(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0044754
(43) 공개일자 2022년04월11일

(51) 국제특허분류(Int. Cl.)
H04N 19/119 (2014.01) *H04N 19/107* (2014.01)
H04N 19/109 (2014.01) *H04N 19/122* (2014.01)
H04N 19/14 (2014.01) *H04N 19/176* (2014.01)
H04N 19/186 (2014.01) *H04N 19/513* (2014.01)
(52) CPC특허분류
H04N 19/119 (2015.01)
H04N 19/107 (2015.01)
(21) 출원번호 10-2022-7005449
(22) 출원일자(국제) 2020년07월22일
심사청구일자 없음
(85) 번역문제출일자 2022년02월17일
(86) 국제출원번호 PCT/US2020/043143
(87) 국제공개번호 WO 2021/030019
국제공개일자 2021년02월18일
(30) 우선권주장
62/887,039 2019년08월15일 미국(US)
62/903,970 2019년09월23일 미국(US)

(71) 출원인
알리바바 그룹 홀딩 리미티드
케이만군도, 그랜드 케이만, 조지 타운, 피.오. 박스 847, 원 캐피탈 플레이스, 플로어 4
(72) 발명자
랴오, 룰링
중국 저장성 311121 항저우 위항 디스트릭트 웨스트 옌 이 알디 엔오 969 비엘디지 3 5/에프 알리바바 그룹 리갈 디파트먼트
첸, 지에
중국 저장성 311121 항저우 위항 디스트릭트 웨스트 옌 이 알디 엔오 969 비엘디지 3 5/에프 알리바바 그룹 리갈 디파트먼트
(뒷면에 계속)
(74) 대리인
특허법인 광장리앤고

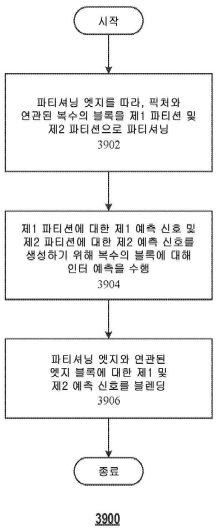
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 **비디오 코딩을 위한 블록 파티셔닝 방법**

(57) 요 약

본 개시는 비디오 콘텐츠를 처리하기 위한 시스템 및 방법을 제공한다. 본 개시는: 파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계; 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측을 수행하는 단계; 및 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계를 포함한다.

대 표 도 - 도39



(52) CPC특허분류
H04N 19/109 (2015.01)
H04N 19/122 (2015.01)
H04N 19/14 (2015.01)
H04N 19/176 (2015.01)
H04N 19/186 (2015.01)
H04N 19/513 (2015.01)
(72) 발명자
예, 얀
미국 캘리포니아 94402 샌 마테오 스위트 400 사우스 엘 카미노 리얼 400 알리바바 그룹 리갈 디파트먼트
루오, 지안콩
미국 캘리포니아 94402 샌 마테오 스위트 400 사우스 엘 카미노 리얼 400 알리바바 그룹 리갈 디파트먼트

명 세 서

청구범위

청구항 1

비디오 콘텐츠를 처리하기 위한 방법으로서,

파티셔닝 엣지(partitioning edge)를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계;

상기 제1 파티션에 대한 제1 예측 신호 및 상기 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 상기 복수의 블록에 대해 인터 예측(inter prediction)을 수행하는 단계; 및

상기 파티셔닝 엣지와 연관된 엣지 블록에 대한 상기 제1 예측 신호 및 상기 제2 예측 신호를 블렌딩하는 단계(blending)를 포함하는, 방법.

청구항 2

제1항에 있어서, 상기 파티셔닝 엣지를 따라 상기 복수의 블록을 파티셔닝하는 단계는:

상기 복수의 블록의 파티션 모드를 결정하는 단계; 및

상기 파티션 모드를 기초로 상기 복수의 블록을 파티셔닝하는 단계를 더 포함하는, 방법.

청구항 3

제2항에 있어서, 상기 복수의 블록은 제1 블록의 서브블록이고, 상기 복수의 블록에 대한 상기 파티션 모드를 결정하는 단계는:

제1 표시 신호에 따라, 상기 제1 블록이 서브블록 병합 모드를 사용하여 코딩되는지를 결정하는 단계;

상기 제1 블록이 상기 서브블록 병합 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 제2 표시 신호에 따라, 상기 제1 블록이 정규 모드 또는 움직임 벡터 차분을 갖는 병합 모드(merge mode with motion vector differences, MMVD) 중 하나를 사용하여 코딩되는지를 결정하는 단계;

상기 제1 블록이 상기 정규 모드 또는 MMVD 중 하나를 사용하여 코딩되지 않는다는 결정에 응답하여, 제3 표시 신호에 따라, 상기 제1 블록이 결합된 인터 및 인트라 예측(combined inter and intra prediction, CIIP) 모드를 사용하여 코딩되는지를 결정하는 단계; 및

상기 제1 블록이 상기 CIIP 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 상기 복수의 블록의 상기 파티션 모드가 삼각형 파티션 모드 또는 기하학적 파티션 모드 중 하나라고 결정하는 단계를 더 포함하는, 방법.

청구항 4

제2항에 있어서, 상기 파티셔닝 엣지를 따라 상기 복수의 블록을 파티셔닝하는 단계는:

파티션 모드 인덱스, 각도 인덱스 또는 거리 인덱스에 따라 타겟 파티셔닝 방식을 결정하는 단계; 및

상기 타겟 파티셔닝 방식에 대응하는 상기 파티셔닝 엣지를 결정하는 단계를 더 포함하는, 방법.

청구항 5

제4항에 있어서,

마스크의 세트를 생성하는 단계;

상기 제1 블록의 크기, 상기 타겟 파티셔닝 방식의 각도 값, 상기 타겟 파티셔닝 방식의 거리 값을 기초로 제1 오프셋 및 제2 오프셋을 결정하는 단계; 및

상기 마스크의 세트를 기초로, 상기 제1 오프셋 및 상기 제2 오프셋을 사용하여 복수의 블렌딩 가중치를 생성하

는 단계를 더 포함하는, 방법.

청구항 6

제5항에 있어서,

상기 마스크의 세트의 수는 상기 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정되고, 상기 타겟 파티셔닝 방식의 상기 각도 값은 상기 파티션 모드 인덱스 및 상기 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정되며, 상기 타겟 파티셔닝 방식의 상기 거리 값은 상기 파티션 모드 인덱스 및 상기 기하학적 파티션 모드에서 지원되는 상기 각도의 수를 기초로 결정되는, 방법.

청구항 7

제5항에 있어서, 상기 파티션 모드 인덱스, 상기 각도 인덱스 또는 상기 거리 인덱스에 따라 상기 타겟 파티셔닝 방식을 결정하는 단계는:

룩업 테이블에 따라 상기 타겟 파티셔닝 방식을 결정하는 단계 - 상기 룩업 테이블은 복수의 파티셔닝 방식과 연관된 복수의 파티션 모드 인덱스, 복수의 각도 인덱스 및 복수의 거리 인덱스를 포함함 -를 더 포함하는, 방법.

청구항 8

제7항에 있어서,

상기 마스크의 세트의 수는 상기 룩업 테이블의 각도 인덱스의 수를 기초로 결정되고, 상기 타겟 파티셔닝 방식의 상기 각도 값은 상기 타겟 파티셔닝 방식에 대응하는 상기 룩업 테이블의 각도 인덱스를 기초로 결정되고, 및 상기 타겟 파티셔닝 방식의 상기 거리 값은 상기 타겟 파티셔닝 방식에 대응하는 상기 룩업 테이블의 거리 인덱스를 기초로 결정되는, 방법.

청구항 9

제7항에 있어서, 상기 룩업 테이블은 상기 복수의 파티셔닝 방식 중에서, 제1 파티션 모드 인덱스와 연관된 제1 파티셔닝 방식 및 제2 파티션 모드 인덱스와 연관된 제2 파티셔닝 방식을 포함하고, 상기 제1 파티셔닝 방식 및 제2 파티셔닝 방식은 상기 삼각형 파티션 모드에 대응하는, 방법.

청구항 10

제9항에 있어서, 상기 제1 파티션 모드 인덱스는 10과 동일하고, 상기 제1 파티션 모드 인덱스와 연관된 상기 제1 파티셔닝 방식은 상기 블록을 상기 블록의 좌측-상단 코너로부터 우측-하단 코너로 분할하는 것에 대응하고, 상기 제2 파티션 모드 인덱스는 24와 동일하고, 상기 제2 파티션 모드 인덱스와 연관된 상기 제2 파티셔닝 방식은 상기 블록을 상기 블록의 우측-상단 코너로부터 좌측-하단 코너로 분할하는 것에 대응하는, 방법.

청구항 11

제5항에 있어서, 상기 복수의 블렌딩 가중치는 상기 엣지 블록에 대한 복수의 루마 가중치(luma weights) 및 상기 엣지 블록에 대한 복수의 크로마 가중치(chroma weights)를 포함하고, 상기 파티셔닝 엣지와 연관된 상기 엣지 블록에 대한 상기 제1 예측 신호 및 상기 제2 예측 신호를 블렌딩하는 단계는:

상기 엣지 서브블록에 대한 상기 복수의 루마 가중치에 따라 상기 엣지 블록의 루마 값을 결정하는 단계; 및

상기 엣지 서브블록에 대한 상기 복수의 크로마 가중치에 따라 상기 엣지 서브블록의 크로마 값을 결정하는 단계를 더 포함하는, 방법.

청구항 12

제11항에 있어서, 상기 복수의 크로마 가중치 중에서, 크로마 가중치는 상기 크로마 가중치에 대응하는 2x2 블록의 좌측-상단 코너에 대한 루마 가중치를 기초로 결정되는, 방법.

청구항 13

제4항에 있어서,

상기 제1 블록의 크기, 상기 타겟 파티셔닝 방식의 각도 값, 상기 타겟 파티셔닝 방식의 거리 값을 기초로 제1 오프셋 및 제2 오프셋을 결정하는 단계; 및

상기 제1 오프셋 및 상기 제2 오프셋을 사용하여, 상기 제1 블록에서 루마 샘플에 대한 복수의 블렌딩 가중치를 생성하는 단계를 더 포함하는, 방법.

청구항 14

제13항에 있어서, 상기 제1 오프셋 및 상기 제2 오프셋은 아래의 식을 사용하여 결정되고:

제1 오프셋

$$= \begin{cases} (-W) \gg 1 & A \,\%\, 16 == 8 \text{ 또는 } (A \,\%\, 16 != 0 \text{ 및 } H \geq W) \\ ((-W) \gg 1) + A < 16 \,?\, (D \times W) \gg 3 : -((D \times W) \gg 3) & \textit{그렇지 않은 경우} \end{cases}$$

제2 오프셋

$$= \begin{cases} ((-H) \gg 1) + A < 16 \,?\, (D \times H) \gg 3 : -((D \times H) \gg 3) & A \,\%\, 16 == 8 \text{ 또는 } (A \,\%\, 16 != 0 \text{ 및 } H \geq W) \\ (-H) \gg 1 & \textit{그렇지 않은 경우} \end{cases}$$

여기서 "W"는 상기 제1 블록의 폭을 나타내고, "H"는 상기 제1 블록의 높이를 나타내고, "A"는 상기 타겟팅 파티셔닝 방식의 상기 각도 값을 나타내고, "D"는 상기 타겟팅 파티셔닝 방식의 상기 거리 값을 나타내는, 방법.

청구항 15

제3항에 있어서, 상기 제3 표시 신호에 따라, 상기 제1 블록이 상기 결합된 인터 및 인트라 예측(CIIP) 모드를 사용하여 코딩되는지를 결정하는 단계 이전에, 상기 방법은:

상기 제1 블록의 크기가 주어진 조건을 충족하는지를 결정하는 단계; 및

상기 제1 블록의 상기 크기가 상기 주어진 조건을 충족한다는 결정에 응답하여, 상기 제3 표시 신호를 생성하는 단계, 또는

상기 블록의 상기 크기가 상기 주어진 조건을 충족하지 못한다는 결정에 응답하여, 상기 블록이 상기 CIIP 모드를 사용하여 코딩된다고 결정하는 단계를 더 포함하는, 방법.

청구항 16

제15항에 있어서, 상기 주어진 조건은:

상기 제1 블록의 폭 및 높이가 각각 8 이상임; 및

상기 폭과 상기 높이 사이의 더 큰 값과 상기 폭과 상기 높이 사이의 더 작은 값의 비율이 4 이하임을 포함하는, 방법.

청구항 17

비디오 콘텐츠를 처리하기 위한 시스템으로서,

명령어의 세트를 저장하는 메모리; 및

적어도 하나의 프로세서를 포함하고, 상기 적어도 하나의 프로세서는 상기 시스템이:

파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 것;

상기 제1 파티션에 대한 제1 예측 신호 및 상기 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 상기 복수의 블록에 대해 인터 예측을 수행하는 것; 및

상기 파티셔닝 엣지와 연관된 엣지 블록에 대한 상기 제1 예측 신호 및 상기 제2 예측 신호를 블렌딩하는 것을 수행하게 하기 위해 상기 명령어의 세트를 실행하도록 구성되는, 시스템.

청구항 18

제17항에 있어서, 상기 파티셔닝 엣지를 따라 상기 복수의 블록을 파티셔닝하는 것에서, 상기 적어도 하나의 프로세서는 상기 시스템이:

상기 복수의 블록의 파티션 모드를 결정하는 것; 및

상기 파티션 모드를 기초로 상기 복수의 블록을 파티셔닝하는 것을 더 수행하게 하기 위해 상기 명령어의 세트를 실행하도록 구성되는, 시스템.

청구항 19

제18항에 있어서, 상기 복수의 블록은 제1 블록의 서브블록이고, 상기 복수의 블록에 대한 상기 파티션 모드를 결정하는 것에서, 상기 적어도 하나의 프로세서는 상기 시스템이:

제1 표시 신호에 따라, 상기 제1 블록이 서브블록 병합 모드를 사용하여 코딩되는지를 결정하는 것;

상기 제1 블록이 상기 서브블록 병합 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 제2 표시 신호에 따라, 상기 제1 블록이 정규 모드 또는 움직임 벡터 차분을 갖는 병합 모드(MMVD) 중 하나를 사용하여 코딩되는지를 결정하는 것;

상기 제1 블록이 상기 정규 모드 또는 상기 MMVD 중 하나를 사용하여 코딩되지 않는다는 결정에 응답하여, 제3 표시 신호에 따라, 상기 제1 블록이 결합된 인터 및 인트라 예측(CIIP) 모드를 사용하여 코딩되는지를 결정하는 것; 및

상기 제1 블록이 상기 CIIP 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 상기 복수의 블록의 상기 파티션 모드가 삼각형 파티션 모드 또는 기하학적 파티션 모드 중 하나라고 결정하는 것을 더 수행하게 하기 위해 상기 명령어의 세트를 실행하도록 구성되는, 시스템.

청구항 20

컴퓨터 시스템의 적어도 하나의 프로세서에 의해 실행가능한 명령어를 저장하는 비일시적 컴퓨터 판독가능 매체로서, 상기 명령어의 실행은 상기 컴퓨터 시스템이:

파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계;

상기 제1 파티션에 대한 제1 예측 신호 및 상기 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 상기 복수의 블록에 대해 인터 예측을 수행하는 단계; 및

상기 파티셔닝 엣지와 연관된 엣지 블록에 대한 상기 제1 예측 신호 및 상기 제2 예측 신호를 블렌딩하는 단계를 포함하는 방법을 수행하게 하는, 비일시적 컴퓨터 판독가능 매체.

발명의 설명

기 술 분 야

[0001] 연관된 출원에 대한 상호 참조

[0002] 본 개시는 2019년 8월 15일에 출원된 미국가출원 제62/887,039호 및 2019년 9월 23일에 출원된 미국가출원 제62/903,970호에 대한 우선권의 이익을 주장하며, 이들 모두 그 전체가 본원에 참조로 통합된다.

[0003] 기술분야

[0004] 본 개시는 일반적으로 비디오 처리에 관한 것이고, 더 구체적으로 삼각형 파티셔닝(triangle partitioning) 또는 기하학적 파티셔닝을 사용하여 움직임 예측(motion prediction)을 수행하기 위한 방법 및 시스템에 관한 것이다.

배 경 기 술

[0005] 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)을 기초로 하는, 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

[0006] 본 개시의 실시예는 비디오 콘텐츠를 처리하기 위한 방법을 개시한다. 방법은: 파티셔닝 엣지(partitioning edge)를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계; 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측(inter prediction)을 수행하는 단계; 및 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는(blending) 단계를 포함할 수 있다.

[0007] 본 개시의 실시예는 비디오 콘텐츠를 처리하기 위한 시스템을 제공한다. 시스템은: 명령어의 세트를 저장하는 메모리; 및 적어도 하나의 프로세서를 포함할 수 있고, 상기 적어도 하나의 프로세서는 시스템이: 파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 것; 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측을 수행하는 것; 및 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는(blending) 것을 수행하게 하기 위해 명령어의 세트를 실행하도록 구성된다.

[0008] 본 개시의 실시예는 컴퓨터 시스템의 적어도 하나의 프로세서에 의해 실행 가능한 명령어를 저장하는 비일시적 컴퓨터 판독가능 매체를 제공하고, 명령의 실행은 컴퓨터 시스템이 파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계; 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측을 수행하는 단계; 및 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계를 포함하는 방법을 수행하게 한다.

도면의 간단한 설명

[0009] 본 개시의 실시예 및 다양한 양상은 다음의 상세한 설명 및 첨부 도면에 예시된다. 도면에 도시된 다양한 피처(feature)는 축척대로 도시된 것은 아니다.

도 1은 본 개시의 실시예와 일치하는 본 개시의 실시예와 일치하는 예시적인 비디오 시퀀스의 구조를 예시한다.

도 2a는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 예시적인 인코딩 프로세스의 개략도를 예시한다.

도 2b는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 다른 예시적인 인코딩 프로세스의 개략도를 예시한다.

도 3a는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 예시적인 디코딩 프로세스의 개략도를 예시한다.

도 3b는 본 개시의 실시예와 일치하는, 하이브리드 비디오 코딩 시스템의 다른 예시적인 디코딩 프로세스의 개략도를 예시한다.

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

도 5는 본 개시의 실시예와 일치하는, 삼각형 파티션을 기초로 하는 예시적인 인터 예측을 예시한다.

도 6은 본 개시의 실시예와 일치하는, 병합(merge) 인덱스를 움직임 벡터와 연관시키기 위한 예시적인 테이블을 예시한다.

도 7은 본 개시의 실시예와 일치하는, 예시적인 크로마(chroma) 가중치 맵 및 예시적인 루마(luma) 가중치 샘플을 예시한다.

도 8은 본 개시의 실시예와 일치하는, 단방향-예측된(uni-predicted) 또는 양방향-예측된(bi-predicted) 영역에

위치된 움직임 벡터를 저장하기 위한 4×4 서브블록의 예시를 예시한다.

도 9는 본 개시의 실시예와 일치하는, 병합 모드의 예시적인 신택스 구조를 예시한다.

도 10은 본 개시의 실시예와 일치하는, 병합 모드의 다른 예시적인 신택스 구조를 예시한다.

도 11은 본 개시의 실시예와 일치하는, 예시적인 기하학적 파티션을 예시한다.

도 12는 본 개시의 실시예와 일치하는, dis[]에 대한 예시적인 룩업 테이블을 예시한다.

