더 (300) 는 YUV 4:2:0 포맷 및 YUV 4:4:4 포맷 양자 모두를 위한 가중치 세트를 사용할 수도 있으며, 이는 상이한 크로마 포맷의 처리를 통일하는 데 도움이 될 수 있다. 상이한 크로마 포맷의 처리를 통일하면 코덱 프로세싱 오버헤드를 줄이는 데 도움이 될 수 있다. 픽셀 블렌딩을 적용하기 위해, 비디오 디코더 (300) 는, 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 각각 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의 가중 평균을, YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여, 결정하도록 구성될 수도 있다.

[0122] 일부 예에서, 비디오 디코더 (300) 는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로에서 구현되고 화상의 영역을 위한 필터링되지 않은 참조 샘플을 획득하도록 구성된 하나 이상의 프로세서를 포함하는 비디오 데이터를 인코딩하도록 구성된 디바이스의 예를 나타낸다. 이 예에서, 비디오 디코더 (300) 는 YUV 4:2:0 포맷의 그리고 YUV 4:4:4 포맷의 크로마 샘플들을 위해 필터링되지 않은 참조 샘플들의 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 비디오 디코더 (300) 는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다. 즉, 예를 들어, YUV 4:4:4 포맷 및 YUV 4:4:4 포맷 모두의 크로마 성분에 대해, 비디오 디코더 (300) 는 크로마 샘플을 위해 필터링되지 않은 참조 샘플만을 사용할 수도 있다. 이런 식으로, 비디오 데이터를 디코딩하는 복잡성이 감소될 수 있고, 이는 비디오 디코더 (300) 가 비디오 데이터를 디코딩하는 시간의 양을 감소시키거나 및/또는 비디오 디코더 (300) 의 전력 소비를 감소시킬 수 있다. 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화를 위해 여기에 설명된 기술은 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성하기 위해 여기서 설명된 기술과 함께 또는 별도로 사용될 수도 있다.

[0123] 도 5a 는 본 개시의 기법에 따라, 인터 예측에 기초하여 코딩 유닛 (CU) (429) 을 제 1 삼각형 형상 파티션 (433) 및 제 2 삼각형 형상 파티션 (435) 으로 분할하는 제 1 예를 예시하는 개념도이다. 도 5a 의 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 인터 예측이 활성화될 때 그리고

삼각형 예측 유닛 모드가 활성화될 때 코딩 유닛 (429) 을 제 1 삼각형 형상의 파티션 (433) 및 제 2 삼각형 형상의 파티션 (435) 으로 분할할 수도 있다. 삼각형 예측 단위 모드의 예는 예를 들어 J. Chen, Y. Ye, S. Kim, "Algorithm description for Versatile Video Coding and Test Model 4 (VTM 4)" Joint Video Exploration Team of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JVET-M1002, Jan. 2019) 및 다른 곳에서 찾아볼 수도 있다. 예를 들어, 삼각형 예측 유닛 모드의 활성화에 기초하여, 비디오 코더는 CU 를 2개의 삼각형 형상의 파티션으로 분할할 수도 있고 각각의 파티션은 참조 인덱스 및 단방향 모션 벡터를 갖는다. 비디오 코더는 2개의 모션 보상된 파티션을 예측 샘플들의 블록으로 결합하도록 구성될 수도 있고, 분할 에지의 에지 상에서 픽셀 블렌딩이 수행된다. 도 5a 의 예에서, 제 1 삼각형 형상의 파티션 (433) 은 CU (429) 의 우상측 코너 샘플을 포함하고 제 2 삼각형 형상의 파티션 (435) 은 CU (429) 의 좌하측 코너 샘플을 포함한다.

[0124] 다르게 말하면, 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛 (429) 을 생성하도록 구성될 수도 있다. 이 예에서, 비디오 코더는 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛 (429) 을 제 1 삼각형 형상의 파티션 (433) 및 제 2 삼각형 형상의 파티션 (435) 으로 분할할 수도 있다.

[0125] 도 5b 는 본 개시의 기법에 따라, 인터 예측에 기초하여 코딩 유닛 (431) 을 제 1 삼각형 형상 파티션 (434) 및 제 2 삼각형 형상 파티션 (436) 으로 분할하는 제 2 예를 예시하는 개념도이다.

[0126] 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛 (431) 을 생성하도록 구성될 수도 있다. 이 예에서, 비디오 코더는 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛 (431) 을 제 1 삼각형 형상의 파티션 (434) 및 제 2 삼각형 형상의 파티션 (436) 으로 분할할 수도 있다. 도 5b 의 예에서, 제 1 삼각형 형상의 파티션 (433) 은 CU (429) 의 좌상측에 배열된다. 도 55 의 예에서, 제 1 삼각형 형상의 파티션 (434) 은 CU (431) 의 좌상측 코너 샘플을 포함하고 제 2 삼각형 형상의 파티션 (435) 은 CU (431) 의 우하측 코너 샘플을 포함한다.

[0127] 도 6 은 본 개시의 기술들에 따른, 하나의 가중치 세트를 사용한 픽셀 블렌딩을 예시하는 개념도이다. 비디오 코더 (예를 들어, 비디오 인코더 (200) 및/또는 비디오 디코더 (300)) 는 도 6에 도시된 바와 같이 하나의 가중치 세트를 사용한 도 6의 예시적인 픽셀 블렌딩을 수행하도록 구성될 수도 있다. 비디오 코더는 예를 들어, 2개의 삼각형 파티션의 모션 정보에 기초하여 콜로케이트된 모션 보상 픽셀의 가중 평균화에 의해 블렌딩 영역에서 픽셀을 생성하도록 구성될 수도 있다. 본 명세서에 사용된 바와 같이, 콜로케이트된 픽셀은 제 2 파티션 (예: 제 2 삼각형 형상의 파티션) 의 제 2 픽셀과 동일한 픽셀 위치에 위치하는 제 1 파티션 (예: 제 1 삼각형 형상의 파티션) 의 제 1 픽셀을 나타낼 수도 있다. 가중 평균은 도 7a 내지 도 7c 에서 추가로 설명된다.

[0128] 즉, 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 비디오 데이터의 블록의 크로마 성분을 위한 예측된 블록을 생성하기 위해 가중치 세트를 사용하여 픽셀 블렌딩을 적용할 수도 있다. 이 예에서, 픽셀 블렌딩을 적용하기 위해, 비디오 코더는, 제 1 삼각형 형상 파티션 및 제 2 삼각형 형상 파티션의 모션 정보에 기초하여, 제 1 삼각형 형상의 파티션 ("P_1"로 도시됨) 및 제 2 삼각형 형상의 파티션 ("P_2"로 도시됨) 의 콜로케이트된 모션 보상 픽셀의, 가중치 세트를 사용한, 가중 평균을 결정할 수도 있다.

[0129] 예를 들어, 도 6 에 도시된 바와 같이, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 1에 따라 각각의 픽셀에 대해 각각의 값 P 를 계산함으로써 루마 성분을 위한 예측 블록 (537) 및/또는 루마 성분을 위한 예측 블록 (538) 의 "2"로 마크된 픽셀에 대한 픽셀 값 P를 결정할 수도 있다.

[0130] $$P = \frac{2}{8} * P_1 + \frac{6}{8} * P_2$$ 등식 1

[0131] 식중 P_1 은 이 예에서 "2" 로 마크된 각각의 픽셀과 콜로케이트된 제 1 삼각형 형상의 파티션의 제 1 콜로케이트된 모션 보상 픽셀의 제 1 참조 픽셀 값이고, P_2 는 이 예에서 "2" 로 마크된 각각의 픽셀과 콜로케이트된 제 1 삼각형 형상의 파티션의 제 2 콜로케이트된 모션 보상 픽셀의 제 2 참조 픽셀 값이다.

[0132] 도 6 에 더 도시된 바와 같이, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등

식 2에 따라 각각의 픽셀에 대해 각각의 값 P 를 계산함으로써 루마 성분을 위한 예측 블록 (537) 및/또는 루마 성분을 위한 예측 블록 (538) 의 "4"로 마크된 픽셀에 대한 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{4}{8} * P_1 + \frac{4}{8} * P_2$$

[0133] 등식 2

[0134] 식중 P_1 은 이 예에서 "4" 로 마크된 각각의 픽셀과 콜로케이트된 제 1 삼각형 형상의 파티션의 제 1 콜로케이트된 모션 보상 픽셀의 제 1 참조 픽셀 값이고, P_2 는 이 예에서 "4" 로 마크된 각각의 픽셀과 콜로케이트된 제 1 삼각형 형상의 파티션의 제 2 콜로케이트된 모션 보상 픽셀의 제 2 참조 픽셀 값이다.

[0135] 도 6 에 더 도시된 바와 같이, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 3에 따라 각각의 픽셀에 대해 각각의 값 P 를 계산함으로써 루마 성분을 위한 예측 블록 (537) 및/또는 루마 성분을 위한 예측 블록 (538) 의 "7"로 마크된 픽셀에 대한 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{7}{8} * P_1 + \frac{1}{8} * P_2$$

[0136] 등식 3

[0137] 식중 P_1 은 이 예에서 "7" 로 마크된 각각의 픽셀과 콜로케이트된 제 1 삼각형 형상의 파티션의 제 1 콜로케이트된 모션 보상 픽셀의 제 1 참조 픽셀 값이고, P_2 는 이 예에서 "7" 로 마크된 각각의 픽셀과 콜로케이트된 제 1 삼각형 형상의 파티션의 제 2 콜로케이트된 모션 보상 픽셀의 제 2 참조 픽셀 값이다.

[0138] 값 "1" 및 "6" 은 등식 1-3에 대응하는 등식을 나타낸다. 즉, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 4에 따라 각각의 픽셀에 대해 각각의 값 P 를 계산함으로써 루마 성분을 위한 예측 블록 (537) 및/또는 루마 성분을 위한 예측 블록 (538) 의 "1"로 마크된 픽셀에 대한 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{1}{8} * P_1 + \frac{7}{8} * P_2$$

[0139] 등식 4

[0140] 식중 P_1 은 제 1 삼각형 형상 파티션의 제 1 콜로케이트된 모션 보상 픽셀의 제 1 참조 픽셀 값이고 P_2 는 제 1 삼각형 형상의 파티션의 제 2 콜로케이트된 모션 보상 픽셀의 제 2 참조 픽셀 값이다.