도 13은 본 개시의 실시예와 일치하는, GeoFilter[]에 대한 예시적인 룩업 테이블을 예시한다.

도 14(도 14a-14d)는 본 개시의 실시예와 일치하는 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 15a(도 15aa-15ab)는 본 개시의 실시예와 일치하는, stepDis에 대한 예시적인 룩업 테이블을 예시한다.

도 15b(도 15ba-15bb)는 본 개시의 실시예와 일치하는, stepDis에 대한 다른 예시적인 룩업 테이블을 예시한다.

도 15c(도 15ca-15cd)는 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 15d(도 15da-15dc)는 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 108로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 15e(도 15ea-15eb)는 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 80으로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 15f(도 15fa-15fb)는 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 64로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 16은 본 개시의 실시예와 일치하는, 기하학적 파티션 모드에 대한 예시적인 신택스 구조를 예시한다.

도 17a는 본 개시의 실시예와 일치하는, 기하학적 파티션 모드에 대한 다른 예시적인 신택스 구조를 예시한다.

도 17b는 본 개시의 실시예와 일치하는, 기하학적 파티션 모드에 대한 또 다른 예시적인 신택스 구조를 예시한다.

도 18은 본 개시의 실시예와 일치하는, 인터-예측된 블록에 대한 예시적인 서브 블록 변환을 예시한다.

도 19는 본 개시의 실시예와 일치하는 통합된 신택스 구조의 예시를 예시한다.

도 20은 본 개시의 실시예와 일치하는 통합된 신택스 구조의 다른 예시를 예시한다.

도 21은 본 개시의 실시예와 일치하는 통합된 신택스 구조의 또 다른 예시를 예시한다.

도 22a(도 22aa-22ad)는 본 개시의 실시예와 일치하는, 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 22b는 본 개시의 실시예와 일치하는, 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 다른 예시적인 룩업 테이블을 예시한다.

도 23은 본 개시의 실시예와 일치하는, 더 큰 블록 치수만을 파티셔닝하는 각도를 허용하는 예시를 예시한다.

도 24(도 24a-24d)는 본 개시의 실시예와 일치하는, 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 또 다른 예시적인 룩업 테이블을 예시한다.

도 25(도 25a-25d)는 본 개시의 실시예와 일치하는, 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 또 다른 예시적인 룩업 테이블을 예시한다.

도 26은 본 개시의 실시예와 일치하는 Dis[]에 대한 예시적인 룩업 테이블을 예시한다.

도 27은 본 개시의 실시예와 일치하는, 예시적인 코딩 유닛 신택스 구조를 예시한다.

도 28은 본 개시의 실시예와 일치하는, SBT 및 GEO 파티셔닝의 예시를 예시한다.

도 29는 본 개시의 실시예와 일치하는, 다른 예시적인 코딩 유닛 신택스 구조를 예시한다.

도 30(도 30a-30b)은 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 80으로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 31(도 31a-31b)은 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 64로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 32는 본 개시의 실시예와 일치하는, Rho[]에 대한 예시적인 룩업 테이블을 예시한다.

도 33은 본 개시의 실시예와 일치하는, 블록당 각 연산의 수의 테이블을 예시한다.

도 34는 본 개시의 실시예와 일치하는, Rho_{subblk}[]에 대한 예시적인 룩업 테이블을 예시한다.

도 35a는 본 개시의 실시예와 일치하는, 각도 135° 의 예시적인 마스크를 예시한다.

도 35b는 본 개시의 실시예와 일치하는, 각도 45° 의 예시적인 마스크를 예시한다.

도 36a는 본 개시의 실시예와 일치하는, 각도 135° 의 예시적인 마스크를 예시한다.

도 36b는 본 개시의 실시예와 일치하는, 45° 의 예시적인 마스크를 예시한다.

도 37은 본 개시의 실시예와 일치하는, 상이한 블록 형태에 대한 삼각형 파티션 모드의 예시적인 각도를 예시한다.

도 38(도 38a-38d)은 본 개시의 실시예와 일치하는, 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 예시적인 룩업 테이블을 예시한다.

도 39는 본 개시의 실시예와 일치하는, 비디오 콘텐츠를 처리하기 위한 예시적인 방법의 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0010] 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 개시과 일치하는 모든 구현을 나타내는 것은 아니다. 대신에, 이들은 첨부된 청구범위에서 인용된 본 개시에 관련된 양상과 일치하는 장치 및 방법의 예시일 뿐이다. 달리 구체적으로 언급되지 않는 한, "또는"이란 용어는 실행 불가능한 경우를 제외하고 모든 가능한 조합을 포함한다. 예를 들어, 구성요소가 A 또는 B가 포함될 수 있다고 명시되어 있으면, 달리 구체적으로 명시되지 않거나 실행 불가능한 경우를 제외하고, 구성요소는 A 또는 B, 또는 A 및 B를 포함할 수 있다. 제2 예시로서, 구성요소가 A, B 또는 C를 포함할 수 있으면, 달리 구체적으로 명시되지 않거나 실행 불가능한 경우를 제외하고, 구성요소는 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.

[0011] 비디오 코딩 시스템은 예를 들어, 소비되는 저장 공간을 감소시키거나 이러한 신호와 연관된 송신 대역폭 소비를 감소시키기 위해 디지털 비디오 신호를 압축하는 데 종종 사용된다. (예를 들어, 1920×1080 픽셀의 해상도를 갖는) 고화질(HD) 비디오가 온라인 비디오 스트리밍, 화상 회의 또는 비디오 모니터링과 같은 비디오 압축의 다양한 응용에서 인기를 얻으면서, 비디오 데이터의 압축 효율성을 높일 수 있는 비디오 코딩 도구를 개발하려는 지속적인 요구가 있다.

[0012] 예를 들어, 비디오 모니터링 응용은 다수의 응용 시나리오(예를 들어, 보안, 트래픽, 환경 모니터링 등)에서 점점 더 광범위하게 사용되며, 모니터링 디바이스의 수 및 해상도는 계속 빠르게 증가하고 있다. 다수의 비디오 모니터링 응용 시나리오는 사용자에게 HD 비디오를 제공하여 더 많은 정보를 캡처하는 것을 선호하며, 이는 이러한 정보를 캡처하기 위해 프레임당 더 많은 픽셀을 갖는다. 하지만, HD 비디오 비트스트림은 송신을 위한 높은 대역폭 및 저장을 위한 큰 공간을 요구하는 높은 비트레이트를 가질 수 있다. 예를 들어, 평균 1920×1080 해상도를 갖는 모니터링 비디오 스트림은 실시간 송신을 위해 4Mbps의 높은 대역폭을 요구할 수 있다. 또한, 비디오 모니터링은 일반적으로, 7×24를 연속적으로 모니터링하며, 이는 비디오 데이터가 저장되어야 하는 경우 저장 시스템에 큰 과제가 될 수 있다. 그러므로, HD 비디오의 높은 대역폭 및 큰 저장에 대한 요구는 비디오 모니터링에서 그의 대규모 전개에 대한 주요 제약이 된다.

[0013] 비디오는 시각적 정보를 저장하기 위해 시간적인 시퀀스로 배열된 정적 픽처(또는 "프레임")의 세트이다. 비디오 캡처 디바이스(예를 들어, 카메라)는 이 픽처를 시간적인 시퀀스로 캡처하고 저장하기 위해 사용될 수 있고, 비디오 재생 디바이스(예를 들어, 디스플레이의 기능을 갖는 텔레비전, 컴퓨터, 스마트폰, 태블릿 컴퓨터, 비디

오 플레이어 또는 임의의 최종-사용자 단말기)는 이러한 픽처를 시간적인 시퀀스로 디스플레이하기 위해 사용될 수 있다. 또한, 일부 응용에서, 비디오 캡처링 디바이스는 모니터링, 회의 또는 실시간 방송과 같이 캡처된 비디오를 비디오 재생 디바이스(예를 들어, 모니터를 갖는 컴퓨터)에 실시간으로 송신할 수 있다.

[0014] 이러한 응용에 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(예를 들어, 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. 예를 들어, 인코더 및 디코더의 하드웨어 구현은 하나 이상의 마이크로프로세서, 디지털 신호 프로세서(DSP), 애플리케이션-특정 집적 회로(application-specific integrated circuits, ASIC), 필드 프로그램 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 해제는 MPEG-1, MPEG-2, MPEG-4, H.26x 시리즈 등과 같은 다양한 알고리즘 또는 표준에 의해 구현될 수 있다. 일부 응용에서, 코덱은 제1 코딩 표준으로부터 비디오를 압축 해제하고 제2 코딩 표준을 사용하여 압축해제된 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. 폐기된 중요하지 않은 정보가 완전히 재구성될 수 없는 경우, 이러한 인코딩 프로세스는 "손실이 있는 것(lossy)"으로 지칭될 수 있다. 그렇지 않으면, "무손실"로 지칭될 수 있다. 대부분의 인코딩 프로세스는 손실이 있고, 이는 필요한 저장 공간 및 송신 대역폭을 감소시키기 위한 절충안이다.

[0016] 인코딩되는 픽처("현재 픽처"로 지칭됨)의 유용한 정보는 참조 픽처(예를 들어, 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, 이는 그 자신의 참조 픽처임)는 "I-픽처"로 지칭된다. 참조 픽처로서 이전 픽처를 사용하여 코딩된 픽처는 "P-픽처"로 지칭된다. 참조 픽처로서 이전 픽처 및 향후 픽처 둘 모두를 사용하여 코딩된 픽처(즉, 참조는 "양방향"임)는 "B-픽처"로 지칭된다.

[0018] 이전에 언급된 바와 같이, HD 비디오를 사용하는 비디오 모니터링은 높은 대역폭 및 큰 저장의 요구에 직면한다. 이러한 문제를 해결하기 위해 인코딩된 비디오의 비트레이트가 감소될 수 있다. I-픽처, P-픽처 및 B-픽처 중 I-픽처가 가장 높은 비트레이트를 갖는다. 대부분의 모니터링 비디오의 배경이 거의 정적이기 때문에, 인코딩된 비디오의 전체 비트레이트를 감소시키기 위한 한 가지 방법은 비디오 인코딩에 더 적은 I-픽처를 사용하는 것이다.

[0019] 하지만, I-픽처가 인코딩된 비디오에서 일반적으로 지배적이지 않기 때문에, 더 적은 I-픽처를 사용하는 것의 개선은 사소할 수 있다. 예를 들어, 일반적인 비디오 비트스트림에서 I-픽처, B-픽처 및 P-픽처의 비율은 1:20:9일 수 있으며, 여기서 I-픽처는 전체 비트레이트의 10% 미만을 차지할 수 있다. 다시 말해, 이러한 예시에서 모든 I-픽처가 제거되더라도 감소된 비트레이트는 10%를 넘을 수 없다.

[0020] 도 1은 본 개시의 실시예와 일치하는, 예시적인 비디오 시퀀스(100)의 구조를 예시한다. 비디오 시퀀스(100)는 실시간 비디오 또는 캡처되고 보관되는 비디오일 수 있다. 비디오(100)는 실제 비디오, 컴퓨터-생성된 비디오(예를 들어, 컴퓨터 게임 비디오)또는 이들의 조합(예를 들어, 증강-현실 효과를 갖는 실제 비디오)일 수 있다. 비디오 시퀀스(100)는 비디오 캡처 디바이스(예를 들어, 카메라), 이전에 캡처된 비디오를 포함하는 비디오 보관(예를 들어, 저장 디바이스에 저장된 비디오 파일), 또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스(예를 들어, 비디오 브로드캐스트 트랜시버)로부터 입력될 수 있다.

[0021] 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 자신이다. 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(예를 들어, 픽처(104))의 참조 픽처는 픽처 직전 또는 직후가 아닐 수 있다. 예를 들어, 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.

[0022] 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. 예를 들어, 도 1에서의 구조(110)는 비디오 시퀀스(100)의 픽처(예를 들어, 픽처(102 내지 108) 중 임의의 것)의 예시적인 구조를 도시한다. 구조(110)에서, 픽처는 4×4 기본 처리 유닛으로 분할되고, 이의 경계는 점선으로 도시된다. 일부 실시예에서, 기본 처리 유닛은 일부 비디오 코딩 표준(예를 들어, MPEG 패밀리, H.261, H.263, 또는 H.264/AVC)에서 "매크로블록(macroblocks)"으로 지칭될 수 있거나, 또는 일부 다른 비디오 코딩 표준(예를 들어, H.265/HEVC 또는 H.266/VVC)에서 "코딩 트리 유닛(coding tree units)"("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 코딩 효율 및 레벨의 밸런스를 기초로 픽처에 대해 선택될 수 있다.

[0023] 기본 처리 유닛은 컴퓨터 메모리에 (예를 들어, 비디오 프레임 버퍼에) 저장된 상이한 타입의 비디오 데이터의 그룹을 포함할 수 있는 논리적 유닛일 수 있다. 예를 들어, 컬러 픽처의 기본 처리 유닛은 무색의 밝기 정보를 나타내는 루마 성분(Y), 색상 정보를 나타내는 하나 이상의 크로마 성분(예를 들어, Cb 및 Cr) 및 연관된 신택스 요소를 포함할 수 있고, 여기서 루마 및 크로마 성분은 기본 처리 유닛과 동일한 크기를 가질 수 있다. 루마 및 크로마 성분은 일부 비디오 코딩 표준(예를 들어, H.265/HEVC 또는 H.266/VVC)에서 "코딩 트리 블록(coding tree blocks)"("CTB")으로 지칭될 수 있다. 기본 처리 유닛에 대해 수행된 임의의 동작은 그의 루마 및 크로마 성분의 각각에 대해 반복적으로 수행될 수 있다.

[0024] 비디오 코딩은 다수의 동작 스테이지를 가지며, 이의 예시는 도 2a-2b 및 3a-3b에서 상세히 설명될 것이다. 각 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(예를 들어, MPEG 패밀리, H.261, H.263, 또는 H.264/AVC)에서 "블록"으로 지칭될 수 있거나, 또는 일부 다른 비디오 코딩 표준(예를 들어, 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 (예를 들어, 비디오 프레임 버퍼에) 저장된 상이한 타입의 비디오 데이터(예를 들어, Y, Cb, Cr 및 연관된 신택스 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.

[0025] 예를 들어, 모드 결정 스테이지(그의 예시가 도 2b에서 상세히 설명될 것임)에서, 인코더는 기본 처리 유닛에 사용하기 위한 어떤 예측 모드(예를 들어, 인트라-픽처 예측 또는 인터-픽처 예측)를 결정할 수 있으며, 이는 이러한 결정을 하기엔 너무 클 수 있다. 인코더는 기본 처리 유닛을 다수의 기본 처리 서브-유닛(예를 들어, H.265/HEVC 또는 H.266/VVC에서와 같이 CU)으로 분할할 수 있으며, 각 개별적인 기본 처리 서브-유닛에 대해 예측 타입을 결정할 수 있다.

[0026] 다른 예시에 대해, 예측 스테이지(그의 예시가 도 2a에서 상세히 설명됨)에서, 인코더는 기본 처리 서브-유닛(예를 들어, CU)의 레벨에서 예측 동작을 수행할 수 있다. 하지만, 일부 경우에서, 기본 처리 서브-유닛은 처리하기에 여전히 너무 클 수 있다. 인코더는 예측 동작이 수행될 수 있는 레벨에서, 기본 처리 서브-유닛을 더 작은 세그먼트(예를 들어, H.265/HEVC 또는 H.266/VVC에서 "예측 블록" 또는 "PB"로 지칭됨)로 더 분할할 수 있다.

[0027] 다른 예시에 대해, 변환 스테이지(그의 예시가 도 2a에서 상세히 설명됨)에서, 인코더는 잔차 기본 처리 서브-유닛(예를 들어, CU)에 대한 변환 동작을 수행할 수 있다. 하지만, 일부 경우에서, 기본 처리 서브-유닛은 처리

하기에 여전히 너무 클 수 있다. 인코더는 변환 동작이 수행될 수 있는 레벨에서, 기본 처리 서브-유닛을 더 작은 세그먼트(예를 들어, 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. 예를 들어, H.265/HEVC 또는 H.266/VVC에서, 동일한 CU의 예측 블록 및 변환 블록은 상이한 크기 및 개수를 가질 수 있다.

[0028] 도 1의 구조(110)에서, 기본 처리 유닛(112)은 그의 경계가 점선으로 도시된, 3×3 기본 처리 서브-유닛으로 더 분할된다. 동일한 픽처의 상이한 기본 처리 유닛이 상이한 체계에서 기본 처리 서브-유닛으로 분할될 수 있다.

[0029] 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 각 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. 예를 들어, 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.

[0030] 예를 들어, 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.

[0031] 도 2a는 본 개시의 실시예와 일치하는, 예시적인 인코딩 프로세스(200A)의 개략도를 예시한다. 예를 들어, 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 유사하게, 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 각 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 각 원본 픽처에 대한 기본 처리 유닛의 레벨에서 프로세스(200A)를 수행할 수 있다. 예를 들어, 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 각 원본 픽처의 영역(예를 들어, 영역(114-118))에 대해 프로세스(200A)를 병렬로 수행할 수 있다.

[0032] 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"로서 지칭될 수 있다. 재구성 경로는 인코더 및 디코더 둘 모두가 예측을 위해 동일한 참조를 사용하는 것을 보장하도록 사용될 수 있다.

[0033] 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.

[0034] 프로세스(200A)를 참조하면, 인코더는 비디오 캡처링 디바이스(예를 들어, 카메라)에 의해 생성되는 비디오 시

퀀스(202)를 수신할 수 있다. 본원에서 사용된 "수신하다"라는 용어는 수신, 입력, 취득, 검색, 획득, 판독, 액세스 또는 데이터를 입력하기 위한 임의의 방식에서의 임의의 동작을 지칭할 수 있다.

[0035] 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.

[0036] 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. 예를 들어, 인코더는 원본 BPU의 대응하는 픽셀의 값으로부터 예측된 BPU(208)의 픽셀의 값(예를 들어, 그레이스케일 값 또는 RGB 값)을 감산할 수 있다. 잔차 BPU(210)의 각 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.

[0037] 잔차 BPU(210)를 더 압축하기 위해, 변환 스테이지(212)에서, 인코더는 이를 2차원 "기본 패턴" - 각 기본 패턴은 "변환 계수와 연관됨 -으로 분해함으로써 잔차 BPU의 공간 과잉을 저감할 수 있다. 기본 패턴은 동일한 크기(예를 들어, 잔차 BPU(210)의 크기)를 가질 수 있다. 각 기본 패턴은 잔차 BPU(210)의 변동 주파수(variation frequency)(예를 들어, 밝기 변동의 주파수) 성분을 나타낼 수 있다. 기본 패턴 중 어느 것도 임의의 다른 기본 패턴의 임의의 조합(예를 들어, 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(예를 들어, 삼각 함수)와 유사하고 변환 계수는 기본 함수와 연관된 계수와 유사하다.

[0038] 상이한 변환 알고리즘이 상이한 기본 패턴을 사용할 수 있다. 다양한 변환 알고리즘은 예를 들어, 이산 코사인 변환, 이산 사인 변환 등과 같은 변환 스테이지(212)에서 사용될 수 있다. 변환 스테이지(212)에서의 변환은 역으로 이루어진다(invertible). 즉, 인코더는 변환의 역동작("역변환"으로 지칭됨)에 의해 잔차 BPU(210)를 복원할 수 있다. 예를 들어, 잔차 BPU(210)의 픽셀을 복원하기 위해, 역변환은 기본 패턴의 대응하는 픽셀의 값을 각각의 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.

[0039] 인코더는 양자화 스테이지(214)에서 변환 계수를 더 압축할 수 있다. 변환 프로세스에서, 상이한 기본 패턴이 상이한 변동 주파수(예를 들어, 밝기 변동 주파수)를 나타낼 수 있다. 인간의 눈은 일반적으로 낮은-주파수 변화를 더 잘 인식하기 때문에, 인코더는 디코딩에서 현저한 품질 저하를 초래하지 않으면서 높은-주파수 변동의 정보를 무시할 수 있다. 예를 들어, 양자화 스테이지(214)에서, 인코더는 각 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 이루어지고, 여기서 양자화된 변환 계수(216)는 양자화의 역동작("역양자화"로 지칭됨)에서 변환 계수로 재구성될 수 있다.

[0040] 인코더가 반올림 연산에서 이러한 나눗셈의 나머지를 무시하기 때문에, 양자화 스테이지(214)는 손실이 있을 수 있다. 전형적으로, 양자화 스테이지(214)는 프로세스(200A)에서 최대 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.

[0041] 이진 코딩 스테이지(226)에서, 인코더는 예를 들어, 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 예를 들어, 예측 스테이지(204)에서 사용되는 예측 모드, 예측 동작의 파라미터, 변환 스테이지(212)에서의 변환 타입, 양자화 프로세스의 파라미터(예를 들어, 양자화 파라미터), 인코더 제어 파라미터(예를 들어, 비트레이트 제어 파라미터) 등과 같은 이진 코딩 스테이지(226)에서의 다른 정보를 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.

[0042] 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 계수를 기초로, 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.

[0043] 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. 예를 들어, 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.

[0044] 도 2b는 본 개시의 실시예와 일치하는, 다른 예시적인 인코딩 프로세스(200B)의 개략도를 예시한다. 프로세스(200B)는 프로세스(200A)로부터 수정될 수 있다. 예를 들어, 프로세스(200B)는 하이브리드 비디오 인코딩 표준(예를 들어, 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.

[0045] 일반적으로, 예측 기술은 두 개의 타입: 공간 예측 및 시간 예측으로 카테고리화될 수 있다. 공간 예측(예를 들어, 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 리던던시를을 감소시킬 수 있다. 시간 예측(예를 들어, 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 과잉을 저감할 수 있다.

[0046] 프로세스(200B)를 참조하면, 순방향 경로에서, 인코더는 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)에서 예측 동작을 수행한다. 예를 들어, 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 예를 들어, 선형 외삽(linear extrapolation) 또는 내삽(interpolation), 다항식 외삽 또는 내삽 등을 포함할 수 있다. 일부 실시예에서, 인코더는 가령, 예측된 BPU(208)의 각 픽셀에 대해 대응하는 픽셀의 값을 외삽함으로써, 픽셀 레벨에서 외삽을 수행할 수 있다. 외삽을 위해 사용된 이웃하는 BPU는 (예를 들어, 원본 BPU의 상부에서) 수직 방향, (예를 들어, 원본 BPU의 좌측에서) 수평 방향, (예를 들어, 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 예를 들어, 원본 BPU에 대한 사용된 이웃하는 BPU의 위치(예를 들어, 좌표), 사용된 이웃하는 BPU의 크기, 외삽의 파라미터, 사용된 이웃하는 BPU의 방향 등을 포함할 수 있다.

[0047] 다른 예시에 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. 예를 들어, 인코더는 재구성된 BPU를 생성하기 위해, 재구성된 잔차 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"으로 지칭됨)에서 매칭 영역을 검색하기 위해, "움직임 추정"의 동작을 수행할 수 있다. 참조 픽처에서 검색 윈도우의 위치는 현재 픽처에서 원본 BPU의 위치를 기초로 결정될 수 있다. 예를 들어, 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 (예를 들어, 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(예를 들어, 그보다 작거나, 이와 동일하거나, 그보다 크거나 또는 상이한 형태인)를 가질 수 있다. 참조 픽처 및 현재 픽처가 (예를 들어, 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. (예를 들어, 도 1에서의 픽처(106)와 같이) 다수의 참조 픽처가 사용될 때, 인코더는 매칭 영역을 검색하고, 각 참조 픽처에 대해 그의 연관된 움직임 벡터를 결정할 수 있다. 일부 실시예에서, 인코더는 각각의 매칭 참조 픽처의 매칭 영역의 픽셀값에 대해 가중치를 할당할 수 있다.

[0048] 움직임 추정은 예를 들어, 병진(translation), 회전, 주밍(zooming) 등과 같은 다양한 타입의 움직임을 식별하기 위해 사용될 수 있다. 인터 예측에 대해, 예측 데이터(206)는 예를 들어, 매칭 영역의 위치(예를 들어, 좌표), 매칭 영역과 관련된 움직임 벡터, 참조 픽처의 개수, 참조 픽처와 연관된 가중치 등을 포함할 수 있다.

[0049] 예측된 BPU(208)를 생성하기 위해, 인코더는 "움직임 보상"의 동작을 수행할 수 있다. 움직임 보상은 예측 데이터(206)(예를 들어, 움직임 벡터) 및 예측 참조(224)를 기초로 예측된 BPU(208)를 재구성하는데 사용될 수 있다. 예를 들어, 인코더는 움직임 벡터에 따라 참조 픽처의 매칭 영역을 이동시킬 수 있으며, 여기서 인코더는 현재 픽처의 원본 BPU를 예측할 수 있다. 다수의 참조 픽처가 (예를 들어, 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.

[0050] 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. 예를 들어, 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 방향 둘 모두에서 하나 이상의 참조 픽처를 사용할 수 있다. 예를 들어, 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.

[0051] 프로세스(200B)의 순방향 경로를 다시 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(예를 들어, 인트라 예측 또는 인터 예측 중 하나)를 선택할 수 있다. 예를 들어, 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.

[0052] 프로세스(200B)의 재구성 경로에서, 순방향 경로에서 인트라 예측 모드가 선택된 경우, 예측 참조(224)(예를 들어, 현재 픽처에서 인코딩되고 재구성된 현재 BPU)를 생성한 이후에, 인코더는 차후 사용을 위해(예를 들어, 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)(예를 들어, 모든 BPU가 인코딩되고 재구성된 현재 픽처)를 생성한 이후에, 인코더는 인터 예측에 의해 도입된 왜곡(예를 들어, 블로킹 아티팩트(blocking artifacts))을 감소시키거나 제거하기 위해 예측 참조(224)에 루프 필터를 적용할 수 있는 루프 필터 스테이지(232)에 예측 참조(224)를 공급할 수 있다. 인코더는 예를 들어, 디블록킹(deblocking), 샘플 적응적 오프셋, 적응적 루프 필터 등과 같은 루프 필터 스테이지(232)에 다양한 루프 필터 기법을 적용할 수 있다. 루프-필터링된 참조 픽처는 차후 사용을 위해(예를 들어, 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(예를 들어, 루프 필터 강도)를 인코딩할 수 있다.

[0053] 도 3a는 본 개시의 실시예와 일치하는 예시적인 디코딩 프로세스(300A)의 개략도를 예시한다. 프로세스(300A)는 도 2a에서의 압축 프로세스(200A)에 대응하는 압축해제 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 압축해제 프로세스(예를 들어, 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 각 픽처에 대해 기본 처리 유닛(BPU)의 레벨에서 프로세스(300A)를 수행할 수 있다. 예를 들어, 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 각 픽처의 영역(예를 들어, 영역(114 내지 118))에 대해 병렬로 프로세스(300A)를 수행할 수 있다.

[0054] 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(예를 들어, 컴퓨터 메모리에서 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.

[0055] 디코더는 인코딩된 픽처의 각 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.

[0056] 이진 디코딩 스테이지(302)에서, 디코더는 인코더에 의해 사용된 이진 코딩 기법(예를 들어, 엔트로피 코딩, 가변 길이 코딩, 산술 코딩, 허프만 코딩, 컨텍스트-적응적 이진 산술 코딩 또는 임의의 다른 무손실 압축 알고리즘)의 역동작을 수행할 수 있다. 일부 실시예에서, 예측 데이터(206) 및 양자화된 변환 계수(216) 이외에, 디코더는 예를 들어, 예측 모드, 예측 동작의 파라미터, 변환 타입, 양자화 프로세스의 파라미터(예를 들어, 양자화 파라미터), 인코더 제어 파라미터(예를 들어, 비트레이트 제어 파라미터) 등과 같은 이진 디코딩 스테이지(302)에서 다른 정보를 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.

[0057] 도 3b는 본 개시의 일부 실시예와 일치하는 다른 예시적인 디코딩 프로세스(300B)의 개략도를 예시한다. 프로세스(300B)는 프로세스(300A)로부터 수정될 수 있다. 예를 들어, 프로세스(300B)는 하이브리드 비디오 코딩 표준(예를 들어, 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.

[0058] 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 사용된 것인지에 의존하여, 다양한 타입의 데이터를 포함할 수 있다. 예를 들어, 현재 BPU를 인코딩하기 위해, 인코더에 의해 인트라 예측이 사용된 경우, 예측 데이터(206)는 인트라 예측을 나타내는 예측 모드 표시자(예를 들어, 플래그 값), 인트라 예측 동작의 파라미터 등을 포함할 수 있다. 인트라 예측 동작의 파라미터는 예를 들어, 참조로서 사용된 하나 이상의 이웃하는 BPU의 위치(예를 들어, 좌표), 이웃하는 BPU의 크기, 외삽의 파라미터, 원본 BPU에 대한 이웃하는 BPU의 방향 등을 포함할 수 있다. 다른 예시에 대해, 현재 BPU를 인코딩하기 위해, 인코더에 의해 인터 예측이 사용되는 경우, 예측 데이터(206)는 인터 예측을 나타내는 예측 모드 표시자(예를 들어, 플래그 값), 인터 예측 동작의 파라미터 등을 포함할 수 있다. 인터 예측 동작의 파라미터는 예를 들어, 현재 BPU와 연관된 참조 픽처의 개수, 참조 픽처와 각각 연관된 가중치, 각각의 참조 픽처에서의 하나 이상의 매칭 영역의 위치(예를 들어, 좌표), 매칭 영역과 각각 연

관된 하나 이상의 움직임 벡터 등을 포함할 수 있다.

[0059] 예측 모드 표시자를 기초로, 디코더는 공간 예측 스테이지(2042)에서 공간 예측(예를 들어, 인트라 예측)을 수행할지, 또는 시간 예측 스테이지(2044)에서 시간 예측(예를 들어, 인터 예측)을 수행할 지를 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.

[0060] 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. 예를 들어, 현재 BPU가 공간 예측 스테이지(2042)에서 인트라 예측을 사용하여 디코딩되는 경우, 예측 참조(224)(예를 들어, 디코딩된 현재 BPU)를 생성한 이후에, 디코더는 차후 사용을 위해(예를 들어, 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)(예를 들어, 모든 BPU가 디코딩된 참조 픽처)를 생성한 이후에, 인코더는 왜곡(예를 들어, 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(예를 들어, 비디오 비트스트림(228)의 향후 인코딩된 픽처에 대한 인터-예측 참조 픽처로서 사용되도록) 버퍼(234)(예를 들어, 컴퓨터 메모리에서 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(예를 들어, 루프 필터 강도)를 더 포함할 수 있다.

[0061] 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. 예를 들어, 프로세서(402)는 중앙 처리 유닛(또는 "CPU"), 그래픽 처리 유닛(또는 "GPU"), 신경 처리 유닛(neural processing unit, "NPU"), 마이크로컨트롤러 유닛(microcontroller unit, "MCU"), 광학 프로세서, 프로그래머블 로직 컨트롤러, 마이크로컨트롤러, 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 소자(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. 예를 들어, 도 4에 도시된 바와 같이, 프로세서(402)는 프로세서(402a), 프로세서(402b) 및 프로세서(402n)를 포함하는 다수의 프로세서를 포함할 수 있다.

[0062] 장치(400)는 또한, 데이터(예를 들어, 명령어의 세트, 컴퓨터 코드, 중간 데이터 등)를 저장하도록 구성되는 메모리(404)를 포함할 수 있다. 예를 들어, 도 4에 도시된 바와 같이, 저장된 데이터는 프로그램 명령어(예를 들어, 프로세스(200A, 200B, 300A 또는 300B)에서의 스테이지를 구현하기 위한 프로그램 명령어) 및 처리를 위한 데이터(예를 들어, 비디오 시퀀스(202), 비디오 비트스트림(228) 또는 비디오 스트림(304))를 포함할 수 있다. 프로세서(402)는 프로그램 명령어 및 처리를 위한 데이터에 (예를 들어, 버스(410)를 통해) 액세스할 수 있고, 처리를 위한 데이터에 대한 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.

[0063] 버스(410)는 내부 버스(예를 들어, CPU-메모리 버스), 외부 버스(예를 들어, 통합 직렬 버스 포트(universal serial bus port,), 주변 구성요소 상호연결 고속 포트(peripheral component interconnect express port)) 등

과 같은, 장치(400) 내의 구성요소 사이에서 데이터를 전송하는 통신 디바이스일 수 있다.

[0064] 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.

[0065] 장치(400)는 네트워크(예를 들어, 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 컨트롤러(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.

[0066] 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(예를 들어, 마우스, 터치패드 또는 터치스크린), 키보드, 디스플레이(예를 들어, 음극선관 디스플레이, 액정 디스플레이 또는 발광 다이오드 디스플레이), 비디오 입력 디바이스(예를 들어, 비디오 아카이브(video archive)에 결합된 카메라 또는 입력 인터페이스) 등을 포함할 수 있지만, 이에 제한되지 않는다.

[0067] 비디오 코덱(예를 들어, 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. 예를 들어, 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 예시에 대해, 프로세스(200A, 200B, 300A 또는 300B)의 일부 또는 모든 스테이지는 특수화된 데이터 처리 회로(예를 들어, FPGA, ASIC, NPU 등)와 같은 장치(400)의 하나 이상의 하드웨어 모듈로서 구현될 수 있다.

[0068] 본 개시는 움직임 예측에 사용하기 위한 블록 분할 방법을 제공한다. 개시된 방법은 인코더 또는 디코더에 의해 수행될 수 있음이 고려된다.

[0069] 인터 예측을 위한 삼각형 파티션 모드가 지원된다. 삼각형 파티션 모드는 8×8 이상의 블록에 적용될 수 있고, 삼각형 스킵 또는 병합 모드로 코딩된다. 삼각형 스킵/병합 모드는 정규 병합 모드, MMVD 모드, 결합된 인터 및 인트라 예측(Combined Inter and Intra Prediction, CIIP) 모드 또는 서브블록 병합 모드와 병렬로 시그널링된다.

[0070] 삼각형 파티션 모드가 사용될 때, 대각선 분할 또는 반-대각선 분할을 사용하여 블록은 두 개의 삼각형-형태의 파티션으로 균등하게 분할할 수 있다(도 5). 블록에서 각 삼각형 파티션은 그 자신의 움직임을 사용하여 인터-예측된다. 각 파티션에 대해 단방향-예측만이 허용된다. 다시 말해, 각 파티션은 하나의 움직임 벡터 및 하나의 참조 인덱스를 갖는다. 단방향-예측 움직임 제약은 종래의 양방향-예측과 동일하게, 각 블록에 대해 두 개의 움직임 보상된 예측만이 필요하다는 것을 보장하기 위해 적용된다. 각 파티션에 대한 단방향-예측 움직임은 확장된 병합 예측을 위해 구성된 병합 후보 목록으로부터 직접적으로 도출되며, 목록에서 주어진 병합 후보로부터 단방향-예측 움직임의 선택은 아래에서 설명된 절차에 따른다.

[0071] 현재 블록에 대해 삼각형 파티션 모드가 사용되면, 삼각형 파티션의 방향(대각선 또는 역-대각선)을 나타내는 플래그 및 두 개의 병합 인덱스(각 분할에 대해 하나씩)가 더 시그널링된다. 삼각형 파티션의 각각을 예측한 이후에, 대각선 또는 역-대각선 엣지를 따르는 샘플 값은 적응적 가중치를 갖는 블렌딩 처리를 사용하여 조정된다. 이는 전체 블록에 대한 예측 신호이며, 변환 및 양자화를 위한 프로세스는 다른 예측 모드에서와 같이, 전체 블록에 적용될 수 있다. 서브-블록 변환(SBT) 모드는 삼각형 파티션 모드를 사용하여 코딩된 블록에 적용될 수 없음이 유의된다. 삼각형 파티션 모드를 사용하여 예측된 블록의 움직임 필드는 4×4 서브블록에 저장될 수 있다.

[0072] 삼각형 파티션 모드에 대한 단방향-예측 후보 목록 구성은 아래에서 설명된다.

[0073] 병합 후보 인덱스가 주어지면, 단방향-예측 움직임 벡터는 도 6에 예시된 바와 같이 확장된 병합 예측을 위해 구성된 병합 후보 목록으로부터 도출된다. 목록에서의 후보에 대해, 병합 후보 인덱스 값(즉, X=0 또는 1)의 패리티와 동일한 X를 갖는 그의 LX(L0 또는 L1) 움직임 벡터는 삼각형 파티션 모드에 대한 단방향-예측 움직임 벡터로 사용된다. 이 움직임 벡터는 도 6에서 "x"로 표시되어 있다. 대응하는 LX 움직임 벡터가 존재하지 않을

때, 확장된 병합 예측 후보 목록에서 동일한 후보의 L(1-X) 움직임 벡터는 삼각형 파티션 모드에 대한 단방향-예측 움직임 벡터로 사용한다.

[0074] 삼각형 파티션 엣지를 따라 블렌딩하는 것이 아래에서 설명된다.

[0075] 그 자신의 움직임을 사용하여 각 삼각형 파티션을 예측한 이후에, 대각선 엣지 또는 반-대각선 엣지 주변의 샘플을 도출하기 위해 두 예측 신호에 블렌딩이 적용된다. 블렌딩 프로세스에서 다음의 가중치가 사용된다:

[0076] 도 7에 도시된 바와 같이, 루마의 경우 {7/8, 6/8, 5/8, 4/8, 3/8, 2/8, 1/8} 및 크로마의 경우 {6/8, 4/8, 2/8}.