[0141] 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 5에 따라 각각의 픽셀에 대해 각각의 값 P 를 계산함으로써 루마 성분을 위한 예측 블록 (537) 및/또는 루마 성분을 위한 예측 블록 (538) 의 "6"로 마크된 픽셀에 대한 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{6}{8} * P_1 + \frac{2}{8} * P_2$$

[0142] 등식 5

[0143] 식중 P_1 은 제 1 삼각형 형상 파티션의 제 1 콜로케이트된 모션 보상 픽셀의 제 1 참조 픽셀 값이고 P_2 는 제 1 삼각형 형상의 파티션의 제 2 콜로케이트된 모션 보상 픽셀의 제 2 참조 픽셀 값이다.

[0144] 도 6에 예시된 예는 루마를 위한 {7/8, 6/8, 4/8, 2/8, 1/8} 및 크로마를 위한 {7/8, 4/8, 1/8} 의 예시적인 가중치 세트를 보여주지만, 다른 예는 상이한 가중치를 사용할 수도 있다. 예를 들어, 2개의 가중치 세트가 있을 수도 있고 각각의 세트에서 크로마 가중치와 루마 가중치가 따로따로 정의될 수도 있다:

[0145] · 제 1 세트: 루마를 위한 {7/8, 6/8, 4/8, 2/8, 1/8} 및 크로마를 위한 {7/8, 4/8, 1/8}.

[0146] · 제 2 세트: 루마를 위한 {7/8, 6/8, 5/8, 4/8, 3/8, 2/8, 1/8} 및 크로마를 위한 {6/8, 4/8, 2/8}.

[0147] 제 2 가중치 값 세트 (예를 들어, {6/8, 4/8, 2/8}) 은 예시의 목적으로만 도 7a 내지 도 7c를 참조하여 설명된다.

[0148] 도 7a 는 본 개시의 기술에 따른, 도 6의 픽셀 블렌딩을 위한 제 1 삼각형 형상의 파티션 (639) 의 추가 세부사항을 예시하는 개념도이다. 도시된 바와 같이, 제 1 삼각형 형상의 파티션 (639) 은 제 1 삼각형 형상의 파

티션 (639) 의 콜로케이트된 모션 보상된 픽셀 A, B 및 C를 포함한다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 제 1 삼각형 형상 파티션 (639) 을 위한 모션 정보 (예를 들어, 단방향 모션 벡터) 를 사용하여 제 1 삼각형 형상의 파티션 (639) 의 콜로케이트된 모션 보상된 픽셀 A, B 및 C를 생성할 수도 있다. 이 예에서, 비디오 코더는 제 1 픽셀 그룹 (642) 에 있는 픽셀에 가중 평균화를 적용하지 않을 수도 있다.

[0149] 도 7b 는 본 개시의 기술에 따른, 도 6의 픽셀 블렌딩을 위한 제 2 삼각형 형상의 파티션 (640) 의 추가 세부사항을 예시하는 개념도이다. 도시된 바와 같이, 제 2 삼각형 형상의 파티션 (640) 은 제 2 삼각형 형상의 파티션 (640) 의 콜로케이트된 모션 보상된 픽셀 A, B 및 C를 포함한다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 제 2 삼각형 형상 파티션 (640) 을 위한 모션 정보 (예를 들어, 단방향 모션 벡터) 를 사용하여 제 2 삼각형 형상의 파티션 (640) 의 콜로케이트된 모션 보상된 픽셀 A, B 및 C를 생성할 수도 있다. 이 예에서, 비디오 코더는 제 2 픽셀 그룹 (643) 에 있는 픽셀에 가중 평균화를 적용하지 않을 수도 있다.

[0150] 도 7c 는 본 개시의 기술에 따른, 도 6의 픽셀 블렌딩을 사용하여 형성된 블록 (641) 의 파티션의 추가 세부사항을 예시하는 개념도이다. 이 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 제 1 삼각형 형상의 파티션 (639) 및 제 2 삼각형 형상의 파티션 (640) 의 모션 정보에 기초하여 제 1 삼각형 형상의 파티션 (639) 및 제 2 삼각형 형상의 파티션 (640) 의 콜로케이트된 모션 보상된 픽셀의 가중 평균을 결정할 수도 있다.

[0151] 도 7c의 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 제 2 가중치 값 세트 (예를 들어, 크로마를 위한 {6/8, 4/8, 2/8}) 을 사용할 수도 있다. 예를 들어, 비디오 코더는 등식 6에 따라 픽셀 값 P를 계산함으로써 예측 블록 (641) 의 픽셀 'A'의 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{4}{8} * P_1 + \frac{4}{8} * P_2,$$

[0152] 등식 6

[0153] 식중 P_1 은 제 1 삼각형 형상의 파티션 (639) 의 제 1 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7a 의 제 1 삼각형 형상의 파티션 (639) 의 'A') 의 제 1 참조 픽셀 값이고 P_2 는 제 2 삼각형 형상의 파티션 (640) 의 제 2 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7b 의 제 2 삼각형 형상의 파티션 (640) 의 'A') 의 제 2 참조 픽셀 값이다.

[0154] 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 7에 따라 픽셀 값 P를 계산함으로써 예측 블록 (641) 의 픽셀 'B' 의 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{2}{8} * P_1 + \frac{6}{8} * P_2$$

[0155] 등식 7

[0156] 식중 P_1 은 제 1 삼각형 형상의 파티션 (639) 의 제 1 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7a 의 제 1 삼각형 형상의 파티션 (639) 의 'B') 의 제 1 참조 픽셀 값이고 P_2 는 제 2 삼각형 형상의 파티션 (640) 의 제 2 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7b 의 제 2 삼각형 형상의 파티션 (640) 의 'B') 의 제 2 참조 픽셀 값이다.

[0157] 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 8에 따라 픽셀 값 P를 계산함으로써 예측 블록 (641) 의 픽셀 'C' 의 픽셀 값 P를 결정할 수도 있다.

$$P = \frac{6}{8} * P_1 + \frac{2}{8} * P_2$$

[0158] 등식 8

[0159] 식중 P_1 은 제 1 삼각형 형상의 파티션 (639) 의 제 1 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7a 의 제 1 삼각형 형상의 파티션 (639) 의 'C') 의 제 1 참조 픽셀 값이고 P_2 는 제 2 삼각형 형상의 파티션 (640) 의 제 2 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7b 의 제 2 삼각형 형상의 파티션 (640) 의 'C') 의 제 2 참조 픽셀 값이다.

[0160] 일부 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 제 1 가중치 값 세트 (예를 들어, 크로마를 위한 {7/8, 4/8, 1/8}) 을 사용할 수도 있다. 예를 들어, 비디오 코더는 등식 9에 따라 픽셀 값 P를 계산함으로써 예측 블록 (641) 의 픽셀 'A'의 픽셀 값 P를 결정할 수도 있다.

[0161] $$P = \frac{4}{8} * P_1 + \frac{4}{8} * P_2,$$ 등식 9

[0162] 식중 P_1 은 제 1 삼각형 형상의 파티션 (639) 의 제 1 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7a 의 제 1 삼각형 형상의 파티션 (639) 의 'A') 의 제 1 참조 픽셀 값이고 P_2 는 제 2 삼각형 형상의 파티션 (640) 의 제 2 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7b 의 제 2 삼각형 형상의 파티션 (640) 의 'A') 의 제 2 참조 픽셀 값이다.

[0163] 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 10에 따라 픽셀 값 P를 계산함으로써 예측 블록 (641) 의 픽셀 'B' 의 픽셀 값 P를 결정할 수도 있다.

[0164] $$P = \frac{1}{8} * P_1 + \frac{7}{8} * P_2$$ 등식 10

[0165] 식중 P_1 은 제 1 삼각형 형상의 파티션 (639) 의 제 1 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7a 의 제 1 삼각형 형상의 파티션 (639) 의 'B') 의 제 1 참조 픽셀 값이고 P_2 는 제 2 삼각형 형상의 파티션 (640) 의 제 2 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7b 의 제 2 삼각형 형상의 파티션 (640) 의 'B') 의 제 2 참조 픽셀 값이다.

[0166] 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 등식 11에 따라 픽셀 값 P를 계산함으로써 예측 블록 (641) 의 픽셀 'C' 의 픽셀 값 P를 결정할 수도 있다.

[0167] $$P = \frac{7}{8} * P_1 + \frac{1}{8} * P_2$$ 등식 11

[0168] 식중 P_1 은 제 1 삼각형 형상의 파티션 (639) 의 제 1 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7a 의 제 1 삼각형 형상의 파티션 (639) 의 'C') 의 제 1 참조 픽셀 값이고 P_2 는 제 2 삼각형 형상의 파티션 (640) 의 제 2 콜로케이트된 모션 보상 픽셀 (예를 들어, 도 7b 의 제 2 삼각형 형상의 파티션 (640) 의 'C') 의 제 2 참조 픽셀 값이다.

[0169] 일부 비디오 코더에서, YUV 4:4:4 포맷을 사용할 때 루마와 크로마의 가중치가 통일될 수도 있다. 그래서, 비디오 인코더 (예를 들어, 비디오 인코더(200)) 가 YUV 4:4:4 포맷의 비디오 신호를 인코딩하는 경우, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 각각의 가중치 세트에 대해 크로마 부분을 사용하지 않고 루마 및 크로마 에지 양자 모두에 각각의 가중치 세트의 루마 부분을 적용하도록 구성될 수도 있다.