[0077] 삼각형 파티션 모드를 사용하여 예측된 블록 내의 루마 및 크로마 샘플의 각각의 가중치는 다음 식을 사용하여 계산된다:

$$R = (CuW_L > CuH_L)\ ?\ \frac{CuW_L}{CuH_L} : \frac{CuH_L}{CuW_L}$$

[0078] - 비율

[0079] - 블록이 제1 방향으로부터(예를 들어, 좌측-상단 코너로부터 우측-하단 코너로) 분할되는 경우, 분할 방향 splitDir은 0으로 설정된다. 그렇지 않으면, 블록이 제2 방향으로부터(예를 들어, 우측-상단 코너로부터 좌측-하단 코너로) 분할되는 경우, splitDir은 1로 설정된다.

$$sampleWeight_L[x][y]$$

$$= \begin{cases} clip3(0,8,\frac{x}{R} - y + 4) & if\ splitDir = 0\ and\ CuW_L > CuH_L \\ clip3(0,8,x - \frac{y}{R} + 4) & if\ splitDir = 0\ and\ CuW_L \le CuH_L \\ clip3(0,8,CuH_L - 1 - \frac{x}{R} - y + 4) & if\ splitDir = 1\ and\ CuW_L > CuH_L \\ clip3(0,8,CuW_L - 1 - x - \frac{y}{R} + 4) & if\ splitDir = 1\ and\ CuW_L \le CuH_L \end{cases}$$

$$sampleWeight_C[x][y]$$

$$= \begin{cases} clip3(0,4,\frac{x}{R} - y + 2) \times 2 & if\ splitDir = 0\ and\ CuW_C > CuH_C \\ clip3(0,4,x - \frac{y}{R} + 2) \times 2 & if\ splitDir = 0\ and\ CuW_C \le CuH_C \\ clip3(0,4,CuH_C - 1 - \frac{x}{R} - y + 2) \times 2 & if\ splitDir = 1\ and\ CuW_C > CuH_C \\ clip3(0,4,CuW_C - 1 - x - \frac{y}{R} + 2) \times 2 & if\ splitDir = 1\ and\ CuW_C \le CuH_C \end{cases}$$

[0080]

[0081] 여기서 $sampleWeight_L$은 루마 샘플에 대한 가중치 맵을 나타내고, $sampleWeight_C$는 크로마 샘플에 대한 가중치 맵을 나타내며, (x,y)는 루마/크로마 샘플의 위치를 나타내고, (CuW_L, CuH_L)은 루마 샘플에서의 블록 폭 및 높이를 나타내고, (CuW_C, CuH_C)는 크로마 샘플에서의 블록 폭 및 높이를 나타낸다.

[0082] 그 후, 삼각형 파티션 모드에서의 움직임 필드 저장이 아래에서 설명될 것이다.

[0083] 삼각형 파티션 모드에서 코딩된 블록의 움직임 벡터는 4×4 서브블록에 저장된다. 각 4×4 서브블록의 위치에 의존하여, 단방향-예측 또는 양방향-예측 움직임 벡터가 저장된다. Mv1 및 Mv2는 도 5에서의 파티션 1 및 파티션 2에 대한 단방향-예측 움직임 벡터로 각각 표시된다. 단방향-예측된 영역에서 4×4 서브 블록이 위치된 경우, 그 4×4 서브블록에 대해 Mv1 또는 Mv2가 저장된다. 그렇지 않으면, 4×4 서브블록이 양방향-예측된 영역에 위치되는 경우, 양방향-예측 움직임 벡터가 저장된다. 양방향-예측 움직임 벡터는 다음의 프로세스에 따라 Mv1 및 Mv2로부터 도출된다.

[0084] 1. Mv1 및 Mv2가 상이한 참조 픽처 목록으로부터(하나는 L0로부터, 다른 하나는 L1로부터) 있으면, Mv1 및 Mv2는 양방향-예측 움직임 벡터를 형성하기 위해 간단히 결합된다.

[0085] 2. 그렇지 않으면, Mv1 및 Mv2가 동일한 목록으로부터 있는 경우, 양방향-예측 움직임 대신에, 단방향-예측 움직임 Mv2만이 저장된다.

[0086] 4×4 서브블록 내의 모든 샘플이 가중될 때, 4×4 서브블록은 양방향-예측된 영역에 있는 것으로 간주된다는 것이 유의된다. 그렇지 않으면, 4×4 서브블록이 단방향-예측된 영역에 있는 것으로 간주된다. (그림자 영역인) 양방향-예측된 영역 및 단방향-예측된 영역의 예시가 도 8에 도시된다.

[0087] 4×4 서브블록이 양방향-예측된 영역에 위치되는지를 결정하기 위해 다음의 식이 사용될 수 있다:

[0088] - 비율

$$R = (CuW_L > CuH_L)\ ?\ \frac{CuW_L}{CuH_L} : \frac{CuH_L}{CuW_L}$$

[0089] - 블록이 좌측-상단 코너로부터 우측-하단 코너로 분할되는 경우, 분할 방향 splitDir은 0으로 설정된다. 그렇지 않으면, 블록이 우측-상단 코너로부터 좌측-하단 코너로 분할되는 경우, splitDir은 1로 설정된다.

[0090] - 변수

$$minSb = \min\left(\frac{CuW_L}{4}, \frac{CuH_L}{4}\right) - 1$$

[0091] - $CuW_L > CuH_L$ 및 $splitDir$=0인 경우,

[0092] - $\frac{x/4}{R} = \frac{y}{4}$ 인 경우, 4×4 서브블록은 양방향-예측된 영역에 위치된다;

[0093] - 그렇지 않으면, 4×4 서브블록은 단방향-예측된 영역에 위치된다;

[0094] - $CuW_L \leq CuH_L$ 및 $splitDir$=0인 경우,

[0095] - $\frac{x}{4} = \frac{y/4}{R}$ 인 경우, 4×4 서브블록은 양방향-예측된 영역에 위치된다;

[0096] - 그렇지 않으면, 4×4 서브블록은 단방향-예측된 영역에 위치된다;

[0097] - $CuW_L > CuH_L$ 및 $splitDir$=1인 경우,

[0098] - $\frac{x/4}{R} + \frac{y}{4} = minSb$ 인 경우, 4×4 서브블록은 양방향-예측된 영역에 위치된다;

[0099] - 그렇지 않으면, 4×4 서브블록은 단방향-예측된 영역에 위치된다;

[0100] - $CuW_L \leq CuH_L$ 및 $splitDir$=1인 경우,

[0101] - $\frac{x}{4} + \frac{y/4}{R} = minSb$ 인 경우, 4×4 서브블록은 양방향-예측된 영역에 위치된다;

[0102] - 그렇지 않으면, 4×4 서브블록은 단방향-예측된 영역에 위치된다;

[0103] 여기서 (x,y)는 4×4 서브블록의 좌측-상단 루마 샘플의 위치를 나타내고 (CuW_L,CuH_L)은 루마 샘플에서의 블록 폭 및 블록 높이를 나타낸다.

[0104] 삼각형 파티션 모드에 대한 예시적인 신택스 구조는 아래에서 설명된다.

[0105] 병합 모드의 예시적인 신택스 구조는 도 9-10에 각각 도시된다. 도면에 도시된 CIIP 플래그는 삼각형 파티션 모드를 사용하여 블록이 예측되는지를 나타내기 위해 사용된다.

[0106] 본 개시와 일치하는 기하학적 파티션 모드는 다음과 같이 설명된다.

[0107] 개시된 실시예에서, 기하학적 파티션 모드는 또한, 비디오 콘텐츠를 코딩하기 위해 사용될 수 있다. 기하학적 파티션 모드에서, 블록은 두 개의 파티션으로 분할되고, 두 개의 파티션은 도 11에 도시된 바와 같이 직사각형 또는 직사각형이 아닌 형태로 있을 수 있다. 그 후, 두 개의 파티션은 그 자신의 움직임 벡터로 인터-예측된다. 단방향-예측 움직임은 도 6을 참조하여 위에서 설명된 것과 동일한 프로세스를 사용하여 도출된다. 삼각형 파티션 모드에서 사용된 프로세스와 유사하게 기하학적 파티션의 각각을 예측한 이후에, 파티셔닝 엣지를 따라 샘플 값은 적응적 가중치를 갖는 블렌딩 처리를 사용하여 조정된다. 이는 전체 블록에 대한 예측 신호이며, 다른 예

측 모드에서와 같이, 변환 및 양자화 프로세스가 전체 블록에 적용될 수 있다. SBT 모드는 기하학적 파티션 모드를 사용하여 코딩된 블록에 적용될 수 있다는 것이 유의된다. 마지막으로, 기하학적 파티션 모드를 사용하여 예측된 블록의 움직임 필드는 4×4 서브블록에 저장될 수 있다. 기하학적 파티션 모드의 이점은 그것이 움직임 보상을 위한 더 유연한 파티셔닝 방법을 제공한다는 것이다.

[0108] 기하학적 파티션 모드는 폭 및 높이가 모두 8 이상이고 max(폭, 높이)/min(폭, 높이)의 비율이 4 이하인 블록에만 적용되며, 기하학적 스킵 또는 병합 모드로 코딩된다. 정규 병합 모드, MMVD 모드, CIIP 모드, 서브블록 병합 모드 또는 삼각형 파티션 모드와 병렬로 각 블록에 대해 기하학적 파티션 모드가 시그널링된다. 현재 블록에 대해 이 모드가 사용될 때, 140개의 파티셔닝 방법(32개의 양자화된 각도 + 5개의 양자화된 거리) 중 어느 하나를 나타내는 기하학적 파티션 모드 인덱스가 현재 블록을 분할하는데 사용되고, 두 개의 병합 인덱스가 더 시그널링된다. 기하학적 파티셔닝 서브모드의 총 수는 상이한 설정에 의존하는, 140(16개의 양자화된 각도 + 9개의 양자화된 거리), 108(16개의 양자화된 각도 + 7개의 양자화된 거리), 80(12개의 양자화된 각도 + 7개의 양자화된 거리) 및 64(10개의 양자화된 각도 + 7개의 양자화된 거리) 중 하나일 수 있다.

[0109] 기하학적 파티션 엣지를 따라 블렌딩하는 것은 아래와 같이 논의될 것이다. 그 자신의 움직임을 사용하여 각 기하학적 파티션을 예측한 이후에, 블렌딩 프로세스가 파티션 엣지 주위의 샘플을 도출하기 위해 두 개의 예측 신호에 적용된다. 일부 실시예에서, 각 루마 샘플에 대한 가중치는 다음의 식을 사용하여 계산된다.

$$distFromLine = ((x \ll 1) + 1) \times Dis[displacementX] + ((y \ll 1) + 1) \times Dis[displacementY] - rho$$

$$distScaled = Min((abs(distFromLine)) \gg 4,14)$$

[0110] $$sampleWeight_L[x][y] = distFromLine \leq 0 \; ? \; GeoFilter[distScaled] : 8 - GeoFilter[distScaled]$$

[0111] 여기서 (x,y)는 각 루마 샘플의 위치를 나타내고, *Dis*[] 및 *GeoFilter*[]는 각각 도 12-13의 테이블 12 및 테이블 13A-13B에 도시된 두 개의 룩업 테이블이다.

[0112] *displacementX*, *displacementY* 및 *rho* 파라미터는 다음과 같이 계산된다:

$$displacementX = angleIdx$$

$$displacementY = (displancementX + NumAngles \gg 2)\%NumAngles$$

$$rho = distanceIdx \times stepSize \times scaleStep + CuW \times Dis[displacementX] + CuH \times Dis[displacementY]$$

$$stepSize = stepDis + 64$$

$$scaleStep = (wIdx \geq hIdx) \; ? \; (1 \ll hIdx) : (1 \ll wIdx)$$

$$wIdx = log2(CuW) - 3$$

$$hIdx = log2(CuH) - 3$$

$$whRatio = (wIdx \geq hIdx) \; ? \; (wIdx - hIdx) : (hIdx - wIdx)$$

[0113]
$$angleN = \begin{cases} angleIdx & angleIdx > 0 \; and \; angleIdx \leq 8 \\ 16 - angleIdx & angleIdx > 8 \; and \; angleIdx \leq 16 \\ angleIdx - 16 & angleIdx > 16 \; and \; angleIdx \leq 24 \\ 32 - angleIdx & otherwise \end{cases}$$

[0114] $$angleN = (wIdx \geq hIdx) \; ? \; 8 - angleN : angleN$$

[0115] 여기서 (*CuW*,*CuH*)는 루마 샘플에서의 블록 폭 및 높이이고, *NumAngles*는 32로 설정되고, *angleIdx* 및

*distanceIdx*는 도 14의 테이블 14로부터 도출되고, *stepDis*는 도 15a의 테이블 15A로부터 도출된다.

[0116] 일부 실시예에서, *displacementX*, *displacementY* 및 *rho* 파라미터는 또한, 다음과 같이 계산될 수 있다.

$$displacementX = angleIdx$$

$$displacementY = (displancementX + NumAngles \gg 2)\%NumAngles$$

$$rho = distanceIdx \times (stepSize \ll scaleStep) + Dis[displacementX] \ll wIdx + Dis[displacementY] \ll hIdx$$

$$stepSize = stepDis + 77$$

$$scaleStep = (wIdx \geq hIdx)\ ?\ hIdx - 3 : wIdx - 3$$

$$wIdx = log2(CuW)$$

$$hIdx = log2(CuH)$$

$$whRatio = (wIdx \geq hIdx)\ ?\ (wIdx - hIdx) : (hIdx - wIdx)$$

$$angleN = \begin{cases} angleIdx\ \%\ 8 & wIdx \geq hIdx\ and\ (angleIdx \gg 3)\%\ 1 = 1 \\ 8 - (angleIdx\ \%\ 8) & wIdx \geq hIdx\ and\ (angleIdx \gg 3)\%\ 1 = 0 \\ 8 - (angleIdx\ \%\ 8) & wIdx < hIdx\ and\ (angleIdx \gg 3)\%\ 1 = 1 \\ angleIdx\ \%\ 8 & wIdx < hIdx\ and\ (angleIdx \gg 3)\%\ 1 = 0 \end{cases}$$

[0117]

[0118] 여기서 (*CuW*,*CuH*)는 루마 샘플에서의 블록 폭 및 높이이고, *NumAngles*는 32로 설정되고, *stepDis*는 도 15b의 테이블 15B로부터 도출된다. 기하학적 파티셔닝 서브모드의 총 수가 각각 140, 108, 80 및 64로 설정될 때, *angleIdx* 및 *distanceIdx*는 도 15c-15f의 테이블 15C, 테이블 15D, 테이블 15E 및 테이블 15F로부터 도출된다.

[0119] 크로마 샘플에 대한 가중치는 예를 들어 YUV 4:2:0 비디오 포맷에 대해 각 2x2 루마 서브블록의 좌측-상단 루마 샘플 가중치로부터 서브샘플링된다.

[0120] 기하학적 파티션 모드에서의 움직임 필드 저장은 아래에서 설명될 것이다.

[0121] 기하학적 파티션 모드로 코딩된 블록의 움직임 벡터는 4×4 서브 블록에 저장된다. 각 4×4 서브블록에 대해, 단방향-예측 또는 양방향-예측 움직임 벡터가 저장된다. 양방향-예측 움직임에 대한 도출 프로세스는 앞서 설명한 프로세스와 동일하다. 4×4 서브블록에 대해 단방향-예측 또는 양방향-예측 움직임 벡터가 저장되는지를 결정하기 위해 두 개의 종류의 방법이 제안된다.

[0122] 제1 방법에서, 4×4 서브블록에 대해, 그의 4개의 코너의 샘플 가중치 값이 합산된다. 합이 threshold2보다 작고 threshold1보다 큰 경우, 이 4×4 서브블록에 대해 양방향-예측 움직임 벡터가 저장된다. 그렇지 않으면, 단방향-예측 움직임 벡터가 저장된다. threshold1 및 threshold2는 각각 32≫(((*log2*(*CuW*)+*log2*(*CuH*))≫1)-1) 및 32-*threshold*1로 설정된다.

[0123] 제2 방법에서, 다음의 식은 그의 위치에 의존하여 이 4×4 서브블록에 대해 어떤 움직임 벡터가 저장되는지를 결정하는 데 사용된다.

$$rho_{subblk} = 3 \times Dis[displacementX] + 3 \times Dis[displacementY]$$

$$distFromLine_{subblk} = \left((x_{subblk} \ll 3) + 1\right) \times Dis[displacementX] + \left((y_{subblk} \ll 3) + 1\right) \times Dis[displacementY] - rho + rho_{subblk}$$

$$motionMask[x_{subblk}][y_{subblk}] = abs(distFromLine_{subblk}) < 256\ ?\ 2 : (distFromLine_{subblk} \leq 0\ ?\ 0 : 1)$$

[0124]

[0125] 여기서 (x_{subblk},y_{subblk})는 각 4×4 서브블록의 위치를 나타낸다. 변수 *Dis*[], *displacementX*, *displacementY* 및

*rho*는 위에서 설명된 변수와 동일하다. *motionMask*[x_{subblk}][y_{subblk}]의 값이 2와 동일할 때, 양방향-예측 움직임 벡터는 이 4×4 서브블록에 대해 저장된다. 그렇지 않으면, 이 4×4 서브블록에 대해 단방향-예측 움직임 벡터가 저장된다.

[0126] 기하학적 파티션 모드에 대한 세 개의 예시적인 신택스 구조가 각각 도 16-17b에 도시된다.

[0127] 일부 실시예에서, 서브 블록 변환이 사용될 수 있다. 서브 블록 변환에서, 잔차 블록은 도 18에 도시된 두 개의 잔차 서브 블록으로 분할된다. 두 개의 잔차 서브 블록 중 하나만이 코딩된다. 다른 잔차 서브 블록의 경우, 나머지는 0과 동일하게 설정된다.

[0128] 나머지를 갖는 인터 예측 블록에 대해, 서브 블록 변환이 적용되는지의 여부를 나타내기 위해 CU 레벨 플래그가 시그널링된다. 서브 블록 변환 모드가 사용될 때, 잔차 블록이 수평 또는 수직 방향으로 두 개의 서브 블록으로 대칭 또는 비대칭 분할임을 나타내기 위해 파라미터가 시그널링된다.

[0129] 삼각형 파티션 및 기하학적 파티션 모드는 움직임 보상의 코딩 효율을 개선하기 위한 두 개의 파티셔닝 방법이다. 삼각형 파티션은 기하학적 파티션의 서브세트로 볼 수 있다. 하지만, 현재 구현에서 기하학적 파티션 모드의 신택스 구조, 블렌딩 프로세스 및 움직임 필드 저장은 삼각형 파티션 모드의 것과 상이하다. 예를 들어, 다음 프로세스는 이 두 모드에서 상이하다.

[0130] 1. 병합 모드로 코딩된 블록의 경우, (삼각형 파티션 모드 플래그 및 기하학적 파티션 모드 플래그를 포함하는) 두 개의 플래그가 시그널링된다. 더욱이, 폭 또는 높이가 4와 동일한 블록에 삼각형 파티션 모드가 적용될 수 있다. 하지만, 이 블록에는 기하학적 파티션 모드가 적용될 수 없다.

[0131] 2. 삼각형 파티션 모드에서 코딩된 루마 샘플의 가중치의 계산에 사용된 식은 기하학적 파티션 모드에서 루마 샘플 가중치 계산과 상이하다. 더욱이, 삼각형 파티션 모드로 코딩된 크로마 샘플의 가중치는 별도로 계산되는 한편, 기하학적 파티션 모드로 코딩된 크로마 샘플의 가중치는 대응하는 루마 샘플로부터 서브샘플링된다.

[0132] 3. 삼각형 파티션 또는 기하학적 파티션 모드로 코딩된 블록의 움직임 벡터는 모두 4×4 서브블록에 저장된다. 그리고 각 4×4 서브블록의 위치에 의존하여, 단방향-예측 또는 양방향-예측 움직임 벡터가 저장된다. 하지만, 4×4 서브블록에 대해 저장될 단방향-예측 또는 양방향-예측 움직임 벡터 중 하나를 선택하는 프로세스는 삼각형 파티션 모드 및 기하학적 파티션 모드와 상이하다.

[0133] 4. 삼각형 파티션 모드의 경우 SBT 모드가 허용되지 않지만, 이는 기하학적 파티션 모드에 적용될 수 있다.

[0134] 삼각형 파티션 모드가 기하학적 파티션 모드의 서브세트로 볼 수 있으므로, 이는 삼각형 파티션 및 기하학적 파티션 모드에서 사용되는 프로세스 모두를 통합할 수 있다.