[0170] 비디오 코덱에서, 인트라 예측에 사용되는 샘플은 동일한 화상의 재구성된 영역으로부터 획득될 수 있고 획득된 샘플은 예측 신호의 품질을 향상시키기 위해 일부 필터를 적용하여 평활화될 수도 있다. 예를 들어, HEVC 의 인트라 참조 샘플 평활화에 대한 예시적인 설명은 J. Lainema, F. Bossen, W. Han, J. Min and K. Ugur, "Intra Coding of the HEVC Standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1792-1801, Dec. 2012 에서 찾아볼 수 있다. 일부 예에서, 비디오 인코더 (200) 또는 비디오 디코더 (300) 는 다른 인트라 참조 샘플 평활화 예를 사용하도록 구성될 수도 있다.

[0171] 일부 VVC 설계에서, 크로마 포맷이 YUV 4:2:0인 경우, 크로마 성분은 항상 필터링되지 않은 참조 샘플을 사용할 것이다. 그러나, YUV 4:4:4 의 경우, 크로마 성분은 필터링된 참조 샘플을 사용해야 하는지 여부를 결정하기 위해 루마와 동일한 규칙을 받는다. 달리 말하면, 일부 VVC 설계에서, 크로마 성분이 YUV 4:2:0 포맷으로 생성되어야 할 때, 비디오 인코더 (200) 또는 비디오 디코더 (300) 는 크로마 성분을 생성하기 위해 항상 필터링되지 않은 참조 샘플을 사용하도록 구성될 수도 있다. 이 예에서, 크로마 성분이 YUV 4:4:4 포맷으로 생성되어야 할 때, 비디오 인코더 (200) 또는 비디오 디코더 (300) 는 크로마 성분을 생성하기 위해 필터링되지

않은 참조 샘플 또는 필터링된 참조 샘플을 사용할지 여부를 결정하도록 구성될 수도 있다.

[0172] 예를 들어, 루마 샘플들을 생성하기 위한 인트라 참조 샘플 평활화를 활성화하기로 결정하는 것에 응답하여, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 필터링된 참조 샘플을 생성하기 위해 필터링되지 않은 참조 샘플들에 대해 하나 이상의 필터 동작들을 수행하도록 구성될 수도 있다. 이 예에서, 비디오 코더는 필터링된 참조 샘플들에 기초하여 화상의 블록에 대한 루마 샘플들을, 인트라 예측을 사용하여, 생성하도록 구성될 수도 있다. 일부 예에서, 필터링되지 않은 참조 샘플들에 대해 하나 이상의 필터 동작들을 수행하는 것은 필터링되지 않은 참조 샘플들에 대해 하나 이상의 디블로킹 동작들을 수행하는 것을 포함할 수도 있다.

[0173] 그래서, 일부 VVC 설계는 삼각형 PU 모드에서 상이한 크로마 포맷을 상이하게 처리할 수도 있으며 코덱 설계에 추가 복잡성이 도입된다. 동시에, 루마 성분을 위해 정의된 동작은 크로마 성분을 위해 정의된 동작보다 더 복잡할 수도 있다 (예를 들어, 도 6에 도시된 바와 같이, 루마를 위한 블렌딩 영역은 3개의 샘플의 폭을 갖는 반면 크로마를 위한 블렌딩 영역은 2개 샘플이다).

[0174] 본 개시의 기법에 따르면, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 상이한 크로마 포맷들에 대한 크로마 성분들의 처리를 통일하도록 구성될 수도 있다. 예를 들어, 삼각형 PU 모드의 경우, 비디오 코더 (인코더 (200) 또는 디코더 (300)) 는 YUV 4:4:4 포맷의 크로마 성분을 위해 루마 성분을 위한 것들 대신에 YUV 4:2:0 크로마 성분을 위한 가중치를 사용하도록 구성될 수도 있다. 현재 VVC 드래프트의 예에서, YUV 4:2:0 포맷의 크로마 성분을 위해 설계된 2개의 가중치 세트는 다음과 같다:

[0175]
1) {1,4,7}
2) {2,4,6}

[0176] 이 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더(300)) 는 YUV 4:4:4 크로마 성분 및 YUV 4:2:0 크로마 성분을 동일하게 핸들링하도록 구성될 수도 있다. 비디오 코더는 이들 2개의 가중치 세트의 조합을 YUV4:4:4 크로마 성분에 적용하도록 구성될 수도 있다. 예를 들어, 비디오 코더는 가중치 세트 {1, 4, 7}, 가중치 세트 {2, 4, 6}, 또는 가중치 세트 {1, 4, 7} 및 {2, 4, 6} 을 적용하도록 구성될 수도 있다.

[0177] 다르게 말하면, 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 비디오 코더가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 비디오 코더가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 비디오 데이터의 블록의 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용할 수도 있다. 이 예에서, 비디오 코더는 가중치 세트 {1, 4, 7}, 가중치 세트 {2, 4, 6}, 또는 가중치 세트 {1, 4, 7} 및 {2, 4, 6} 을 적용하도록 구성될 수도 있다.

[0178] 일부 예에서, 픽셀 블렌딩을 적용하는 것은 도 6 및 도 7a 내지 도 7c 에서 상세히 논의된 바와 같이 가중 평균을 결정하는 것을 포함한다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정하도록 구성될 수도 있다.

[0179] 삼각형 병합 모드를 위한 가중 샘플 예측 프로세스의 예는 다음과 같다.

[0180] 이 프로세스에의 입력들은:

[0181] - 현재 코딩 블록의 너비와 높이를 지정하는 2개의 변수 nCbW 및 nCbH,

[0182] - 2개의 (nCbW)x(nCbH) 어레이 predSamplesLA 및 predSamplesLB,

[0183] - 파티션 방향을 지정하는 변수 triangleDir,

[0184] - 색상 성분 인덱스를 지정하는 변수 cIdx.

[0185] 이 프로세스의 출력은 예측 샘플 값의 (nCbW)x(nCbH) 어레이 pbSamples 이다.

[0186] 변수 nCbR 는 다음과 같이 도출된다:

[0187] $$nCbR = (nCbW > nCbH) \ ? \ (nCbW / nCbH) \ : \ (nCbH / nCbW) \quad (8\text{-}841)$$

[0188] 변수 bitDepth 는 다음과 같이 도출된다:

[0189] - cIdx 가 0 과 동일한 경우, bitDepth 은 $BitDepth_Y$와 동일하게 설정된다.

[0190] - 그렇지 않으면, bitDepth 는 $BitDepth_C$ 와 동일하게 설정된다.

[0191] 변수 shift1 및 offset1은 다음과 같이 도출된다:

[0192] - 변수 shift1은 Max(5, 17 - bitDepth) 와 동일하게 설정된다.

[0193] - 변수 offset1은 1 << (shift1 - 1) 와 동일하게 설정된다.

[0194] triangleDir, wS 및 cIdx 의 값에 따라, 예측 샘플 pbSamples[x][y] (여기서 $x = 0..nCbW - 1$ 및 $y = 0..nCbH - 1$) 는 다음과 같이 도출된다:

[0195] - 변수 wIdx 는 다음과 같이 도출된다:

[0196] - cIdx 가 0 과 동일하고 triangleDir 가 0 과 동일한 경우, 다음이 적용된다:

[0197] $$wIdx = (nCbW > nCbH) \ ? \ (Clip3(0, 8, (x / nCbR - y) + 4)) \quad (8\text{-}842)$$
$$: (Clip3(0, 8, (x - y / nCbR) + 4))$$

[0198] - 그렇지 않고, cIdx 가 0 과 동일하고 triangleDir 가 1 과 동일한 경우, 다음이 적용된다:

[0199] $$wIdx = (nCbW > nCbH) \ ? \ (Clip3(0, 8, (nCbH - 1 - x / nCbR - y) + 4)) \quad (8\text{-}843)$$
$$(Clip3(0, 8, (nCbW - 1 - x - y / nCbR) + 4))$$

[0200] - 그렇지 않고, cIdx 가 0 보다 크고 triangleDir 가 0 과 동일한 경우, 다음이 적용된다:

[0201] $$wIdx = (nCbW > nCbH) \ ? \ (Clip3(0, 4, (x / nCbR - y) + 2)) \quad (8\text{-}844)$$
$$: (Clip3(0, 4, (x - y / nCbR) + 2))$$

[0202] - 그렇지 않은 경우 (cIdx 가 0 보다 크고 triangleDir 가 1 과 동일한 경우), 다음이 적용된다:

[0203] $$wIdx = (nCbW > nCbH) \ ? \ (Clip3(0, 4, (nCbH - 1 - x / nCbR - y) + 2)) \quad (8\text{-}845)$$
$$(Clip3(0, 4, (nCbW - 1 - x - y / nCbR) + 2))$$

[0204] - 예측 샘플의 가중치를 지정하는 변수 wValue 는 다음과 같이 wIdx 및 cIdx를 사용하여 도출된다:

[0205] $$wValue = (cIdx == 0) \ ? \ Clip3(0, 8, wIdx) \ : \ Clip3(0, 8, wIdx * 2) \quad (8\text{-}846)$$

[0206] - 예측 샘플 값은 다음과 같이 도출된다:

$$pbSamples[x][y] = Clip3(0, (1 << bitDepth) - 1, (predSamplesLA[x][y] * wValue + (8-847)$$

$$predSamplesLB[x][y] * (8 - wValue) + offset1) >> shift1)$$

[0207]

[0208] 위의 예에서, cIdx > 0 은 크로마를 의미한다. 추가적으로, wValue = clip3(0,8, wIdx*2) 는 크로마 성분을 위한 것이다. 그래서, wIdx = 0, 1, 2, 3, 4 일 때, 각각, wValue = 8, 6, 4, 2, 0. 더욱이, blended_pixel = (wValue * predLA + (8-mValue)*predLB)>>shift. 그래서, 크로마에 사용되는 블렌딩 가중치는 0, 2, 4, 6, 8 (예를 들어, {2, 4, 6}) 이다. 이 예에서, 가중치 0 또는 8 은 두 픽셀을 블렌딩하는 대신 하나의 후보 픽셀을 복사하는 것을 나타낼 수도 있다.

[0209] 일부 예에서, 인트라 예측을 위해, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 YUV 4:4:4 포맷의 크로마 성분들에 대한 인트라 참조 샘플 평활화의 거동을 YUV 4:2:0 포맷의 크로마 성분에 대한 거동에 정렬하도록 구성될 수도 있다. 현재 VVC 드래프트에서, 인트라 참조 샘플 평활화는 YUV4:2:0 크로마 성분에 대해 비활성화된다. 일부 예에 따르면, 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 YUV 4:4:4 크로마 성분을 위한 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다.

[0210] 다르게 말하면, 예를 들어, 비디오 코더 (예컨대, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 화상의 영역에 대한 필터링되지 않은 참조 샘플들을 획득하도록 구성될 수도 있다. 이 예에서, 비디오 코더는 YUV 4:2:0 포맷의 크로마 샘플 그리고 YUV 4:4:4 포맷의 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화하도록 구성될 수도 있다. 이 예에서, 비디오 코더는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다.

[0211] 도 8 는 현재의 블록을 인코딩하는 예시적인 방법을 예시하는 흐름도이다. 현재 블록은 현재 CU 를 포함할 수도 있다. 비디오 인코더 (200) (도 1 및 도 3) 에 대해 설명되었지만, 다른 디바이스들이 도 8과 유사한 방법을 수행하도록 구성될 수도 있다는 것이 이해되야 한다.

[0212] 이 예에서, 비디오 인코더 (200) 는 초기에 현재 블록을 예측한다 (750). 일부 예에서, "현재 블록"은 본 명세서에서 간단히 "블록" 으로 지칭될 수도 있다. 예를 들어, 비디오 인코더 (200) 는 상이한 크로마 포맷을 위한 통일된 가중치 정보를 사용하여 현재 블록을 위한 예측 블록을 형성할 수도 있다. 일부 예에서, 비디오 인코더 (200) 는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다.

[0213] 그 후, 비디오 인코더 (200) 는 현재 블록을 위한 잔차 블록을 계산할 수도 있다 (752). 잔차 블록을 계산하기 위해, 비디오 인코더 (200) 는 원래의 코딩되지 않은 블록과 현재 블록을 위한 예측 블록 간의 차이를 계산하여, 잔차 블록을 생성할 수도 있다. 다음으로, 비디오 인코더 (200) 는 잔차 블록의 잔차 값들을 변환하고 변환 계수들을 양자화할 수도 있다 (754). 다음으로, 비디오 인코더 (200) 는 잔차 블록의 양자화된 변환 계수들을 스캔할 수도 있다 (756). 스캔 동안 또는 스캔 후에, 비디오 인코더 (200) 는 계수들을 엔트로피 인코딩할 수도 있다 (758). 예를 들어, 비디오 인코더 (200) 는 CAVLC 또는 CABAC 를 이용하여 계수들을 인코딩할 수도 있다. 비디오 인코더 (200) 는 블록의 엔트로피 코딩된 데이터를 출력할 수도 있다 (760).

[0214] 도 9 은 본 개시의 기법들에 따른 비디오 데이터의 현재 블록을 디코딩하기 위한 예시적인 방법을 예시하는 흐름도이다. 현재 블록은 현재 CU 를 포함할 수도 있다. 비디오 디코더 (300) (도 1 및 도 4) 에 관하여 설명되었지만, 다른 디바이스들이 도 9과 유사한 방법을 수행하도록 구성될 수도 있다는 것이 이해되야 한다.

[0215] 비디오 디코더 (300) 는 현재 블록에 대응하는 잔차 블록의 계수들을 위한 엔트로피 코딩된 예측 정보 및 엔트로피 코딩된 데이터와 같은, 현재 블록을 위한 엔트로피 코딩된 데이터를 수신할 수도 있다 (870). 비디오

디코더 (300) 는 엔트로피 코딩된 데이터를 엔트로피 디코딩하여 현재 블록을 위한 예측 정보를 결정하고 잔차 블록의 계수들을 재생성할 수도 있다 (872).

[0216] 비디오 디코더 (300) 는 현재 블록을 예측할 수도 있다 (874). 예를 들어, 비디오 디코더 (300) 는 현재 블록을 위한 예측 블록을 계산하기 위해, 예를 들어 현재 블록을 위한 예측 정보에 의해 표시된 바와 같이 인트라 또는 인터 예측 모드를 사용하여, 상이한 크로마 포맷에 대해 통일된 가중치 정보를 사용하여 현재 블록을 예측할 수도 있다. 일부 예에서, 비디오 디코더 (300) 는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성하도록 구성될 수도 있다.

[0217] 비디오 디코더 (300) 는 그 다음, 재생성된 계수들을 역 스캔하여 (876), 양자화된 변환 계수들의 블록을 생성할 수도 있다. 그 후에, 비디오 디코더 (300) 는 계수들을 역 양자화 및 역 변환하여 잔차 블록을 생성할 수도 있다 (878). 비디오 디코더 (300) 는 예측 블록과 잔차 블록을 조합함으로써 궁극적으로 현재 블록을 디코딩할 수도 있다 (880).

[0218] 도 10 은 본 개시의 기법들에 따른, YUV 4:2:0 를 위한 가중치 세트를 사용한 비디오 데이터 블록을 위한 크로마 성분들의 픽셀 블렌딩을 위한 예시적인 프로세스를 예시하는 흐름도이다. 비디오 인코더 (200) (도 1 및 도 3) 및 비디오 디코더 (300) (도 1 및 도 4) 에 대해 설명되었지만, 다른 디바이스들이 도 10과 유사한 방법을 수행하도록 구성될 수도 있다는 것이 이해되야 한다.

[0219] 본 개시의 기법들에 따르면, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의 비디오 데이터 블록의 크로마 성분을 위한 코딩 유닛을 생성하도록 구성될 수도 있다 (982). 비디오 코더는 삼각형 예측 유닛 모드의 활성화에 기초하여 크로마 성분을 위한 코딩 유닛을 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션으로 분할할 수도 있다 (984).

[0220] 비디오 코더는 하나 이상의 프로세서가 YUV 4:2:0 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 그리고 하나 이상의 프로세서가 YUV 4:4:4 포맷의 크로마 성분을 위한 코딩 유닛을 생성할 때 크로마 성분을 위한 예측 블록을 생성하기 위해 YUV 4:2:0 포맷을 위한 가중치 세트를 사용하여 픽셀 블렌딩을 적용할 수도 있다 (986).

일부 예에서, 픽셀 블렌딩을 적용하기 위해, 비디오 코더는, 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 모션 정보에 기초하여 제 1 삼각형 형상의 파티션 및 제 2 삼각형 형상의 파티션의 콜로케이트된 모션 보상된 픽셀의, YUV 4:2:0 포맷을 위한 가중치 세트를 사용한, 가중 평균을 결정할 수도 있다.

[0221] 비디오 코더가 비디오 디코더 (예를 들어, 비디오 디코더 (300)) 를 포함하는 예에서, 비디오 디코더는 비디오 데이터의 블록을 위한 잔차 블록을 디코딩하고 크로마 성분을 위한 예측된 블록과 잔차 블록을 조합하여 비디오 데이터의 블록을 디코딩할 수도 있다. 비디오 코더가 비디오 인코더를 포함하는 예에서, 비디오 인코더는 비디오 데이터의 블록과 예측된 블록 사이의 차이에 기초하여 비디오 데이터의 블록을 위한 잔차 블록을 생성하고 잔차 블록을 인코딩할 수도 있다.

[0222] 도 11은 인트라 예측을 이용하여 예측 블록의 크로마 샘플들을 생성하기 위한 예시적인 프로세스를 예시하는 흐름도이다. 비디오 인코더 (300) (도 1 및 도 3) 및 비디오 디코더 (300) (도 1 및 도 4) 에 대해 설명되었지만, 다른 디바이스들이 도 11과 유사한 방법을 수행하도록 구성될 수도 있다는 것이 이해되야 한다.

[0223] 개시의 기법에 따르면, 비디오 코더 (예를 들어, 비디오 인코더 (200) 또는 비디오 디코더 (300)) 는 화상의 영역을 위한 필터링되지 않은 참조 샘플을 획득할 수도 있다 (1090). 일부 예에서, 비디오 코더는 YUV 4:2:0 포맷 그리고 YUV 4:4:4 포맷의 크로마 샘플을 위해 필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를 비활성화할 수도 있다. 비디오 코더는 YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고 YUV 4:4:4 포맷의 크로마 성분을 생성할 때 필터링되지 않은 참조 샘플에 기초하여 화상의 블록을 위한 예측 블록의 크로마 샘플을, 인트라 예측을 이용하여, 생성할 수도 있다 (1092).

[0224] 비디오 코더가 비디오 디코더 (예를 들어, 비디오 디코더 (300)) 를 포함하는 예에서, 비디오 디코더는 비디오 데이터의 블록을 위한 잔차 블록을 디코딩하고 크로마 성분을 위한 예측된 블록과 잔차 블록을 조합하여 비디오 데이터의 블록을 디코딩할 수도 있다. 비디오 코더가 비디오 인코더를 포함하는 예에서, 비디오 인코더는 비디오 데이터의 블록과 예측된 블록 사이의 차이에 기초하여 비디오 데이터의 블록을 위한 잔차 블록을 생성하고 잔차 블록을 인코딩할 수도 있다.

[0225] 본 개시의 예시적인 예들은 다음을 포함한다:

[0226] 예 1 : 비디오 데이터를 프로세싱하는 방법으로서, 제 1 크로마 포맷을 위한 가중치 정보를 결정하는 단계; 제 1 크로마 포맷을 위한 가중치 정보에 대응하도록 제 1 크로마 포맷과 상이한 제 2 크로마 포맷을 위한 가중치 정보를 결정하는 단계; 및 제 1 크로마 포맷을 위한 가중치 정보에 기초하여 예측 정보를 생성하는 단계를 포함하는, 비디오 데이터를 프로세싱하는 방법.

[0227] 예 2: 예 1에 있어서, 포함하는, 모드가 삼각형 예측 유닛 (PU) 모드인 것을 결정하는 단계를 포함하고, 제 2 크로마 포맷을 위한 가중치 정보를 결정하는 단계는 모드가 삼각형 PU 모드라는 결정에 응답하는, 비디오 데이터를 프로세싱하는 방법.