[0135] 삼각형 파티션 및 기하학적 파티션 모드의 신택스를 통합하기 위해, 이는 블록이 두 개의 파티션 내로 분할되는지를 나타내기 위해 하나의 플래그만을 사용할 수 있다. 블록의 크기가 64개의 루마 샘플 이상일 때, 플래그는 시그널링된다. 플래그가 참(true)일 때, 파티션 모드 인덱스는 블록을 분할하는 데 어느 파티션 방법이 사용되는 지를 나타내기 위해 더 시그널링된다.

[0136] 일 실시예에서, 서브블록 병합, 정규 병합 및 MMVD 모드를 사용하여 블록이 코딩되지 않을 때 플래그(예를 들어, 도 19-20에서의 CIIP 플래그)가 시그널링된다.

[0137] 다른 실시예에서, 플래그(예를 들어, 도 21에서의 삼각형/기하학적 플래그)는 병합 신택스 구조의 시작 부분에서 시그널링된다.

[0138] 블록이 두 개의 파티션으로 분할될 때, 파티션 모드 인덱스는 어느 파티션 방법이 사용되는지를 나타내기 위해 더 시그널링된다.

[0139] 일 실시예에서, 두 개의 삼각형 파티션 모드(예를 들어, 블록을 좌측-상단 코너로부터 우측-하단 코너로 또는 우측 상단 코너로부터 좌측 하단 코너로 분할하는 것)는 파티션 모드 목록의 앞에 있으며, 기하학적 파티션 모드가 후속한다. 다시 말해, 파티션 모드 인덱스가 0 또는 1과 동일할 때, 삼각형 파티션 모드를 사용하여 블록이 분할된다. 그렇지 않으면, 블록은 기하학적 파티션 모드를 사용하여 분할된다.

[0140] 다른 실시예에서, 삼각형 파티션 모드는 도 22a의 테이블 22A에 도시된 바와 같이 기하학적 파티션 모드 중 하나로 처리된다. 예를 들어, 파티션 모드 인덱스가 19와 동일할 때, 블록은 좌측-상단 코너로부터 우측-하단 코너로 분할된다. 다른 예시로서, 파티션 모드 인덱스(58)는 블록이 우측 상단 코너로부터 좌측 하단 코너로 분할

되었음을 나타낸다.

[0141] 또 다른 실시예에서, 삼각형 파티션 모드는 또한 도 22b의 테이블 22B에 도시된 바와 같이, 기하학적 파티션 모드 중 하나로 처리된다. 파티션 모드 인덱스가 10과 동일할 때, 블록은 좌측-상단 코너로부터 우측-하단 코너로 분할된다. 더욱이, 파티션 모드 인덱스가 24라는 것은 블록이 우측-상단 코너로부터 좌측-하단 코너로 분할되었음을 나타낸다.

[0142] 블록에 대한 파티션 모드의 수는 블록 크기 및/또는 블록 형태에 의존할 수 있다는 것이 유의된다.

[0143] 일 실시예에서, 그의 폭 또는 높이가 4와 동일하거나 또는 max(폭, 높이)/min(폭, 높이)의 비율이 4보다 큰 경우 블록에 대해 두 개의 파티션 모드만이 허용된다. 그렇지 않으면 142개의 파티션 모드가 허용된다.

[0144] 다른 실시예에서, 블록의 크기가 임계치보다 클 때, 파티션 모드의 수가 감소된다. 예를 들어, 블록의 크기가 1024개의 루마 샘플보다 큰 경우 24개의 양자화된 각도 및 4개의 양자화된 거리만이 허용된다.

[0145] 또 다른 실시예에서, 블록 형태가 좁고-길거나 또는 넓고-평평할 때, 파티션 모드의 수가 감소된다. 예를 들어, max(폭, 높이)/min(폭, 높이)의 비율이 2보다 클 때, 24개의 양자화된 각도 및 4개의 양자화된 거리만이 허용된다. 또한, 도 23에 도시된 바와 같이, 더 큰 치수를 따라 블록을 파티셔닝하는 각도만이 허용될 수 있다. 도 23에 도시된 바와 같이, 폭이 넓고 평평한 블록의 경우, 점선으로 도시된 세 개의 각도는 허용되지 않고, 실선으로 표시된 각도만이 허용된다.

[0146] 또 다른 실시예에서, 블록의 크기가 임계치보다 크고 블록 형태가 좁고 길거나 또는 넓고 평평할 때, 파티션 모드의 수가 감소될 수 있다. 예를 들어, 블록의 크기가 1024개의 루마 샘플보다 크고 max(폭, 높이)/min(폭, 높이)의 비율이 2보다 클 때, 24개의 양자화된 각도 및 4개의 양자화된 거리만이 허용된다. 이는 도 23에 도시된 제한과 더 결합될 수 있다.

[0147] 각도 인덱스 및 거리 인덱스에 대한 룩업 테이블이 변경될 수 있다.

[0148] 일 실시예에서, 파티션 모드 인덱스는 도 24의 테이블 24에 도시된 바와 같이, 각도 인덱스 제1 순서(angle index first order) 대신에 거리 인덱스 제1 순서(distance index first order)로 사용된다.

[0149] 다른 실시예에서, 파티션 모드 인덱스 순서는 파티셔닝 방법의 발생에 관련된다. 더 높은 발생 파티셔닝 방법은 룩업 테이블의 앞에 있다. 예시는 도 25의 테이블 25에 도시된다. 거리 0을 갖는 각도 인덱스 0, 4, 8 및 12는 블록을 파티셔닝 하는데 사용될 더 높은 확률을 갖는다.

[0150] 이전에 언급된 바와 같이, 삼각형 파티션 모드는 크기가 64개의 루마 샘플 이상인 블록에 적용될 수 있다. 하지만, 기하학적 파티션 모드는 폭 및 높이가 모두 8 이상이고 max(폭, 높이)/min(폭, 높이)의 비율이 4 이하인 블록에 적용될 수 있다. 이는 삼각형 파티션 및 기하학적 파티션 모드의 블록 크기 및 블록 형태에 대한 제한을 통합할 수 있다.

[0151] 일 실시예에서, 삼각형 파티션 모드 및 기하학적 파티션 모드 모두는 폭 및 높이가 모두 8 이상이고 max(폭, 높이)/min(폭, 높이)의 비율이 4 이하인 블록에 적용될 수 있다.

[0152] 다른 실시예에서, 삼각형 파티션 모드 및 기하학적 파티션 모드 모두는 크기가 64개의 루마 샘플 이상인 블록에 적용될 수 있다.

[0153] 또 다른 실시예에서, 삼각형 파티션 모드 및 기하학적 파티션 모드는 크기가 64개의 루마 샘플 이상이고 max(폭, 높이)/min(폭, 높이)의 비율이 4 이하인 블록에 적용될 수 있다.

[0154] 본 개시의 실시예는 또한, 삼각형 파티션 및 기하학적 파티션 모드의 가중치 계산 프로세스를 통합하는 방법을 제공한다.

[0155] 일 실시예에서, 삼각형 파티션의 루마 샘플에 대한 가중치 계산 프로세스는 다음 두 가지 수정으로 기하학적 파티션 모드(위의 블렌딩 프로세스에서 설명됨)에서 사용되는 프로세스로 대체된다.

[0156] 1. *Dis*[]에서의 값은 도 26의 테이블 26에서의 값으로 대체된다.

[0157] 2. wIdx=log2(CuW)-2 및 hIdx=log2(CuH)-2.

[0158] 덧붙여, 삼각형 파티션 모드로 코딩된 블록에 대해, 블록이 좌측-상단 코너로부터 우측-하단 코너로 분할되는 경우, *angleIdx* 및 *distanceIdx*는 각각 4 및 0으로 설정된다. 그렇지 않으면(예를 들어, 블록이 우측-상단 코너

로부터 좌측-하단 코너로 분할됨), *angleIdx* 및 *distanceIdx*는 각각 12 및 0으로 설정된다.

[0159] 다른 실시예에서, 삼각형 파티션 모드 및 기하학적 파티션 모드 모두에 대해, 크로마 샘플에 대한 가중치는 각 2x2 루마 서브블록의 좌측-상단 루마 샘플의 가중치로부터 서브샘플링된다.

[0160] 또 다른 실시예에서, 삼각형 파티션 모드 및 기하학적 파티션 모드 모두에 대해, 크로마 샘플에 대한 가중치는 기하학적 파티션 모드에서 코딩된 루마 샘플에 대해 사용된 것과 동일한 프로세스를 사용하여 계산된다.

[0161] 본 개시에서, 이는 또한, 삼각형 파티션 모드 및 기하학적 파티션 모드에서 사용되는 움직임 필드 저장의 프로세스를 통합할 수 있다.

[0162] 일 실시예에서, 삼각형 파티션 모드에 대한 움직임 필드 저장은 기하학적 파티션 모드에 대한 것으로 대체된다. 즉, 4×4 서브블록의 4개의 코너에 위치된 4개의 루마 샘플의 가중치가 합산된다. 합이 threshold2보다 작고 threshold1보다 큰 경우, 이 4×4 서브블록에 대해 양방향-예측 움직임 벡터가 저장될 수 있다. 그렇지 않으면, 단방향-예측 움직임 벡터가 저장된다. threshold1 및 threshold2는 각각 $32 \gg (((log2(CuW)+log2(CuH)) \gg 1)-1)$ 및 $32-threshold1$로 설정된다.

[0163] 다른 실시예에서, 삼각형 파티션 모드 또는 기하학적 파티션 모드를 사용하여 코딩된 블록에 대해, 각 루마 샘플의 가중치가 검사된다. 루마 샘플의 가중치가 0 또는 8과 동일하지 않은 경우, 루마 샘플은 가중된 샘플로 보인다. 4×4 서브블록에서의 모든 루마 샘플이 가중되는 경우, 4×4 서브블록에 대한 양방향-예측 움직임이 저장된다. 그렇지 않으면. 단방향-예측 움직임이 저장된다.

[0164] SBT와 기하학적 모드 사이의 상호작용을 SBT와 삼각형 파티션 모드 사이의 상호 작용과 조화시키기 위해, 이는 본 개시에서의 기하학적 파티션 모드에 대해 SBT를 비활성화(disable)시킬 수 있다. SBT 및 기하학적 파티션 모드의 조합은 블록 내에서 두 개의 교차 경계를 생성할 수 있으며, 이는 주관적인 품질 문제를 야기할 수 있다.

[0165] 일 실시예에서, 도 27의 테이블 27에 도시된 바와 같이, 기하학적 파티션이 사용될 때 cu_sbt_flag는 시그널링되지 않는다. 관련된 신택스는 도 27에서 이탤릭체로 되어 있고, 회색으로 강조되어 있다.

[0166] 다른 실시예에서, SBT 파티션 모드의 일부는 GEO 파티션 모드에 의존하여 비활성화된다. SBT 파티셔닝 엣지가 GEO 파티셔닝 엣지와 교차될 때, 이 SBT 파티션 모드는 허용되지 않는다. 그렇지 않으면 SBT 파티션 모드가 허용된다. 도 28은 SBT 파티션 모드 및 GEO 파티션 모드의 일 예시를 도시한다. 또한, 각도 및 거리 인덱스는 GEO 파티셔닝 엣지와 SBT 파티셔닝 엣지 사이에 교차점이 있는지를 결정하는 데 사용될 수 있다. 일 예시에서, 현재 블록의 각도 인덱스가 0일 때(즉, 수직 파티셔닝 엣지), 수평 SBT 파티셔닝은 현재 블록에 적용될 수 없다. 더욱이, SBT 신택스는 변경이 이탤릭 및 회색으로 강조된, 도 29의 테이블 29로 수정될 수 있다.

[0167] 기하학적 파티션 모드는 블록을 두 개의 기하학적 형태 파티션으로 분할하고, 각 기하학적 파티션은 그 자신의 움직임 벡터로 움직임 보상을 수행한다. 기하학적 파티션 모드는 인터 예측의 예측 정확도를 개선한다. 하지만, 이는 다음의 양상에서 복잡할 수 있다.

[0168] 제1 양상에서, 기하학적 파티셔닝 서브모드의 총 수는 상당히 많다. 그러므로, 실제 구현에서 블렌딩 가중치 및 움직임 필드 저장을 위한 마스크 모두를 저장하는 것은 불가능하다. 140개의 서브모드의 경우 마스크를 저장하는 데 필요한 총 비트 수는 다음과 같다:

[0169] - 블렌딩 가중치에 대해: $(8\times8+8\times16+8\times32+8\times64+16\times8+16\times16+16\times32+16\times64+32\times8+32\times16+32\times32+32\times64+64\times8+64\times16+64\times32+64\times64+64\times128+128\times64+128\times128)\times140\times4=26{,}414{,}080$ *비트* = $3{,}301{,}760$ *바이트* ≅ 3.3 *메가바이트*
- 움직임 필드 저장에 대해: $(2\times2+2\times4+2\times8+2\times16+4\times2+4\times4+4\times8+4\times16+8\times2+8\times4+8\times8+8\times16+16\times2+16\times4+16\times8+16\times16+16\times32+32\times16+32\times32)\times140\times2=825{,}440$ *비트* = $103{,}180$ *바이트* ≅ 103 *킬로바이트*

	140 파티셔닝 서브모드	108 파티셔닝 서브모드	80 파티셔닝 서브모드	64 파티셔닝 서브모드
블렌딩 가중치	3,301,760 바이트	2,547,072 바이트	1,886,720 바이트	1,509,376 바이트
움직임 저장	103,180 바이트	79,596 바이트	58,960 바이트	47,168 바이트
총계	3,404,940 바이트	2,626,668 바이트	1,945,680 바이트	1,556,544 바이트

[0170]

[0171] 제2 양상에서, 이를 저장하는 대신에, 마스크가 즉각적으로(on the fly) 계산되면, 계산 복잡성이 증가된다. 가중치 및 움직임 필드 저장을 블렌딩하기 위한 마스크를 계산하기 위한 식은 복잡하다. 더 구체적으로, 곱셈(×), 시프트(<<), 덧셈(+), 비교 연산의 수가 상당히 많다. 크기가 W×H의 블록을 가정하면, 블록당 각 연산의 수는 다음과 같다:

[0172]
- 곱셈: $5+2\times W\times H+2\times(W\times H/16)$;
- 시프트 $4+3\times W\times H+2\times(W\times H/16)$;
- 덧셈: $8+6\times W\times H+5\times(W\times H/16)$;
- 비교: $4+2\times W\times H+2\times(W\times H/16)$.

[0173] 세부사항은 다음의 테이블에 열거된다.

식	×	<<, >>	+	비교
블록 마다	**5**	**4**	**8**	**4**
$displacementX = angleIdx$	0	0	0	0
$displacementY = (displancementX + NumAngles \gg 2)\%NumAngles$	1	1	1	0
$rho = distanceIdx \times (stepSize \ll scaleStep) + Dis[displacementX] \ll wIdx + Dis[displacementY] \ll hIdx$	1	3	2	0
$stepSize = stepDis + 77$	0	0	1	0
$scaleStep = (wIdx \geq hIdx)\ ?\ hIdx - 3 : wIdx - 3$	0	0	2	1
$wIdx = log2(CuW)$	0	0	0	0
$hIdx = log2(CuH)$	0	0	0	0
$whRatio = (wIdx \geq hIdx)\ ?\ (wIdx - hIdx) : (hIdx - wIdx)$	0	0	1	1
$angleN = \begin{cases} angleIdx\ \%\ 8 & wIdx \geq hIdx \ \&\&\ (angleIdx \gg 3)\%\ 1 = 1 \\ 8 - (angleIdx\ \%\ 8) & wIdx \geq hIdx \ \&\&\ (angleIdx \gg 3)\%\ 1 = 0 \\ 8 - (angleIdx\ \%\ 8) & wIdx < hIdx \ \&\&\ (angleIdx \gg 3)\%\ 1 = 1 \\ angleIdx\ \%\ 8 & wIdx < hIdx \ \&\&\ (angleIdx \gg 3)\%\ 1 = 0 \end{cases}$	1	0	1	2
$rho_{subblk} = 3 \times Dis[displacementX] + 3 \times Dis[displacementY]$	2	0	0	0
루마 샘플 마다	**2**	**3**	**6**	**2**
$distFromLine = \left((x \ll 1) + 1\right) \times Dis[displacementX] + \left((y \ll 1) + 1\right) \times Dis[displacementY] - rho$	2	2	4	0
$distScaled = Min((abs(distFromLine) + 4) \gg 3.26)$	0	1	1	1
$sampleWeight_L[x][y] = distFromLine \leq 0\ ?\ GeoFilter[distScaled] : 8 - GeoFilter[distScaled]$	0	0	1	1
4×4 서브블록 마다	**2**	**2**	**5**	**2**
$distFromLine_{subblk} = \left((x_{subblk} \ll 3) + 1\right) \times Dis[displacementX] + \left((y_{subblk} \ll 3) + 1\right) \times Dis[displacementY] - rho + rho_{subblk}$	2	2	5	0
$motionMask[x_{subblk}][y_{subblk}] = abs(distFromLine_{subblk}) < 256\ ?\ 2 : (distFromLine_{subblk} \leq 0\ ?\ 0 : 1)$	0	0	0	2

[0174]

[0175] 더욱이, Dis[], GeoFilter[], stepDis[] 및 *angleIdx* 및 *distanceIdx*에 대한 룩업 테이블인, 네 개의 미리 계산된 테이블을 저장하기 위해 메모리가 여전히 필요하다. 각 테이블의 크기는 다음과 같이 열거된다.

테이블	크기
Dis[]	32×8 = 256 비트
GeoFilter[]	27×4 = 108 비트
stepDis[]	45×11 = 495 비트
angleIdx 및 distanceIdx 에 대한 룩업 테이블	140 모드: 140× (5 + 3) = 1120 비트 108 모드: 108× (5 + 2) = 756 비트 80 모드: 80× (5 + 2) = 560 비트 64 모드: 64× (5 + 2) = 448 비트

[0176]

[0177] 제3 양상에서, 현재 기하학적 파티션 모드 설계에서, 45° /135° 및 distanceIdx 0의 조합은 VVC에서의 삼각형 파티션 모드가 이 파티셔닝 옵션을 지원한다고 가정하기 때문에 항상 허용되지 않는다. 하지만, 사각형이 아닌 블록의 경우, 삼각형 파티션 모드에서 파티션 각도는 다음의 테이블에 도시된 바와 같이 45° 또는 135° 가 아니다. 따라서, 정사각형이 아닌 블록에 대해 이 두 개의 파티셔닝 각도를 제외하는 것은 의미가 없다.

폭 : 높이 비율	1 : 1	1 : 2	1 : 4	1 : 8	1 : 16
삼각형 각도 1	135°	116.5°	104°	97.1°	93.6°
삼각형 각도 2	45°	63.5°	76°	82.9°	86.4°
폭 : 높이 비율		2 : 1	4 : 1	8 : 1	16 : 1
삼각형 각도 1		153.4°	166°	172.9°	176.4°
삼각형 각도 2		26.6°	14°	7.1°	3.6°

[0178]

[0179] 제4 양상에서, 각도 및 거리의 일부 조합이 distanceIdx 0을 통한 수평 분할 또는 distanceIdx 0을 통한 수직 분할과 같은, 기하학적 파티션 모드에서 지원되지 않기 때문에(이는 이진 트리 파티션과의 리던던시를 회피하기 위함임), 룩업 테이블은 각 기하학적 파티셔닝 서브모드에 대한 각도 및 거리를 도출하는 데 사용된다. 각도와 거리의 조합의 제한이 제거되는 경우, 룩업 테이블은 필요하지 않을 수 있다.