[0228] 예 3. 예 1 내지 예 2 의 임의의 조합에 있어서, 제 1 크로마 포맷은 YUV 4:2:0 이고, 제 2 크로마 포맷은 YUV 4:4:4 인, 비디오 데이터를 프로세싱하는 방법.

[0229] 예 4. 예 3에 있어서, 제 1 크로마 포맷을 위한 가중치 정보는 1) {1, 4, 7}2) {2, 4, 6} 인, 비디오 데이터를 프로세싱하는 방법.

[0230] 예 5. 예 1에 있어서, 인트라 예측이 적용되는 것을 결정하는 것에 응답하여, YUV 4:4:4 포맷의 크로마 성분에 대한 인트라 참조 샘플 평활화의 거동을 YUV 4:2:0 포맷에 대한 그것에 정렬하는 단계를 포함하는, 비디오 데이터를 프로세싱하는 방법.

[0231] 예 6. 예 5에 있어서, 정렬하는 단계는 YUV4:2:0 크로마 성분을 위한 인트라 참조 샘플 평활화를 비활성화하는 단계; 및 YUV4:4:4 크로마 성분을 위한 인트라 참조 샘플 평활화를 비활성화하는 단계를 포함하는, 비디오 데이터를 프로세싱하는 방법.

[0232] 예 7. 예 1 내지 예 6 중 어느 예에 있어서, 프로세싱은 디코딩을 포함하는, 비디오 데이터를 프로세싱하는 방법.

[0233] 예 8. 예 1 내지 예 6 중 어느 예에 있어서, 프로세싱은 인코딩을 포함하는, 비디오 데이터를 프로세싱하는 방법.

[0234] 예 9. 비디오 데이터를 코딩하기 위한 디바이스로서, 예 1 내지 예 6 중 어느 예의 방법을 수행하기 위한 하나 이상의 수단을 포함하는, 디바이스.

[0235] 예 10. 예 9에 있어서, 하나 이상의 수단은 회로에서 구현되는 하나 이상의 프로세서를 포함하는, 디바이스.

[0236] 예 11. 예 9 및 예 10 중 어느 예에 있어서, 비디오 데이터를 저장하기 위한 메모리를 더 포함하는, 디바이스.

[0237] 예 12. 예 9 내지 예 11 중 어느 예에 있어서, 디코딩된 비디오 데이터를 표시하도록 구성된 디스플레이를 더 포함하는, 디바이스.

[0238] 예 13. 예 9 내지 예 12 중 어느 예에 있어서, 상기 디바이스는 카메라, 컴퓨터, 모바일 디바이스, 브로드캐스트 수신기 디바이스 또는 셋탑 박스 중 하나 이상을 포함하는, 디바이스.

[0239] 예 14. 예 9 내지 예 13 중 어느 예에 있어서, 상기 디바이스는 비디오 디코더를 포함하는, 디바이스.

[0240] 예 15. 예 9 내지 예 14 중 어느 예에 있어서, 상기 디바이스는 비디오 인코더를 포함하는, 디바이스.

[0241] 예 16. 명령들이 저장된 컴퓨터 판독가능 저장 매체로서, 상기 명령들은, 실행될 경우, 하나 이상의 프로세서들로 하여금 예 1 내지 예 6 중 어느 예의 방법을 수행하게 하는, 컴퓨터 판독가능 저장 매체.

[0242] 예에 따라, 여기에 기재된 기술 중 어느 것의 특정 행위들 또는 이벤트들이 상이한 시퀀스에서 수행될 수 있거나, 추가될 수 있거나, 병합될 수 있거나, 또는 전부 생략될 수 있다 (예를 들어, 모든 설명된 행위들 또는 이벤트들이 기술의 실시를 위해 필요한 것은 아니다) 는 것이 인식되야 한다. 또한, 특정 예들에서, 행위들 또는 이벤트들은, 예를 들어, 순차적으로 보다는 멀티 스레드 프로세싱, 인터럽트 프로세싱 또는 다수의 프로세서들을 통해, 동시적으로 수행될 수도 있다.

[0243] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되면, 그 기능들은 컴퓨터 판독가능 매체 상의 하나 이상의 명령 또는 코드로서 저장되거나 송신될 수도 있고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체는, 예를 들면, 통신 프로토콜에 따라, 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체 또는 데이터 저장 매체와 같은 유형의 매체에 대응하는 컴퓨터 판독가능

저장 매체를 포함할 수도 있다. 이런 방식으로, 컴퓨터 판독가능 매체는 일반적으로, (1) 비일시적인 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는, 본 개시에서 설명된 기술들의 구현을 위해 명령들, 코드 및/또는 데이터 구조들을 취출하기 위하여 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수 있다.

[0244] 비한정적 예로서, 이러한 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장, 자기 디스크 저장 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 명령 또는 데이터 구조의 형태로 원하는 프로그램 코드를 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터 판독가능 매체로 적절히 칭해진다. 예를 들어, 명령들이 동축 케이블, 광섬유 케이블, 연선 (twisted pair), 디지털 가입자 라인 (DSL), 또는 적외선, 전파 (radio), 및 마이크로파와 같은 무선 기술을 사용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되면, 그 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 전파, 및 마이크로파와 같은 무선 기술은 매체의 정의 내에 포함된다. 하지만, 컴퓨터 판독가능 저장 매체 및 데이터 저장 매체는 접속, 캐리어 파, 신호 또는 다른 일시적 매체를 포함하는 것이 아니라, 대신에 비일시적, 유형의 저장 매체에 관한 것이 이해되야 한다. 여기에 사용된, 디스크 (disk) 및 디스크 (disc) 는 콤팩트 디스크 (CD), 레이저 디스크, 광 디스크, DVD (digital versatile disc), 플로피 디스크 및 블루레이 디스크를 포함하며, 여기서, 디스크 (disk) 는 보통 데이터를 자기적으로 재생하지만, 디스크 (disc) 는 레이저를 이용하여 광학적으로 데이터를 재생한다. 또한, 상기의 조합은 컴퓨터 판독가능 매체의 범위 내에 포함되어야 한다.

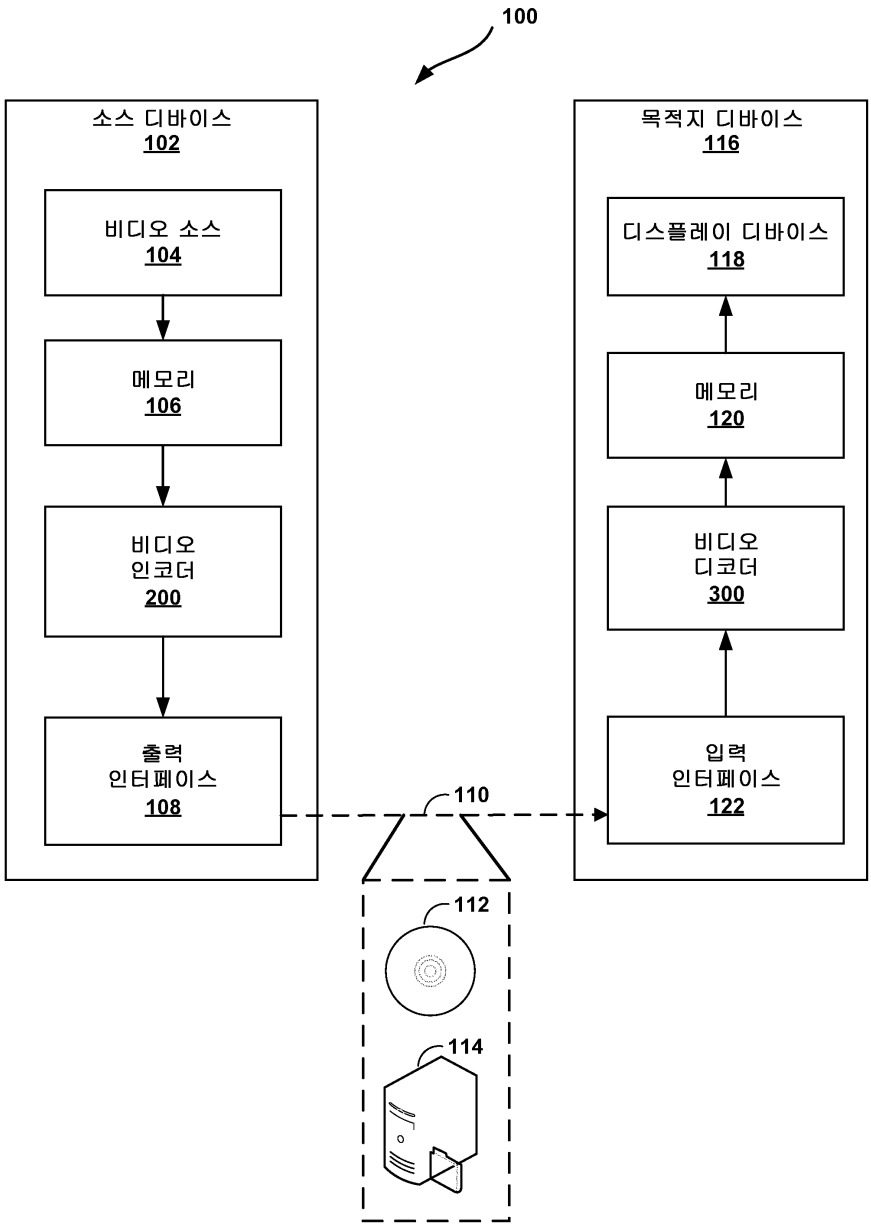
[0245] 명령들은 하나 이상의 프로세서, 예컨대 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 또는 다른 동등한 집적 또는 이산 로직 회로에 의해 실행될 수도 있다. 따라서, 본 명세서에서 사용된 바와 같이 용어들 "프로세서" 및 “프로세싱 회로”는 전술한 구조들 중 임의의 것 또는 본 명세서 설명된 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 추가로, 일부 양태에서, 여기서 설명된 기능성은 인코딩 및 디코딩을 위해 구성된 전용 하드웨어 모듈 및/또는 소프트웨어 모듈 내에 제공되거나 또는 결합된 코덱에 포함될 수도 있다. 또한, 그 기법들은 하나 이상의 회로 또는 로직 요소들에서 완전히 구현될 수 있다.

[0246] 이 개시의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC 들의 세트 (예를 들어, 칩셋) 를 포함하는 광범위한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들, 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하는 것은 아니다. 오히려, 위에 설명된 바와 같이, 다양한 유닛들은 코덱 하드웨어 유닛에서 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에 설명된 바와 같은 하나 이상의 프로세서들을 포함하는, 상호운용가능한 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

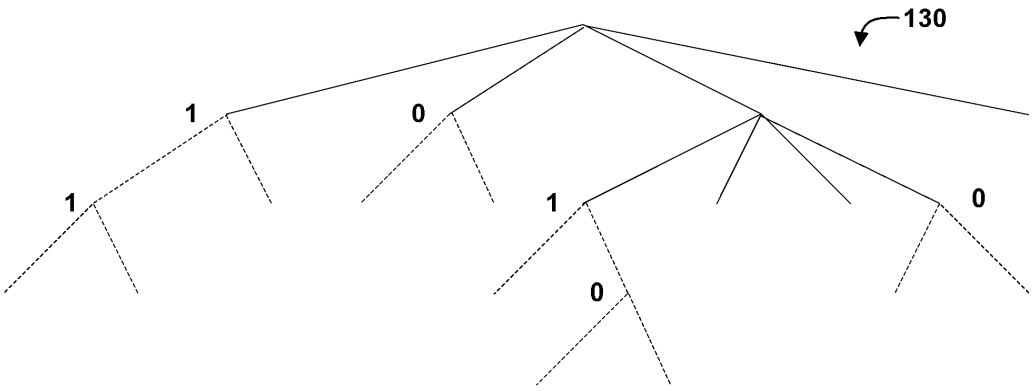
[0247] 다양한 예들이 설명되었다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

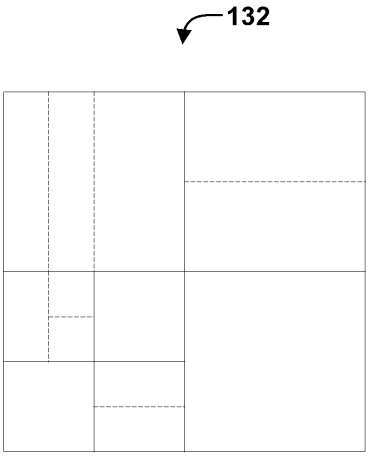
도면1


100
소스 디바이스
102
비디오 소스
104
메모리
106
비디오
인코더
200
출력
인터페이스
108
110
112
114
목적지 디바이스
116
디스플레이 디바이스
118
메모리
120
비디오
디코더
300
입력
인터페이스
122