[0180] 제5 양상에서, 삼각형 파티션 모드 및 기하학적 파티션 모드에서 사용되는 블렌딩 프로세스, 움직임 필드 저장 및 신택스 구조가 통합되지 않으며, 이는 소프트웨어 및 하드웨어 구현 모두에서 이 두 개의 모드에 대해 두 개의 종류의 로직이 요구된다는 것을 의미한다. 게다가, 삼각형 모드에 대한 마스크를 저장하는 데 필요한 비트의 총 수는 다음과 같다:

- 블렌딩 가중치에 대해: $(4\times16+4\times32+4\times64+8\times8+8\times16+8\times32+8\times64+16\times4+16\times8+16\times16+16\times32+16\times64+32\times4+32\times8+32\times16+32\times32+32\times64+64\times4+64\times8+64\times16+64\times32+64\times64+64\times128+128\times64+128\times128)\times2\times4=384{,}512$ *비트* $=48{,}064$ *바이트* $\cong$ 48 *킬로바이트*
- 움직임 필드 저장에 대해: $(1\times4+1\times8+1\times16+2\times2+2\times4+2\times8+2\times16+4\times1+4\times2+4\times4+4\times8+4\times16+8\times1+8\times2+8\times4+8\times8+8\times16+16\times1+16\times2+16\times4+16\times8+16\times16+16\times32+32\times16+32\times32)\times2\times2=12{,}016$ *비트* $=1{,}502$ *바이트* $\cong 1.5$ *킬로바이트*

140 기하학적 파티셔닝 서브모드 + 삼각형 모드	3,454,506 바이트
108 기하학적 파티셔닝 서브모드 + 삼각형 모드	2,676,234 바이트
80 기하학적 파티셔닝 서브모드 + 삼각형 모드	1,995,246 바이트
64 기하학적 파티셔닝 서브모드 + 삼각형 모드	1,606,110 바이트

[0181]

[0182] 전술한 문제를 해결하기 위해, 수개의 해결책이 제안된다.

[0183] 제1 해결책은 기하학적 파티션 모드의 단순화에 관련된다.

[0184] 블렌딩 가중치 및 움직임 필드 저장을 위한 마스크를 즉각적으로 계산하는 것을 회피하기 위해, 크기가 256×256 또는 64×64인 수 개의 미리 계산된 마스크로부터의 각 블록에 대한 마스크를 도출하는 것이 제안된다. 제안된 방법은 마스크를 저장하기 위한 메모리를 감소시킬 수 있다.

[0185] 제안된 크롭된 방법(cropped method)에서, 마스크의 제1 세트 및 제2 세트가 미리 정의된다. 마스크의 제1 세트 $g_sampleWeight_L[]$은 수 개의 마스크를 포함할 수 있으며, 마스크의 각각은 256×256의 크기를 갖고, 마스크는 각 블록에 대한 블렌딩 가중치를 도출하는 데 사용된다. 마스크의 제2 세트인 $g_motionMask[]$는 수 개의 마스크를 포함할 수 있으며, 마스크의 각각은 64×64의 크기를 갖고, 마스크는 각 블록에 대한 움직임 필드 저장을 위한 마스크를 도출하는 데 사용된다. 제1 세트 및 제2 세트에서 마스크의 수는 기하학적 파티셔닝 서브모드의 수에 의존한다. 상이한 크기의 블록의 경우, 그의 마스크는 제1 세트 및 제2 세트에서의 마스크 중 하나로부터 크롭된다.

[0186] 일 실시예에서, 제1 세트 및 제2 세트에서의 미리 정의된 마스크는 도 14 및 15a-15f를 참조하여, 그리고 움직임 필드 저장에 대해 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서 마스크의 수는 모두 N이며, 여기서 N은 기하학적 파티션 모드에서 지원되는 각도의 수로 설정된다. 제1 세트 및 제2 세트에서 인덱스 n을 갖는 n번째 마스크는 각도 n의 마스크를 나타내며, 여기서 n은 0 내지 N-1의 범위에 있다.

[0187] 일 예시에서, 기하학적 파티셔닝 서브모드의 수가 140, 즉 16개의 각도에 9개의 거리를 더한 것으로 설정될 때, 변수 N은 16으로 설정된다. 다른 예시에서, 기하학적 파티셔닝 서브모드의 수가 108, 즉 16개의 각도에 7개의 거리를 더한 것으로 설정될 때, 변수 N은 16으로 설정된다. 다른 예시에서, 기하학적 파티셔닝 서브모드의 수가 각각 80(12개의 각도 + 7개의 거리) 및 64(10개의 각도 + 7개의 거리)로 설정될 때 변수 N은 12 및 10으로 설정된다.

[0188] 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0189] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 도 15c의 테이블 15C, 도 15d의 테이블 15D, 도 22b의 테이블 22B, 도 30의 테이블 30 및 도 31의 테이블 31에 도시된다.

[0190] - 변수 offsetX 및 offsetY는 다음과 같이 계산된다.

[0191] $$offsetX = \begin{cases} (256 - W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256 - W) \gg 1) + A < N \,?\, (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

[0192] $$offsetY = \begin{cases} ((256 - H) \gg 1) + A < N \,?\, (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256 - H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0193] - $sampleWeight_L[x][y] = g_sampleWeight_L[A \,\%\, N][x + offsetX][y + offsetY]$

[0194] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, 각 대응하는 2x2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치는 YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로 사용된다.

[0195] 더욱이, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다.

[0196] - 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산된다:

[0197] $$offsetXmotion = \begin{cases} (64 - (W \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64 - (W \gg 2)) \gg 1) + A < N \,?\, (D \times W) \gg 5 : -((D \times W) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

[0198] $$offsetYmotion = \begin{cases} ((64 - (H \gg 2)) \gg 1) + A < N \,?\, (D \times H) \gg 5 : -((D \times H) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64 - (H \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0199] - $motionMask[x_{subblk}][y_{subblk}] = g_motionMask[A \,\%\, N][x_{subblk} + offsetXmotion][y_{subblk} + offsetYmotion]$

[0200] 미리 정의된 마스크를 저장하는 데 필요한 비트의 수는 다음과 같이 열거된다:

[0201] - 블렌딩 가중치에 대해: $(256 \times 256) \times 16 \times 4 = 4{,}193{,}304$ 비트 = 524,288 바이트 ≅ 524 킬로바이트

[0202] - 움직임 필드 저장에 대해: $(64 \times 64) \times 16 \times 2 = 131{,}072$ 비트 = 16,384 바이트 ≅ 16 킬로바이트

[0203]

	140 파티셔닝 서브모드	108 파티셔닝 서브모드	80 파티셔닝 서브모드	64 파티셔닝 서브모드
블렌딩 가중치	524,288 바이트	524,288 바이트	393,216 바이트	327,680 바이트
움직임 저장	16,384 바이트	16,384 바이트	12,288 바이트	10,240 바이트
총계	540,672 바이트	540,672 바이트	405,504 바이트	337,920 바이트
원본 기하학적 설계에 비해 감소%	84.1%	79.4%	79.2%	78.3%

[0204] 더욱이, 마스크는 다음과 같이 단순화된 식을 사용하여 즉각적으로 계산될 수 있다.

$$distFromLine = \left(((x + offsetX) \ll 1) + 1\right) \times Dis[displacementX] + \left(((y + offsetY) \ll 1) + 1\right) \times Dis[displacementY] - Rho[displacementX]$$

$$distScaled = Min((abs(distFromLine) + 4) \gg 3,26)$$

$$sampleWeight_L[x][y] = distFromLine \leq 0\ ?\ GeoFilter[distScaled] : 8 - GeoFilter[distScaled]$$

[0205]

[0206] 여기서 (x,y)는 각 루마 샘플의 위치를 나타내고, *Dis*[] 및 *GeoFilter*[]는 각각 테이블 12 및 테이블 13에 도시된 두 개의 룩업 테이블이다. *Rho*[]는 도 32의 테이블 32에 도시된 룩업 테이블이다.

[0207] *displacementX* 및 *displacementY* 파라미터는 다음과 같이 계산된다:

$$displacementX = angleIdx\ \%\ 16$$

$$displacementY = (displancementX + NumAngles \gg 2)\%NumAngles$$

[0208]

[0209] 여기서 *NumAngles*는 32로 설정된다. *angleIdx*는 기하학적 파티셔닝 서브모드의 총 수가 각각 140, 108, 80 및 64로 설정될 때, 테이블 15C, 테이블 15D, 테이블 15E 및 테이블 15F로부터 도출된다. angleIdx는 또한, 테이블 22B, 테이블 30 및 테이블 31에 도시된 룩업 테이블로부터 도출될 수 있다.

$$distFromLine_{subblk} = \left(((x_{subblk} + offsetXmotion) \ll 3) + 1\right) \times Dis[displacementX] + \left(((y_{subblk} + offsetYmotion) \ll 3) + 1\right) \times Dis[displacementY] - Rho_{subblk}[displacementX]$$

$$motioMask[x_{subblk}][y_{subblk}] = abs(distFromLine_{subblk}) < 256\ ?\ 2 : (distFromLine_{subblk} \leq 0\ ?\ 0 : 1)$$

[0210]

[0211] 크기가 W×H의 블록을 가정하면, 블록당 각 연산의 수는 다음과 같다:

- 곱셈: 4+2×W×H+2×(W×H/16);
- 시프트: 9+3×W×H+2×(W×H/16);
- 덧셈: 7+8×W×H+6×(W×H/16);
- 비교: 6+2×W×H+2×(W×H/16).

[0212]

[0213] 블록당 각 연산의 수에 대한 세부사항은 도 33의 테이블 33에서 열거된다.

[0214] 메모리는 Dis[], GeoFilter[], Rho[], Rho_{subblk}[], 및 angleIdx 및 distanceIdx에 대한 룩업 테이블인 다섯 개의 미리-계산된 테이블을 저장하는 데 필요하다. Rho_{subblk}[]에 대한 룩업 테이블은 도 34에 예시된다. 각 테이블의 크기는 다음과 같이 열거된다.

테이블	크기
Dis[]	32×8 = 256 비트
GeoFilter[]	27×4 = 108 비트
Rho[]	16×16 = 256 비트
Rho_{subblk}[]	16×16 = 256 비트
angleIdx 및 distanceIdx 에 대한 룩업 테이블	140 모드: 140× (5 + 3) = 1120 비트 108 모드: 108× (5 + 2) = 756 비트 80 모드: 80× (5 + 2) = 560 비트 64 모드: 64× (5 + 2) = 448 비트

[0215]

[0216] 제안된 방법의 계산적인 복잡도는 원본의 기하학적 설계의 것과 유사하다. 더욱 구체적으로, 원본 기하학적 설계와 비교한다.

[0217] - W×H 블록에 대해 곱셈 연산의 수가 1 증가된다.

[0218] - W×H 블록에 대해 시프트 연산의 수가 5 증가된다.

[0219] - W×H 블록에 대해 비교 연산의 수가 2 증가된다.

[0220] - 덧셈 연산의 수는 W×H 블록에 대해 2×W×H+(W×H/16)-1만큼 증가된다.

[0221] - 메모리 사용은 17비트 증가된다.

[0222] 마스크를 계산하는 데 사용되는 식은 다음과 같이 더 단순화할 수 있다:

[0223] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0224] - 변수 N(제1 세트 및 제2 세트에서 마스크의 수)은 테이블 15C, 테이블 15D 및 테이블 22B를 사용하여 angleIdx 및 distanceIdx가 도출될 때 16으로 설정된다. 다른 한편으로, 각각 테이블 30 및 테이블 31을 사용하여 angleIdx 및 distanceIdx가 도출될 때, 변수 N은 12 및 10으로 설정된다.

[0225] - 변수 offsetX 및 offsetY는 다음과 같이 계산된다:

[0226] $$offsetX = \begin{cases} (-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((-W) \gg 1) + A < N\ ?\ (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

[0227] $$offsetY = \begin{cases} ((-H) \gg 1) + A < N\ ?\ (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0228] - 변수 displacementX 및 displacementY는 다음과 같이 계산된다:

[0229] $$displacementX = angleIdx$$

$$displacementY = (displancementX + NumAngles \gg 2)\%NumAngles$$

[0230] 여기서 *NumAngles*는 32로 설정된다.

[0231] - 위치 (x, y)에 위치된 루마 샘플에 대한 가중치는 다음과 같이 계산된다:

$$weightIdx = (((x + offsetX) << 1) + 1) * disLut[displacementX] + (((y + offsetY) << 1) + 1)) * disLut[displacementY]$$

$$partFlip = (angleIdx >= 13 \&\& angleIdx <= 27) ? 0 : 1$$

$$weightIdxL = partFlip ? 32 + weightIdx : 32 - weightIdx$$

[0232] $$sampleWeight_L[x][y] = Clip3(0, 8, (weightIdxL + 4) \gg 3)$$

[0233] 여기서 $disLut[]$은 다음과 같은 룩업 테이블이다:

angleIdx	0	2	3	4	5	6	8	10	11	12	13	14
disLut[idx]	8	8	8	4	4	2	0	−2	−4	−4	−8	−8
angleIdx	16	18	19	20	21	22	24	26	27	28	29	30
disLut[idx]	−8	−8	−8	−4	−4	−2	0	2	4	4	8	8

[0234]

[0235] 다른 실시예에서, 제1 세트 및 제2 세트에서 미리 정의된 마스크는 도 14 및 도 15a-15f를 참조하여, 그리고 움직임 필드 저장에 대해 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서의 마스크 수는 모두 $N_{reduced}$이고, 여기서 $N_{reduced}=(N\gg1)+1$이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. 일 예시에서, 기하학적 파티셔닝 서브모드의 수가 140으로 설정될 때, 변수 $N_{reduced}$는 9로 설정된다. 즉, 16개의 각도에 9개의 거리를 더한 것이다. 다른 예시에서, 기하학적 파티셔닝 서브모드의 수가 각각 108(16개의 각도 + 7개의 거리), 80(12개의 각도 + 7개의 거리) 및 64(10개의 각도 + 7개의 거리)로 설정될 때, 변수 $N_{reduced}$는 9, 7 및 6으로 설정된다.

[0236] 0과 $N_{reduced}$-1 사이의 각도에 대해, 그 마스크는 제1 세트 및 제2 세트에서 마스크로부터 직접적으로 크롭된다. 다른 한편으로, $N_{reduced}$와 N-1 사이의 각도에 대한 마스크는 제1 세트 및 제2 세트에서의 마스크로부터 크롭되고 수평 방향으로 플립된다(flipped). 본 개시와 일치하는, 도 35a는 각도 135° 의 마스크에 대한 예시를 도시하고, 도 35b는 각도 45° 의 마스크에 대한 예시를 도시한다.

[0237] 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0238] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0239] - 변수 offsetX 및 offsetY는 다음과 같이 계산될 수 있다.

[0240]
$$offsetX = \begin{cases} (256 - W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256 - W) \gg 1) + A < N\ ?\ (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

[0241]
$$offsetY = \begin{cases} ((256 - H) \gg 1) + A < N\ ?\ (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256 - H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0242] - $$sampleWeight_L[x][y] = \begin{cases} g_sampleWeight_L[A\ \%\ N][x + offsetX][y + offsetY] & A\ \%\ N < N_{reduced} \\ g_sampleWeight_L[N - A\ \%\ N][W - 1 - x + offsetX][y + offsetY] & \text{그렇지 않은 경우} \end{cases}$$

[0243] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0244] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0245] 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산된다:

[0246] $$offsetXmotion = \begin{cases} (64-(W \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64-(W \gg 2)) \gg 1) + A < N\ ?\ (D \times W) \gg 5 : -((D \times W) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

[0247] $$offsetYmotion = \begin{cases} ((64-(H \gg 2)) \gg 1) + A < N\ ?\ (D \times H) \gg 5 : -((D \times H) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64-(H \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0248] $$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A\ \%\ N][x_{subblk}-offsetXmotion][y_{subblk}+offsetYmotion] & A\ \%\ N < N_{reduced} \\ g_motionMask[N-A\ \%\ N][(W \gg 2)-1-x_{subblk}+offsetXmotion][y_{subblk}+offsetYmotion] & \text{그렇지 않은 경우} \end{cases}$$

[0249] 미리 정의된 마스크를 저장하는 데 필요한 비트의 수는 다음과 같이 열거된다:

[0250]
- 블렌딩 가중치에 대해:$(256 \times 256) \times 9 \times 4 = 2{,}359{,}296$ 비트 = 294,912 바이트 ≅ 295 킬로바이트
- 움직임 필드 저장에 대해:$(64 \times 64) \times 9 \times 2 = 131{,}072$ 비트 = 16,384 바이트 ≅ 16 킬로바이트

[0251]

	140 파티셔닝 서브모드	108 파티셔닝 서브모드	80 파티셔닝 서브모드	64 파티셔닝 서브모드
블렌딩 가중치	294,912 바이트	294,912 바이트	229,376 바이트	196,608 바이트
움직임 저장	9,216 바이트	9,216 바이트	7,168 바이트	6,144 바이트
총계	304,128 바이트	304,128 바이트	236,544 바이트	202,752 바이트
원본 기하학적 설계에 비해 감소%	91.1%	88.4%	87.8%	87.0%

[0252] 제3 실시예에서, 제1 세트 및 제2 세트에서 미리 정의된 마스크는 위에서 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서의 마스크 수는 모두 $N_{reduced}$이고, 여기서 $N_{reduced}=(N \gg 1)+1$이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. 0과 $N_{reduced}-1$ 사이의 각도에 대해, 그 마스크는 제1 세트 및 제2 세트에서 마스크로부터 직접적으로 크롭된다. 다른 한편으로, $N_{reduced}$와 N-1 사이의 각도에 대한 마스크는 제1 세트 및 제2 세트에서의 마스크로부터 크롭되고 수직 방향으로 플립된다. 본 실시예와 일치하는, 도 36은 각도 135° 의 마스크에 대한 예시를 도시하고, 도 36b는 각도 45° 의 마스크에 대한 예시를 도시한다.