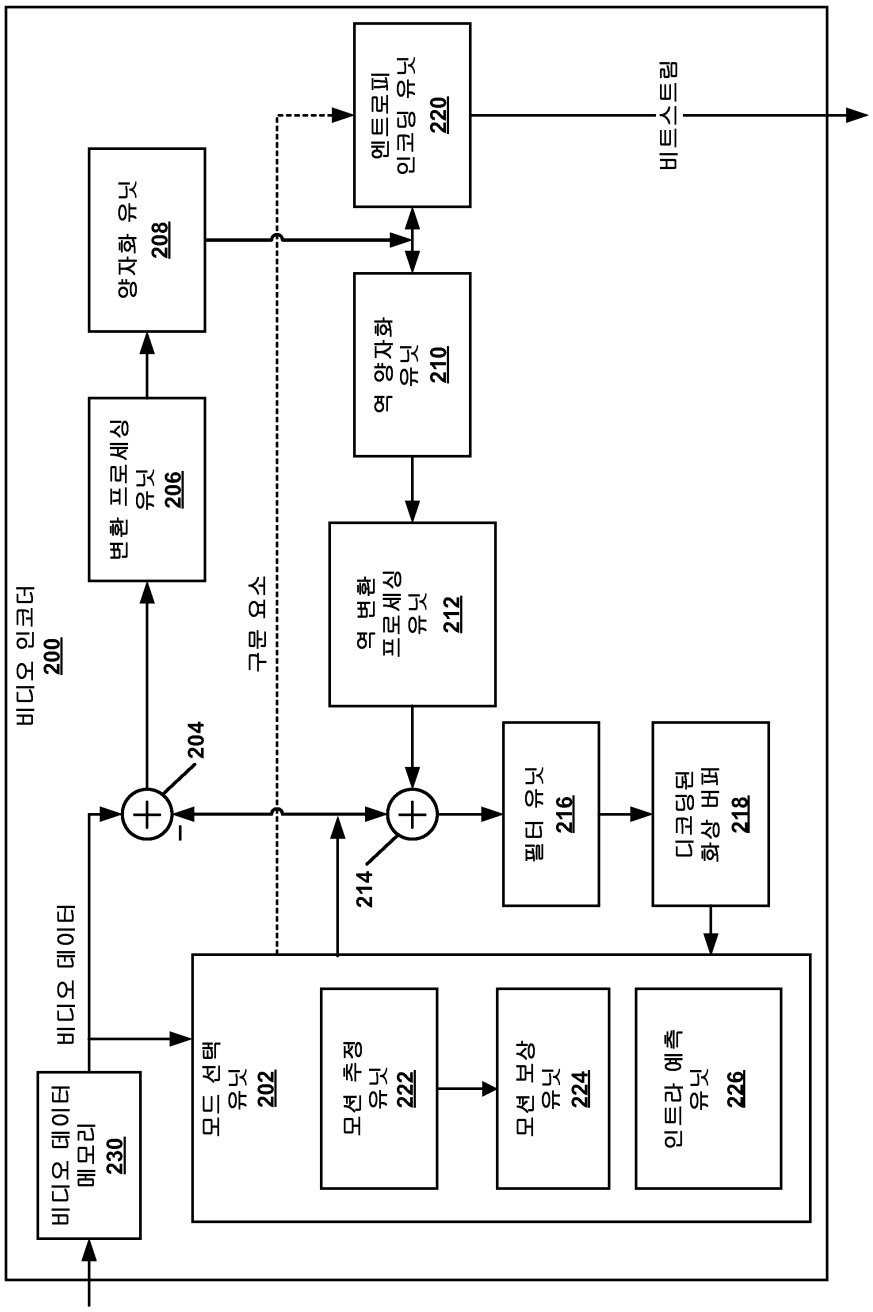
도면2a



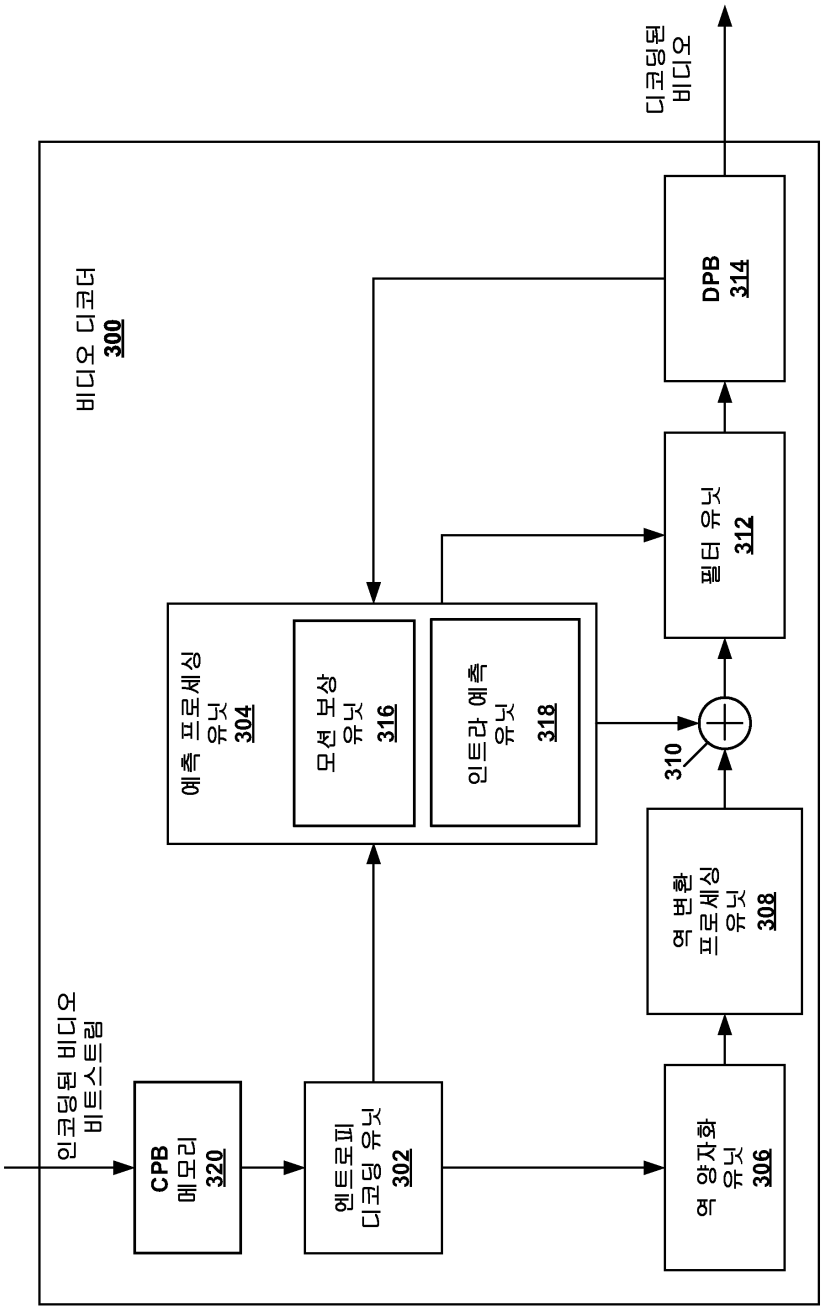
도면2b



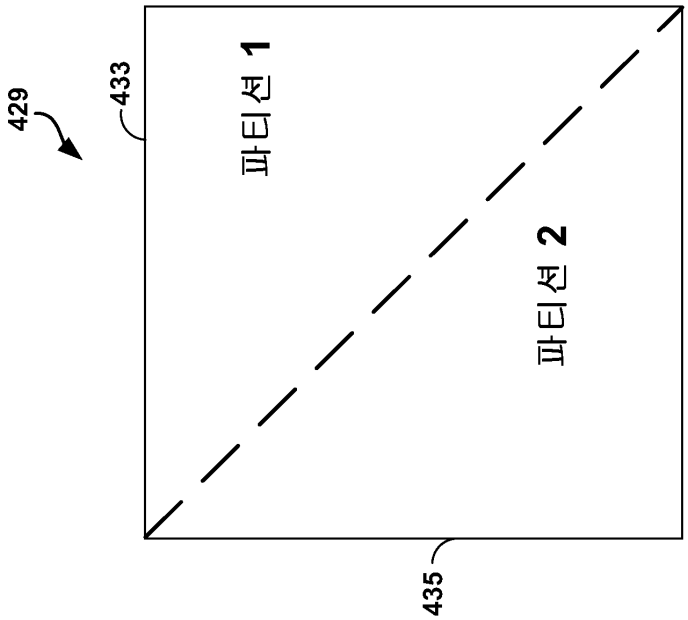
도면3


비디오 인코더
200
비디오 데이터
비디오 데이터 메모리
230
모드 선택 유닛
202
모션 추정 유닛
222
모션 보상 유닛
224
인트라 예측 유닛
226
204
구문 요소
변환 프로세싱 유닛
206
양자화 유닛
208
역 양자화 유닛
210
역 변환 프로세싱 유닛
212
214
필터 유닛
216
디코딩된 화상 버퍼
218
엔트로피 인코딩 유닛
220
비트스트림

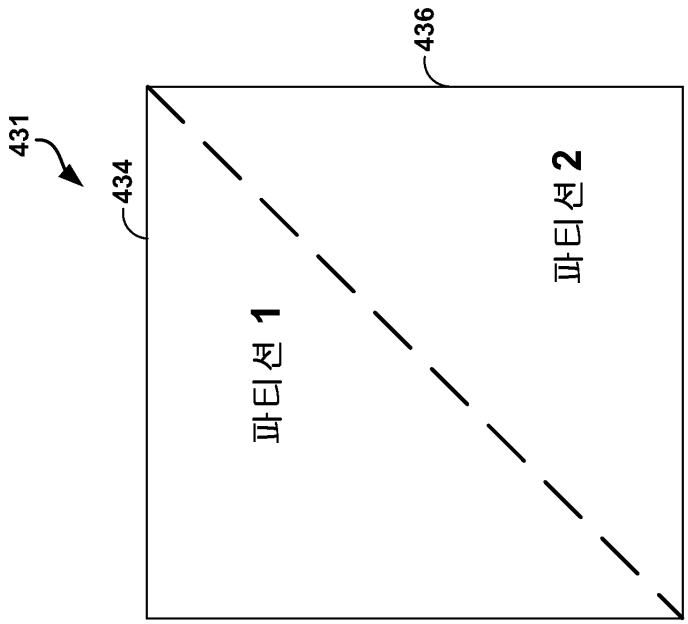
도면4


디코딩된
비디오
비디오 디코더
300
DPB
314
필터 유닛
312
예측 프로세싱
유닛
304
모션 보상
유닛
316
인트라 예측
유닛
318
310
역 변환
프로세싱
유닛
308
역 양자화
유닛
306
엔트로피
디코딩 유닛
302
CPB
메모리
320
인코딩된 비디오
비트스트림

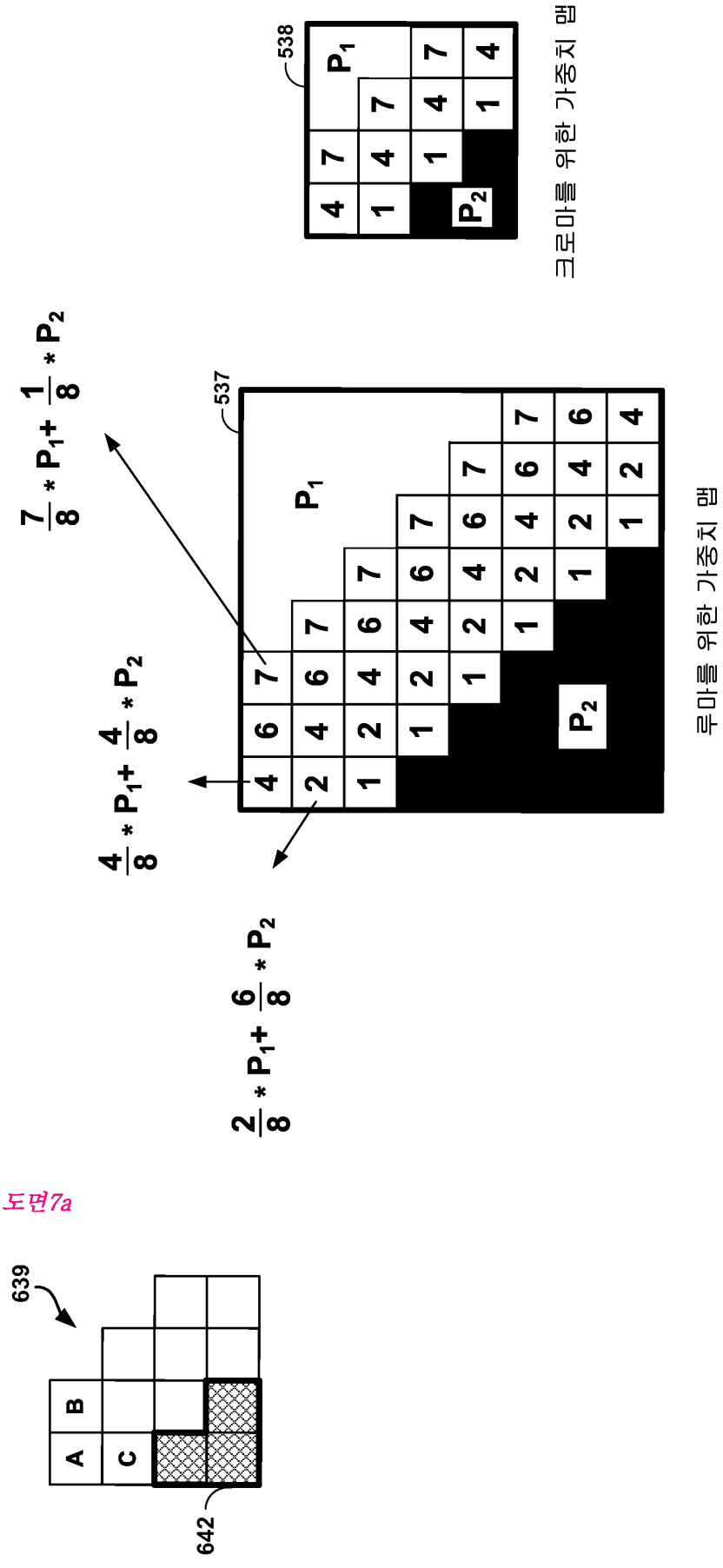
도면5a



도면5b



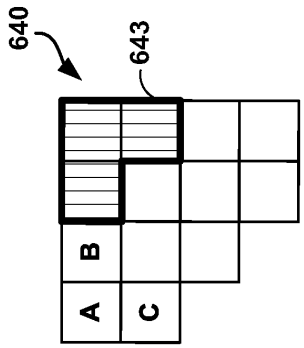
도면6


538
P1
P2
크로마를 위한 가중치 맵
537
P1
P2
$\frac{7}{8} * P_1 + \frac{1}{8} * P_2$
$\frac{4}{8} * P_1 + \frac{4}{8} * P_2$
$\frac{2}{8} * P_1 + \frac{6}{8} * P_2$
루마를 위한 가중치 맵