[0253] 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0254] 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블에서 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0255] 변수 offsetX 및 offsetY는 다음과 같이 계산될 수 있다.

$$offsetX = \begin{cases} (256 - W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256 - W) \gg 1) + A < N\ ?\ (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256 - H) \gg 1) + A < N\ ?\ (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256 - H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

- $sampleWeight_L[x][y] = \begin{cases} g_sampleWeight_L[A\ \%\ N][x + offsetX][y + offsetY] & A\ \%\ N < N_{reduced} \\ g_sampleWeight_L[N - A\ \%\ N][x + offsetX][H - 1 - y + offsetY] & \text{그렇지 않은 경우} \end{cases}$

[0256]

[0257] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0258] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0259] 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산될 수 있다.

$$offsetXmotion = \begin{cases} (64 - (W \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64 - (W \gg 2)) \gg 1) + A < N\ ?\ (D \times W) \gg 5 : -((D \times W) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetYmotion = \begin{cases} ((64 - (H \gg 2)) \gg 1) + A < N\ ?\ (D \times H) \gg 5 : -((D \times H) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64 - (H \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

$$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A\ \%\ N][x_{subblk} + offsetXmotion][y_{subblk} + offsetYmotion] & A\ \%\ N < N_{reduced} \\ g_motionMask[N - A\ \%\ N][x_{subblk} + offsetXmotion][(H \gg 2) - 1 - y_{subblk} + offsetYmotion] & \text{그렇지 않은 경우} \end{cases}$$

[0260]

[0261] 미리 정의된 마스크를 저장하는 데 필요한 비트의 수는 다음과 같이 열거된다:

- 블렌딩 가중치에 대해:$(256 \times 256) \times 9 \times 4 = 2{,}359{,}296$ 비트 = 294,912 바이트 ≅ 295 킬로바이트
- 움직임 필드 저장에 대해:$(64 \times 64) \times 9 \times 2 = 131{,}072$ 비트 = 16,384 바이트 ≅ 16 킬로바이트

[0262]

	140 파티셔닝 서브모드	108 파티셔닝 서브모드	80 파티셔닝 서브모드	64 파티셔닝 서브모드
블렌딩 가중치	294,912 바이트	294,912 바이트	229,376 바이트	196,608 바이트
움직임 저장	9,216 바이트	9,216 바이트	7,168 바이트	6,144 바이트
총계	304,128 바이트	304,128 바이트	236,544 바이트	202,752 바이트
원본 기하학적 설계에 비해 감소%	91.1%	88.4%	87.8%	87.0%

[0263]

[0264] 전술한 실시예에서, 오프셋 도출 및 크로마 가중치 도출을 위한 방법이 수정될 수 있다는 것이 유의된다.

[0265] 오프셋 도출을 위한 상이한 방법은 아래에 제시된다. 위의 실시예와 비교하여, 차이점은 이탤릭으로 및 굵게 강

조된다.

[0266] 오프셋 도출을 위한 식은 distanceIdx가 0이 아닐 때 오프셋이 0과 동일하지 않다는 것을 보장하기 위해 수정될 수 있다. 일 예시에서, 변수 offsetX 및 offsetY는 다음과 같이 계산된다.

$$offsetX = \begin{cases} (256-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((255-W) \gg 1) + A < N \,?\, D \times max(1, W \gg 3) : -(D \times max(1, W \gg 3)) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((255-H) \gg 1) - A < N \,?\, D \times \max(1, H \gg 3) : -(D \times \max(1, H \gg 3)) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0267]

[0268] 오프셋 도출은 기하학적 파티션 모드에서 지원되는 거리의 수를 기초로 할 수 있다. 일 예시에서, 7개의 거리가 지원될 때, 변수 offsetX 및 offsetY는 다음과 같이 계산된다(이는 제1 실시예에서 제시된 방법과 동일하다).

$$offsetX = \begin{cases} (256-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256-W) \gg 1) + A < N \,?\, (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256-H) \gg 1) + A < N \,?\, (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0269]

[0270] 다른 예시에서, 9개의 거리가 지원될 때, 변수 offsetX 및 offsetY는 다음과 같이 계산된다:

$$offsetX = \begin{cases} (256-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256-W) \gg 1) + A < N \,?\, (D \times W) \gg 4 : -((D \times W) \gg 4) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256-H) \gg 1) + A < N \,?\, (D \times H) \gg 4 : -((D \times H) \gg 4) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0271]

[0272] 오프셋 도출은 파티셔닝 각도를 기초로 할 수 있다. 일 예시에서 135° 내지 225° 및 315° 내지 45° 의 각도에 대해, 오프셋은 수직 방향으로 추가된다. 그렇지 않으면, 45° 내지 135° 및 225° 내지 315° 각도에 대해, 오프셋은 수평 방향으로 추가된다. 변수 offsetX 및 offsetY는 다음과 같이 계산된다.

$$offsetX = \begin{cases} (256-W) \gg 1 & \textbf{135° 내지 225° 및 315° 내지 45° 의 각도} \\ ((256-W) \gg 1) - A < N \,?\, (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256-H) \gg 1) + A < N \,?\, (D \times H) \gg 3 : -((D \times H) \gg 3) & \textbf{135° 내지 225° 및 315° 내지 45° 의 각도} \\ (256-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0273]

[0274] 크로마 샘플에 대한 가중치는 마스크의 제1 세트 $g_sampleWeight_L[]$로부터 직접적으로 도출될 수 있다. 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 크로마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0275] - 크로마 블록의 크기는 W'×H'이다

[0276] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0277] - 변수 offsetXchroma 및 offsetYchroma는 다음과 같이 계산된다.

$$offsetXchroma = \begin{cases} (256 - W') \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H' \geq W') \\ ((256 - W') \gg 1) + A < N \,?\, (D \times W') \gg 3 : -((D \times W') \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetYchroma = \begin{cases} ((256 - H') \gg 1) + A < N \,?\, (D \times H') \gg 3 : -((D \times H') \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H' \geq W') \\ (256 - H') \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0278]

[0279] - $sampleWeight_C[x][y] = g_sampleWeight_L[A \,\%\, N][x + offsetXchroma][y + offsetYchroma]$

[0280] 여기서 (x,y)는 각 크로마 샘플의 위치를 나타낸다.

[0281] 제1 세트에서 미리 정의된 마스크의 크기는 256×256이 아닐 수 있다. 크기는 최대 블록 크기 및 최대 시프트 오프셋에 의존할 수 있다. 최대 블록 크기가 S라고 가정하면, 지원된 거리의 수는 N_d이고 각 거리에 대한 시프트 오프셋은 다음: offset=(D×S)≫O과 같이 정의된다. 그 후, 미리 정의된 마스크의 폭 및 높이는 다음과 같이 계산될 수 있다.

$$S + \left(\left((N_d - 1) \gg 1 \right) \times S \right) \gg O \ll 1$$

[0282]

[0283] 일 예시에서 변수 S, N_d 및 O는 각각, 128, 9 및 4로 설정된다. 미리 정의된 마스크의 크기는 192×192로 설정된다. 다른 예시에서, 변수 S, N_d 및 O는 각각, 128, 7 및 3으로 설정된다. 미리 정의된 마스크의 크기는 224×224로 설정된다.

[0284] 일부 실시예에서, 크롭 방법과 동일하게, 마스크의 제1 세트 및 제2 세트가 미리 정의된다. 마스크의 제1 세트 $g_sampleWeight_L[]$은 크기가 256×256인 수개의 마스크를 포함할 수 있고, 마스크는 각 블록에 대한 블렌딩 가중치를 도출하는 데 사용된다. 마스크의 제2 세트 $g_motionMask[]$는 크기가 64×64인 수개의 마스크를 포함할 수 있고, 마스크는 각 블록에 대한 움직임 필드 저장을 위한 마스크를 도출하는 데 사용된다. 정사각형 블록의 경우, 크롭된 방법과 동일하게 마스크는 제1 세트 및 제2 세트에서의 마스크 중 하나로부터 크롭된다. 정사각형이 아닌 블록의 경우, 마스크는 제1 세트 및 제2 세트에서의 마스크 중 하나로부터 크롭되며, 업샘플링 프로세스가 후속된다.

[0285] 일 실시예에서, 제1 세트 및 제2 세트에서의 미리 정의된 마스크는 위에서 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서의 마스크 수는 모두 N이며, 여기서 N은 기하학적 파티션 모드에서 지원되는 각도의 수로 설정된다. 제1 세트 및 제2 세트에서 인덱스 n을 갖는 n번째 마스크는 각도 n의 마스크를 나타내며, 여기서 n은 0 내지 N-1의 범위이다. 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0286] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0287] - 변수 minSize는 min(W, H)으로 설정된다.

[0288] - 변수 ratioWH 및 ratioHW는 각각 log2(MAX(W/H, 1)) 및 log2(MAX(H/W, 1))로 설정된다.

[0289] - 변수 offset x 및 offset Y는 다음과 같이 계산될 수 있다.

$$offsetX = \begin{cases} (256 - minSize) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256 - minSize) \gg 1) + A < N \,?\, (D \times minSize) \gg 3 : -((D \times minSize) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256 - minSize) \gg 1) + A < N \,?\, (D \times minSize) \gg 3 : -((D \times minSize) \gg 3) & (\text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256 - minSize) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0290]

[0291] $$sampleWeight_L[x][y] = g_sampleWeight_L[A\,\%\,N][(x \gg ratioWH) + offsetX][(y \gg ratioHW) + offsetY]$$

[0292] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0293] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0294] - 변수 minSubblk는 min(W, H) >> 2로 설정된다.

[0295] - 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산될 수 있다:

$$offsetXmotion = \begin{cases} (64 - (minSubblk \gg 2)) \gg 1 & 수평 분할 또는 (수직이 아닌 분할 및 H \geq W) \\ ((64 - (minSubblk \gg 2)) \gg 1) + A < N\ ?\ (D \times minSubblk) \gg 5 : -((D \times minSubblk) \gg 5) & 그렇지 않은 경우 \end{cases}$$

$$offsetYmotion = \begin{cases} ((64 - (minSubblk \gg 2)) \gg 1) + A < N\ ?\ (D \times minSubblk) \gg 5 : -((D \times minSubblk) \gg 5) & 수평 분할 또는 (수직이 아닌 분할 및 H \geq W) \\ (64 - (minSubblk \gg 2)) \gg 1 & 그렇지 않은 경우 \end{cases}$$

[0296] $$motionMask[x_{subblk}][y_{subblk}] = g_motionMask[A\,\%\,N][(x_{subblk} \gg ratioWH) + offsetXmotion][(y_{subblk} \gg ratioHW) + offsetYmotion]$$

[0297] 다른 실시예에서, 제1 세트 및 제2 세트에서 미리 정의된 마스크는 위에서 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서 마스크의 수는 모두 $N_{reduced}$이며, 여기서 $N_{reduced}$=($N \gg 1$)+1이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0298] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0299] - 변수 minSize는 min(W, H)으로 설정된다.

[0300] - 변수 ratioWH 및 ratioHW는 각각 log2(MAX(W/H, 1)) 및 log2(MAX(H/W, 1))로 설정된다.

[0301] - 변수 offsetX 및 offsetY는 다음과 같이 계산될 수 있다:

$$offsetX = \begin{cases} (256 - minSize) \gg 1 & 수평 분할 또는 (수직이 아닌 분할 및 H \geq W) \\ ((256 - minSize) \gg 1) + A < N\ ?\ (D \times minSize) \gg 3 : -((D \times minSize) \gg 3) & 그렇지 않은 경우 \end{cases}$$

$$offsetY = \begin{cases} ((256 - minSize) \gg 1) + A < N\ ?\ (D \times minSize) \gg 3 : -((D \times minSize) \gg 3) & 수평 분할 또는 (수직이 아닌 분할 및 H \geq W) \\ (256 - minSize) \gg 1 & 그렇지 않은 경우 \end{cases}$$

$$sampleWeight_L[x][y] = \begin{cases} g_sampleWeight_L[A\,\%\,N][(x \gg ratioWH) + offset][(y \gg ratioHW) + offset] & A\,\%\,N < N_{reduced} \\ g_sampleWeight_L[N - A\,\%\,N][minSize - 1 - (x \gg ratioWH) + offsetX][(y \gg ratioHW) + offsetY] & 그렇지 않은 경우 \end{cases}$$

[0302]

[0303] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0304] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0305] - 변수 minSubblk는 min(W, H) >> 2로 설정된다.

[0306] - 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산될 수 있다:

$$offsetXmotion = \begin{cases} (64 - (\text{minSubblk} \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64 - (\text{minSubblk} \gg 2)) \gg 1) + A < N\ ?\ (D \times \text{minSubblk}) \gg 5 : -((D \times \text{minSubblk}) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetYmotion = \begin{cases} ((64 - (\text{minSubblk} \gg 2)) \gg 1) + A < N\ ?\ (D \times \text{minSubblk}) \gg 5 : -((D \times \text{minSubblk}) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64 - (\text{minSubblk} \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

$$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A\ \%\ N][(x_{subblk} \gg ratioWH) + offsetXmotion][(y_{subblk} \gg ratioHW) + offsetYmotion] & A\ \%\ N < N_{reduced} \\ g_motionMask[N - A\ \%\ N][minSubblk - 1 - (x_{subblk} \gg ratioWH) + offsetXmotion][(y_{subblk} \gg ratioHW) + offsetYmotion] & \text{그렇지 않은 경우} \end{cases}$$

[0307]

[0308] 제3 실시예에서, 제1 세트 및 제2 세트에서 미리 정의된 마스크는 위에서 설명된 식을 사용하여 계산될 수 있다: 제1 세트 및 제2 세트에서 마스크의 수는 모두 $N_{reduced}$이며, 여기서 $N_{reduced}=(N \gg 1)+1$이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0309] - 변수 angleIdx A 및 distanceIdx D는 기하학적 파티셔닝 인덱스 K를 사용하여 룩업 테이블로부터 획득된다. 룩업 테이블의 예시는 테이블 15C, 테이블 15D, 테이블 22B, 테이블 30 및 테이블 31에 도시된다.

[0310] - 변수 minSize는 min(W, H)으로 설정된다.

[0311] - 변수 ratioWH 및 ratioHW는 각각 log2(MAX(W/H, 1)) 및 log2(MAX(H/W, 1))로 설정된다.

[0312] - 변수 offsetX 및 offsetY는 다음과 같이 계산될 수 있다:

$$offsetX = \begin{cases} (256 - minSize) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256 - minSize) \gg 1) + A < N\ ?\ (D \times minSize) \gg 3 : -((D \times minSize) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256 - minSize) \gg 1) + A < N\ ?\ (D \times minSize) \gg 3 : -((D \times minSize) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256 - minSize) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

$$sampleWeight_L[x][y] = \begin{cases} g_sampleWeight_L[A\ \%\ N][(x \gg ratioWH) + offsetX][(y \gg ratioHW) + offsetY] & A\ \%\ N < N_{reduced} \\ g_sampleWeight_L[N - A\ \%\ N][(x \gg ratioWH) + offsetX][minSize - 1 - (y \gg ratioHW) + offsetY] & \text{그렇지 않은 경우} \end{cases}$$

[0313]

[0314] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0315] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0316] - 변수 minSubblk는 min(W, H) >> 2로 설정된다.

[0317] - 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산될 수 있다:

$$offsetXmotion = \begin{cases} (64 - (\text{minSubblk} \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64 - (\text{minSubblk} \gg 2)) \gg 1) + A < N\ ?\ (D \times \text{minSubblk}) \gg 5 : -((D \times \text{minSubblk}) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetYmotion = \begin{cases} ((64 - (\text{minSubblk} \gg 2)) \gg 1) + A < N\ ?\ (D \times \text{minSubblk}) \gg 5 : -((D \times \text{minSubblk}) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64 - (\text{minSubblk} \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

$$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A\ \%\ N][(x_{subblk} \gg ratioWH) + offsetXmotion][(y_{subblk} \gg ratioHW) + offsetYmotion] & A\ \%\ N < N_{reduced} \\ g_motionMask[N - A\ \%\ N][(x_{subblk} \gg ratioWH) + offsetmotionX][minSubblk - 1 - (y_{subblk} \gg ratioHW) + offsetmotionY] & \text{그렇지 않은 경우} \end{cases}$$

[0318]

[0319] 위에서 설명된 오프셋 및 크로마 가중치 도출을 위한 상이한 방법이 또한 본원에 적용될 수 있다는 것이 유의된다.

[0320] 기하학적 파티션 모드에 대한 원본 설계에서, 블록 중심을 135° 또는 45° 로 가로지르는 블록을 파티셔닝하는 것의 조합이 항상 제외된다. 주요 목적은 기하학적 파티션 모드로부터 삼각형 파티션 모드와의 리던던시 파티션 옵션을 제거하는 것이다. 하지만, 삼각형 파티션 모드를 사용하여 코딩된 정사각형이 아닌 블록에 대한 파티셔닝 각도는 135° 도 45° 도 아니다. 그러므로, 블록의 형태를 기초로, 두 개의 각도가 적응적으로 제외될 수 있다.

[0321] 본 개시의 일부 실시예에서, 제외되는 두 개의 각도는 블록 형태를 기초로 변경된다. 정사각형 블록의 경우, 135° 및 45° 는 제외되며, 이는 원본 기하학적 파티션 설계와 동일하다. 다른 블록 형태의 경우, 제외된 각도는 도 37의 테이블 37에 열거된다. 예를 들어, 크기가 8×16인(즉, 폭 대 높이 비율이 1:2인) 블록의 경우, 각도 112.5° 및 67.5° , 즉 angleIdx 10 및 angleIdx 6은 제외된다. 그 후, 기하학적 파티셔닝 인덱스에 대한 룩업 테이블이 도 38의 테이블 38로 수정된다. 제외된 각도에 관련된 테이블 38의 부분은 도 38에서 회색으로 강조된다.

[0322] 이 실시예는 본 개시에서의 다른 실시예와 결합될 수 있다는 것이 유의된다. 예를 들어, 제외된 각도를 사용하는 블렌딩 가중치 및 움직임 필드 저장의 마스크는 삼각형 파티션 모드의 방법을 사용하여 계산될 수 있다. 다른 각도의 경우, 크롭된 방법은 마스크를 도출하기 위해 사용된다.

[0323] 이전에 언급된 바와 같이, 삼각형 파티션 및 기하학적 모드에서 사용되는 블렌딩 프로세스, 움직임 필드 저장 및 신택스 구조는 통합되지 않는다. 본 개시에서, 모든 프로세스를 통합하는 것이 제안된다.

[0324] 일 실시예에서, 위에서 설명된 제안된 크롭된 방법은 삼각형 및 기하학적 파티션 모드 모두에 대한 블렌딩 및 움직임 필드 저장 프로세스에서 사용된다. 게다가, 각 기하학적 파티셔닝 서브모드의 각도 및 거리에 대한 룩업 테이블이 제거된다. 마스크의 제1 세트 및 제2 세트는 미리 정의되고, 각각 위에서 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서의 마스크 수는 모두 $N_{reduced}$이고, 여기서 $N_{reduced}=(N\gg1)+1$이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. N_D는 기하학적 파티션 모드에서 지원되는 거리의 수를 나타낸다고 하자. 따라서, 기하학적 파티션 서브모드의 총 수는 $N\times N_D$이다. 일 예시에서, N 및 N_D는 각각 8 및 7로 설정된다. 다른 예시에서, N 및 N_D는 각각 12 및 7로 설정된다. 다른 예시에서, N 및 N_D는 각각 짝수 및 홀수로 설정된다.

[0325] - 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 A 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0326] - 변수 N_{halfD}는 N_D >> 1로 설정된다.

[0327] - 변수 angleIdx A 및 distanceIdx D는 각각 K % N 및 K/N으로 설정된다.

[0328] - 변수 offsetX 및 offsetY는 다음과 같이 계산된다:

$$offsetX = \begin{cases} (256-W)\gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256-W)\gg 1) + D > N_{halfD} \,?\, ((D-N_{halfD})\times W)\gg 3 : -((D\times W)\gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256-H)\gg 1) + D > N_{halfD} \,?\, ((D-N_{halfD})\times H)\gg 3 : -((D\times H)\gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256-H)\gg 1 & \text{그렇지 않은 경우} \end{cases}$$

- $$sampleWeight_L[x][y] = \begin{cases} g_sampleWeight_L[A][x+offsetX][y+offsetY] & A < N_{reduced} \\ g_sampleWeight_L[N-A][W-1-x+offsetX][y+offsetY] & \text{그렇지 않은 경우} \end{cases}$$

[0329]

[0330] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0331] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0332] - 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산된다.

$$offsetXmotion = \begin{cases} (64 - (W \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64 - (W \gg 2)) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times W) \gg 5 : -((D \times W) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetYmotion = \begin{cases} ((64 - (H \gg 2)) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times H) \gg 5 : -((D \times H) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64 - (H \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

$$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A][x_{subblk} + offsetXmotion][y_{subblk} + offsetYmotion] & A < N_{reduced} \\ g_motionMask[N - A][(W \gg 2) - 1 - x_{subblk} + offsetXmotion][y_{subblk} + offsetYmotion] & \text{그렇지 않은 경우} \end{cases}$$

[0333]

[0334] 미리 정의된 마스크를 저장하는 데 필요한 비트의 수는 (256×256)×((*N*≫1)+1)×4+(64×64)×((*N*≫1)+1)×2이다.