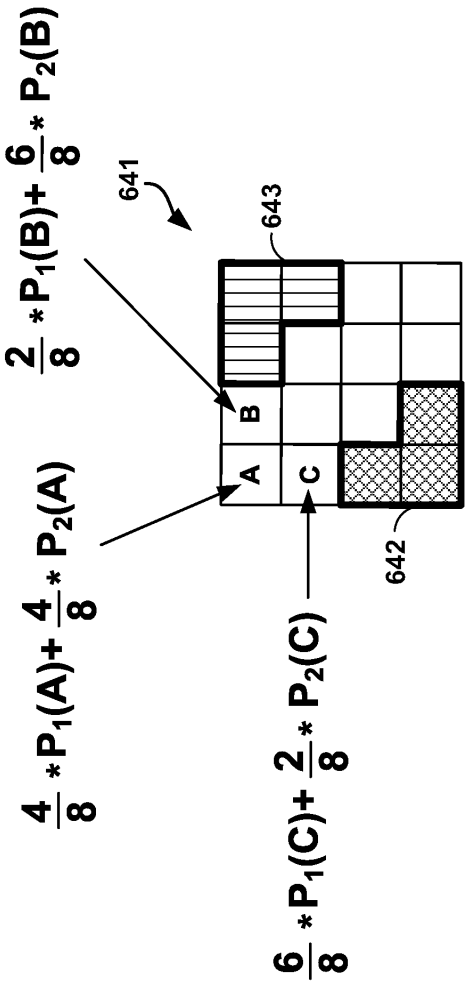
도면7a

639
A
B
C
642

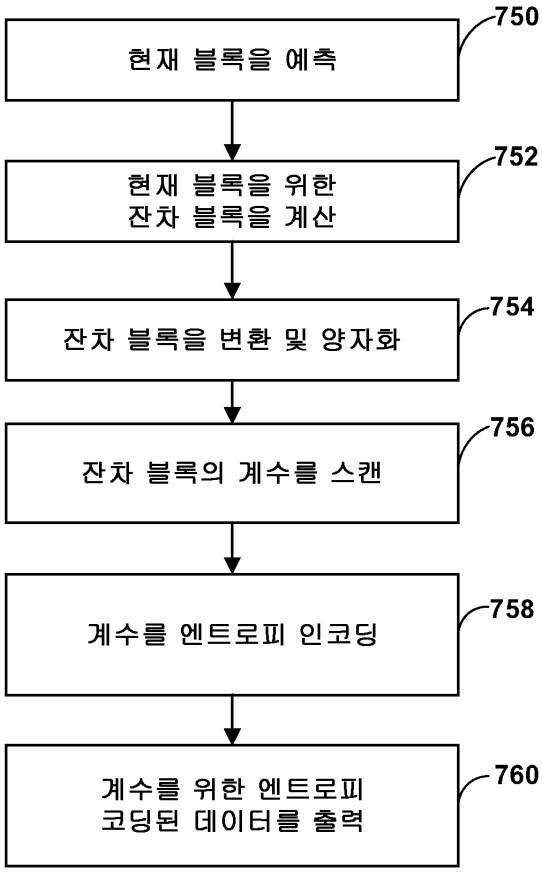
도면7b


640
643
B
A
C

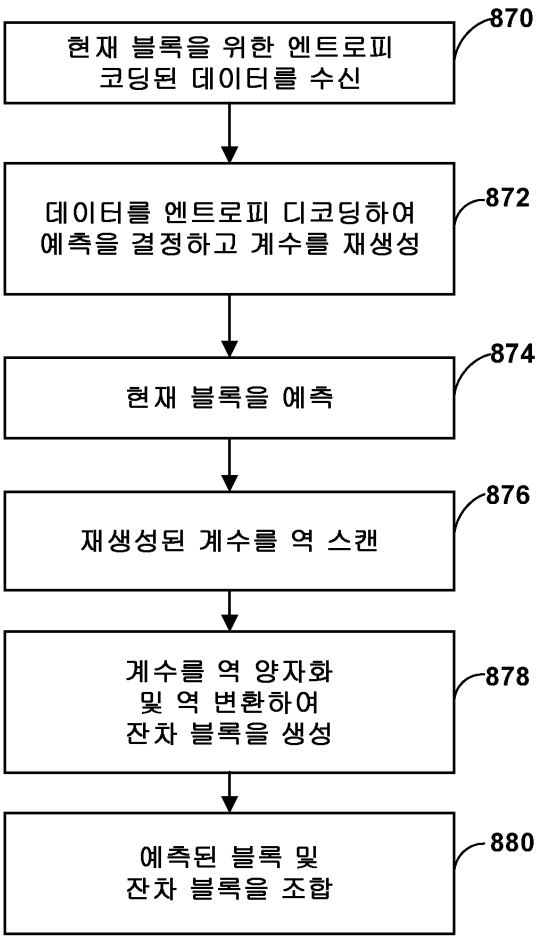
도면7c


$\frac{2}{8}$ *P1(B)+ $\frac{6}{8}$* P2(B)
641
643
B
A
C
642
$\frac{4}{8}$ *P1(A)+ $\frac{4}{8}$* P2(A)
$\frac{6}{8}$ *P1(C)+ $\frac{2}{8}$* P2(C)

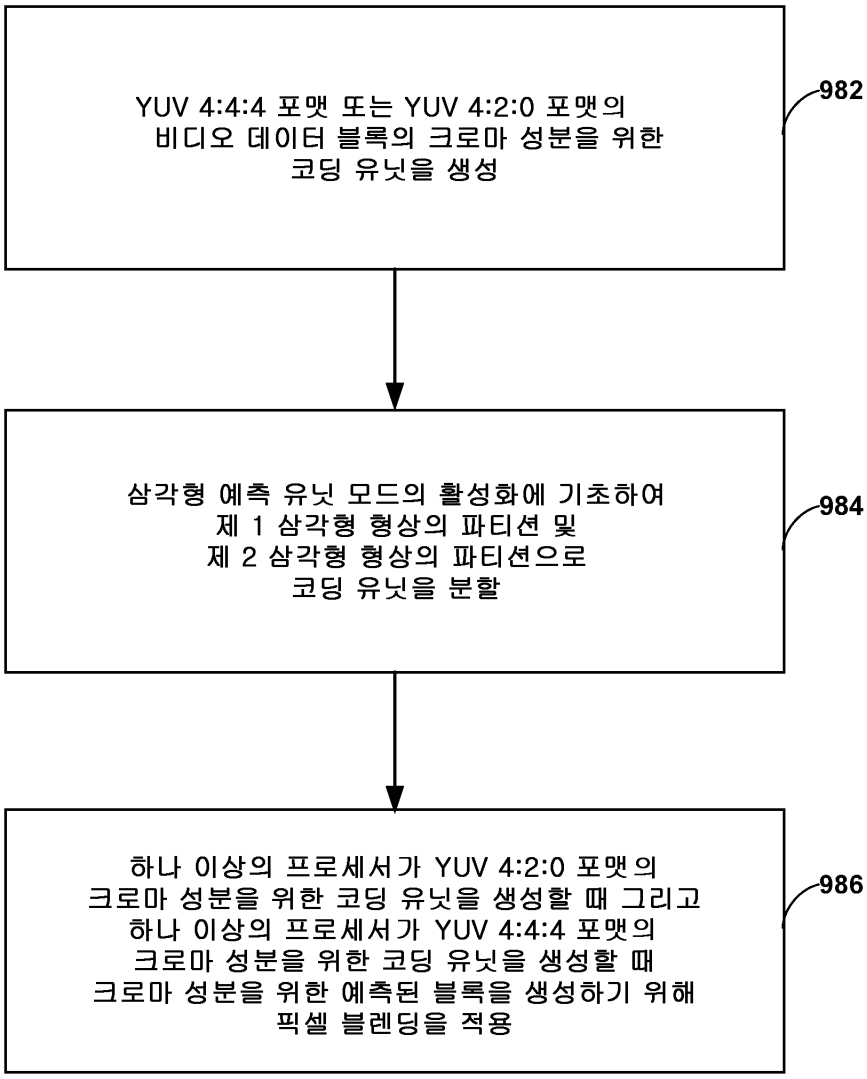
도면8


현재 블록을 예측
750
현재 블록을 위한
잔차 블록을 계산
752
잔차 블록을 변환 및 양자화
754
잔차 블록의 계수를 스캔
756
계수를 엔트로피 인코딩
758
계수를 위한 엔트로피
코딩된 데이터를 출력
760

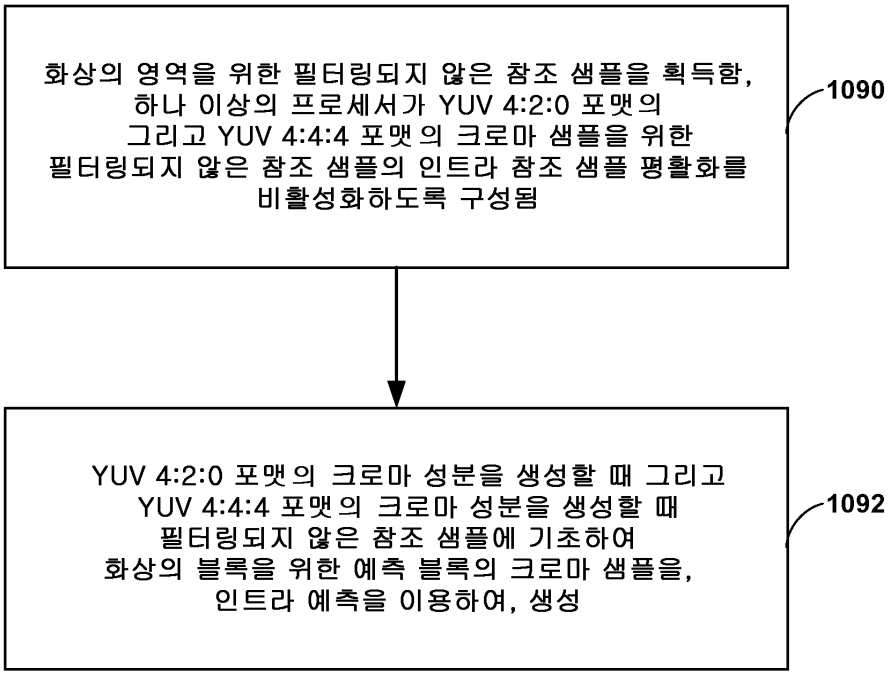
도면9


현재 블록을 위한 엔트로피 코딩된 데이터를 수신
870
데이터를 엔트로피 디코딩하여 예측을 결정하고 계수를 재생성
872
현재 블록을 예측
874
재생성된 계수를 역 스캔
876
계수를 역 양자화 및 역 변환하여 잔차 블록을 생성
878
예측된 블록 및 잔차 블록을 조합
880

도면10


YUV 4:4:4 포맷 또는 YUV 4:2:0 포맷의
비디오 데이터 블록의 크로마 성분을 위한
코딩 유닛을 생성
982
삼각형 예측 유닛 모드의 활성화에 기초하여
제 1 삼각형 형상의 파티션 및
제 2 삼각형 형상의 파티션으로
코딩 유닛을 분할
984
하나 이상의 프로세서가 YUV 4:2:0 포맷의
크로마 성분을 위한 코딩 유닛을 생성할 때 그리고
하나 이상의 프로세서가 YUV 4:4:4 포맷의
크로마 성분을 위한 코딩 유닛을 생성할 때
크로마 성분을 위한 예측된 블록을 생성하기 위해
픽셀 블렌딩을 적용
986

도면11


화상의 영역을 위한 필터링되지 않은 참조 샘플을 획득함,
하나 이상의 프로세서가 YUV 4:2:0 포맷의
그리고 YUV 4:4:4 포맷의 크로마 샘플을 위한
필터링되지 않은 참조 샘플의 인트라 참조 샘플 평활화를
비활성화하도록 구성됨
1090
YUV 4:2:0 포맷의 크로마 성분을 생성할 때 그리고
YUV 4:4:4 포맷의 크로마 성분을 생성할 때
필터링되지 않은 참조 샘플에 기초하여
화상의 블록을 위한 예측 블록의 크로마 샘플을,
인트라 예측을 이용하여, 생성
1092