	N=16	N=12	N=8
제안된 방법의 메모리	304,128 바이트	236,544 바이트	168,960 바이트
140 기하학적 파티셔닝 서브모드 + 삼각형 모드 대비 감소%	91.2%	93.2%	95.1%
108 기하학적 파티셔닝 서브모드 + 삼각형 모드 대비 감소%	88.6%	91.2%	93.7%
80 기하학적 파티셔닝 서브모드 + 삼각형 모드 대비 감소%	84.8%	88.1%	91.5%
64 기하학적 파티셔닝 서브모드 + 삼각형 모드 대비 감소%	81.1%	85.3%	89.5%

[0335]

[0336] 다른 실시예에서, 위에서 설명된 제안된 크롭된 방법은 삼각형 및 기하학적 파티션 모드 모두에 대한 블렌딩 및 움직임 필드 저장의 프로세스에서 사용된다. 게다가, 각 기하학적 파티셔닝 서브모드의 각도 및 거리에 대한 룩업 테이블이 제거된다. 마스크의 제1 세트 및 제2 세트는 미리 정의되고, 각각 위에서 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서의 마스크 수는 모두 $N_{reduced}$이고, 여기서 $N_{reduced}$=(*N*≫1)+1이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. N_D는 기하학적 파티션 모드에서 지원되는 거리의 수를 나타낸다고 한다. 그러므로, 기하학적 파티션 서브모드의 총 수는 $N \times N_D$이다.

[0337] 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0338] - 변수 N_{halfD}는 N_D >> 1로 설정된다.

[0339] - 변수 angleIdx A 및 distanceIdx D는 각각 K % N 및 K/N으로 설정된다.

[0340] - 변수 offsetX 및 offsetY는 다음과 같이 계산된다:

$$offsetX = \begin{cases} (256 - W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((256 - W) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((256 - H) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (256 - H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

- $sampleWeight_L[x][y] =$
$$\begin{cases} g_sampleWeight_L[A][x + offsetX][y + offsetY] & A < N_{reduced} \\ g_sampleWeight_L[N - A][x + offsetX][H - 1 - y + offsetY] & \text{그렇지 않은 경우} \end{cases}$$

[0341]

[0342] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0343] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다.

[0344] - 변수 offsetXmotion 및 offsetYmotion은 다음과 같이 계산된다.

$$offsetXmotion = \begin{cases} (64 - (W \gg 2)) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((64 - (W \gg 2)) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times W) \gg 5 : -((D \times W) \gg 5) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetYmotion = \begin{cases} ((64 - (H \gg 2)) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times H) \gg 5 : -((D \times H) \gg 5) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (64 - (H \gg 2)) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

$$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A][x_{subblk} + offsetXmotion][y_{subblk} + offsetYmotion] & A < N_{reduced} \\ g_motionMask[N - A][x_{subblk} + offsetXmotion][(H \gg 2) - 1 - y_{subblk} + offsetYmotion] & \text{그렇지 않은 경우} \end{cases}$$

[0345]

[0346] 제3 실시예에서, 위에서 설명된 제안된 업샘플링된 방법은 삼각형 및 기하학적 파티션 모드 모두에 대한 블렌딩 및 움직임 필드 저장의 프로세스에서 사용된다. 게다가, 각 기하학적 파티셔닝 서브모드의 각도 및 거리에 대한 룩업 테이블이 제거된다. 마스크의 제1 세트 및 제2 세트는 미리 정의되고, 각각 위에서 설명된 식을 사용하여 계산될 수 있다. 제1 세트 및 제2 세트에서의 마스크 수는 모두 $N_{reduced}$이고, 여기서 $N_{reduced}$=($N \gg 1$)+1이고 N은 기하학적 파티션 모드에서 지원되는 각도의 수이다. N_D는 기하학적 파티션 모드에서 지원되는 거리의 수를 나타낸다고 한다. 따라서, 기하학적 파티션 서브모드의 총 수는 $N \times N_D$이다. 크기가 W×H이고 기하학적 파티셔닝 인덱스가 K로 설정된 블록의 경우, 루마 샘플의 블렌딩 가중치에 대한 마스크는 다음과 같이 도출된다.

[0347] - 변수 N_{halfD}는 $N_D \gg 1$로 설정된다.

[0348] - 변수 angleIdx A 및 distanceIdx D는 각각 K % N 및 K/N으로 설정된다.

[0349] - 변수 minSize는 min(W, H)으로 설정된다.

[0350] - 변수 ratioWH 및 ratioHW는 각각 log2(MAX(W/H, 1)) 및 log2(MAX(H/W, 1))로 설정된다.

[0351] - 변수 offset은 다음과 같이 계산된다:

[0352] $$offset = ((256 - minSize) \gg 1) + D > N_{halfD} \; ? \; ((D - N_{halfD}) \times minSize) \gg 3 : -((D \times minSize) \gg 3)$$

- $sampleWeight_L[x][y] =$
$$\begin{cases} g_sampleWeight_L[A][(x \gg ratioWH) + offset][(y \gg ratioHW) + offset] & A < N_{reduced} \\ g_sampleWeight_L[N-A][minSize - 1 - (x \gg ratioWH) + offset][(y \gg ratioHW) + offset] & \text{그렇지 않은 경우} \end{cases}$$

[0353]

[0354] 크로마 샘플에 대한 블렌딩 가중치는 루마 샘플의 가중치로부터 서브샘플링된다. 즉, YUV 4:2:0 비디오 포맷에 대한 크로마 샘플의 가중치로서 각 대응하는 2×2 루마 서브블록에 대한 좌측-상단 루마 샘플의 가중치가 사용된다.

[0355] 다른 한편으로, 움직임 필드 저장을 위한 마스크는 다음과 같이 도출된다:

[0356] - 변수 minSubblk는 min(W, H) >> 2로 설정된다.

[0357] - 변수 offsetmotion은 다음과 같이 계산된다.

[0358] $$offsetmotion = \left((64 - \mathbf{minSubblk}) \gg 1\right) + D > N_{halfD} \; ? \left((D - N_{halfD}) \times minSize\right) \gg 5 \; : \; -((D \times minSize) \gg 5)$$

[0359] $$motionMask[x_{subblk}][y_{subblk}] = \begin{cases} g_motionMask[A][(x_{subblk} \gg ratioWH) + offsetmotion][(y_{subblk} \gg ratioHW) + offsetmotion] & A < N_{reduced} \\ g_motionMask[N-A][minSubblk - 1 - (x_{subblk} \gg ratioWH) + offsetmotion][(y_{subblk} \gg ratioHW) + offsetmotion] & \text{그렇지 않은 경우} \end{cases}$$

[0360] 도 39는 본 개시의 일부 실시예에 따른, 비디오 콘텐츠를 처리하기 위한 예시적인 방법(3900)의 흐름도이다. 일부 실시예에서, 방법(3900)은 코덱(예를 들어, 도 2a-2b에서의 인코딩 프로세스(200A 또는 200B)를 사용하는 인코더, 또는 도 3a-3b에서의 디코딩 프로세스(300A 또는 300B)를 사용하는 디코더)에 의해 수행될 수 있다. 예를 들어, 코덱은 비디오 시퀀스를 인코딩 또는 트랜스코딩하기 위한 장치(예를 들어, 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소로서 구현될 수 있다. 일부 실시예에서, 비디오 시퀀스는 압축해제된 비디오 시퀀스(예를 들어, 비디오 시퀀스(202)) 또는 디코딩된 압축된 비디오 시퀀스(예를 들어, 비디오 스트림(304))일 수 있다. 일부 실시예에서, 비디오 시퀀스는 장치의 프로세서(예를 들어, 프로세서(402))와 연관된 모니터링 디바이스(예를 들어, 도 4에서의 비디오 입력 디바이스)에 의해 캡처될 수 있는 모니터링 비디오 시퀀스일 수 있다. 비디오 시퀀스는 다수의 픽처를 포함할 수 있다. 장치는 픽처의 레벨에서 방법(3900)을 수행할 수 있다. 예를 들어, 장치는 방법(3900)에서 하나의 픽처를 동시에 처리할 수 있다. 다른 예시에 대해, 장치는 방법(3900)에서 복수의 픽처를 동시에 처리할 수 있다. 방법(3900)은 아래와 같은 단계를 포함할 수 있다.

[0361] 단계 3902에서, 복수의 블록은 파티셔닝 엣지를 따라 제1 파티션 및 제2 파티션으로 파티셔닝될 수 있다.

[0362] 복수의 블록은 픽처와 연관된 제1 블록의 서브블록이다. 픽처는 (제1 블록을 포함하는) 복수의 블록과 연관될 수 있고, 블록의 각각은 복수의 서브블록으로 나누어질 수 있다는 것이 인식된다. 제1 블록은 크로마 블록 및 루마 블록과 연관될 수 있다. 따라서, 복수의 블록(예를 들어, 서브블록)의 각각은 크로마 서브블록 및 루마 서브블록과 연관될 수 있다. 복수의 블록(예를 들어, 서브블록)의 파티션 모드가 결정될 수 있고, 복수의 블록(예를 들어, 서브블록)은 파티션 모드를 기초로 분할될 수 있다. 파티셔닝은 제1 블록에 대한 인터 예측에 대한 개선을 제공할 수 있다. 예시적인 파티션 모드는 삼각형 파티션 모드 또는 기하학적 파티션 모드를 포함할 수 있다.

[0363] 위에서 논의된 바와 같이, 파티션 모드는 적어도 하나의 표시 신호에 따라 결정될 수 있다. 예를 들어, 다시 도 19를 참조하면, 제1 표시 신호(예를 들어, 서브블록 병합 플래그), 제2 표시 신호(예를 들어, 정규 병합/MMVD 플래그), 및 제3 표시 신호(예를 들어, CIIP 플래그)가 제1 블록에 대한 파티션 모드를 결정하기 위해 제공된다. 도 19에 도시된 바와 같이, 제1 표시 신호(예를 들어, 서브블록 병합 플래그)에 따라 제1 블록이 서브블록 병합 모드를 사용하여 코딩되었는지가 결정될 수 있다. 제1 블록이 서브블록 병합 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 제2 표시 신호(예를 들어, 정규 병합/MMVD 플래그)에 따라 제1 블록이 정규 모드 또는 움직임 벡터 차분을 갖는 병합 모드(merge mode with motion vector differences, MMVD) 중 하나를 사용하여 코딩되는지가 결정될 수 있다. 정규 모드 또는 MMVD를 사용하여 제1 블록이 코딩되지 않는다는 결정에 응답하여, 제1 블록이 CIIP를 사용하여 코딩되었는지가 CIIP 플래그를 사용하여 결정될 수 있다.

[0364] 일부 실시예에서, 제3 표시 신호에 따라 제1 블록이 CIIP를 사용하여 코딩되는지를 결정하기 전에, 단계 3902는: 제1 블록의 크기가 주어진 조건을 충족하는지를 결정하는 단계; 제1 블록의 크기가 주어진 조건을 충족한다는 결정에 응답하여 제3 표시 신호를 생성하거나, 또는 제1 블록의 크기가 주어진 조건을 충족하지 못한다는 결정에 응답하여, CIIP 모드를 사용하여 제1 블록이 코딩된다고 결정하는 단계를 포함한다. 주어진 조건은:

제1 블록의 폭 및 높이가 모두 8 이상이고; 폭과 높이 사이의 더 큰 값과 폭과 높이 사이의 더 작은 값의 비율이 4 이하임을 포함할 수 있다.

[0365] 그 후, 제1 블록이 CIIP 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 제1 블록의 파티션 모드가 삼각형 파티션 모드 또는 기하학적 파티션 모드 중 하나인 것으로 결정될 수 있다.

[0366] 제1 블록의 파티션 모드가 삼각형 파티션 모드 또는 기하학적 파티션 모드 중 하나로 결정될 경우, 파티션 모드 인덱스, 각도 인덱스 또는 거리 인덱스에 따라 타겟 파티셔닝 방식이 더 결정될 수 있다. 그 후, 타겟 파티셔닝 방식에 대응하는 파티셔닝 엣지가 결정될 수 있다.

[0367] 일반적으로, 파티션 모드(삼각형 파티션 모드 또는 기하학적 파티션 모드)는 복수의 파티셔닝 방식과 연관될 수 있으며, 파티션 모드 인덱스는 복수의 파티셔닝 방식 중 파티셔닝 방식의 수를 나타낼 수 있다. 파티션 모드 인덱스는 각도 인덱스 및 거리 인덱스와 연관될 수 있다. 각도 인덱스는 파티션 모드 인덱스에 대응하는 주어진 파티셔닝 방식의 파티셔닝 엣지의 각도를 나타낼 수 있고, 거리 인덱스는 파티셔닝 엣지와 제1 블록의 중심 사이의 거리를 나타낼 수 있다.

[0368] 일부 실시예에서, 룩업 테이블(예를 들어, 도 22a의 테이블 22A 또는 도 22b의 테이블 22B)은 복수의 파티셔닝 방식과 연관된, 복수의 파티션 모드 인덱스, 복수의 각도 인덱스 및 복수의 거리 인덱스를 포함할 수 있다. 파티션 모드 인덱스(예를 들어, K)가 주어지면, 파티셔닝 방식과 연관된 각도 인덱스 및 거리 인덱스가 결정될 수 있다. 따라서, 파티션 모드 인덱스에 따라 룩업 테이블에서 타겟 파티셔닝 방식이 결정될 수 있다.

[0369] 복수의 파티셔닝 방식 중에서, 룩업 테이블은 제1 파티션 모드 인덱스와 연관된 제1 파티셔닝 방식 및 제2 파티션 모드 인덱스와 연관된 제2 파티셔닝 방식을 포함할 수 있고, 제1 파티셔닝 방식 및 제2 파티셔닝 방식은 삼각형 파티션 모드에 관련된다. 예를 들어, 도 22b의 테이블 22B를 참조하면, 파티션 모드 인덱스 "10"이 블록을 좌측-상단 코너로부터 우측-하단 코너로 파티셔닝하는 것과 연관된다. 그리고, "24"의 파티션 모드 인덱스는 블록을 우측-상단 코너로부터 좌측-하단 코너로 파티셔닝하는 것과 연관된다.

[0370] 단계 3904에서, 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해 복수의 블록에 대해 인터 예측이 수행될 수 있다. 제1 움직임 벡터는 제1 예측 신호(예를 들어, 제1 파티션에 대한 움직임 벡터)를 생성하기 위해 제1 파티션에 적용될 수 있고, 제2 움직임 벡터는 제2 예측 신호(예를 들어, 제2 파티션에 대한 움직임 벡터)를 생성하기 위해 제2 파티션에 적용될 수 있다. 제1 예측 신호 및 제2 예측 신호는 4×4 서브 블록에 저장될 수 있다.

[0371] 단계 3906에서, 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호가 블렌딩될 수 있다. 제1 블록이 제1 블록 상에 인터 예측을 적용하기 전에 파티셔닝되었기 때문에, 제1 블록의 파티션은 블록의 코딩을 완료하기 위해 블렌딩될 수 있다. 일부 실시예에서, 파티셔닝 엣지와 관련된 엣지 블록이 블렌딩될 수 있다. 각 파티션의 블록에 대한 예측 신호가 동일하다는 것이 인식된다. 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하기 위해, 엣지 블록의 각각에 대한 가중치가 결정될 수 있다.

[0372] 일부 실시예에서, 블렌딩 가중치를 결정하기 위해 마스크의 세트가 생성될 수 있다. 예를 들어, 마스크의 제1 세트(예를 들어, $g_sampleWeight_L[]$)는 수개의 마스크를 포함할 수 있으며 각 마스크의 크기는 256×256이고, 마스크는 각 블록의 블렌딩 가중치를 도출하는 데 사용된다. 마스크의 세트의 수(즉, N)는 기하학적 파티션 모드에서 지원되는 각도의 수로 설정될 수 있다. 예를 들어, 도 22b의 테이블 22B를 참조하면, 기하학적 파티션 모드에서 지원하는 각도의 수는 64이며 따라서 마스크의 수는 16이다. 제1 오프셋(예를 들어, offsetX) 및 제2 오프셋(예를 들어, offsetY)은 블록의 크기, 타겟 파티셔닝 방식의 각도 값 및 타겟 파티셔닝 방식의 거리 값을 기초로 결정될 수 있다. 예를 들어, W×H 크기의 블록에 대해 offsetX 및 offsetY는 아래의 식을 사용하여 결정될 수 있다.

$$offsetX = \begin{cases} (-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ ((-W) \gg 1) + A < N\ ?\ (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((-H) \gg 1) - A < N\ ?\ (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W) \\ (-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0373]

[0374] 여기서 A 및 D는 각각 각도 인덱스 및 거리 인덱스이다. 각도 인덱스 A 및 거리 인덱스 D는 파티션 모드 인덱스

K를 사용하여 룩업 테이블(예를 들어, 도 22b의 테이블 22B)로부터 획득될 수 있다. 따라서, 마스크의 세트의 수는 룩업 테이블의 각도 인덱스의 수를 기초로 결정되고, 타겟 파티셔닝 방식의 각도 값은 타겟 파티셔닝 방식에 대응하는 룩업 테이블의 각도 인덱스를 기초로 결정되며, 타겟 파티셔닝 방식의 거리 값은 타겟 파티셔닝 방식에 대응하는 룩업 테이블의 거리 인덱스를 기초로 결정된다.

[0375] 각도 인덱스 A는 분할이 수평인지 또는 수직인지 결정하는 데 사용된다. 각도 인덱스 A가 도 22b의 룩업 테이블 22B로부터 도출된 경우, 각도 인덱스 A가 8 또는 24와 동일할 때, 블록은 수평 분할된다. 다른 한편으로, 각도 인덱스가 0 또는 16에 동일할 때, 블록은 수직 분할이다. 그러므로, *수평 분할 또는 (수직이 아닌 분할 및 H≥W)* 조건은 $A \% 16==8$ *또는* $(A \% 16 != 0$ *및* $H \geq W)$에 동등하다.

[0376] 다른 예시로서, W×H의 크기를 갖는 블록에 대해 offsetX 및 offsetY는 아래의 식을 사용하여 결정될 수 있다.

$$offsetX = \begin{cases} (-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W\text{)} \\ ((-W) \gg 1) + D > N_{halfD} \ ? \ ((D - N_{halfD}) \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((-H) \gg 1) + D > N_{halfD} \ ? \ ((D - N_{halfD}) \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W\text{)} \\ (-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0377]

[0378] N_{halfD}는 기하학적 파티션 모드에서 지원되는 거리의 수의 절반이고, 각도 인덱스 A 및 거리 인덱스 D는 파티션 모드 인덱스 K를 기초로 결정될 수 있다. 예를 들어, A=K%N 및 D=K/N이다. 따라서, 타겟 파티셔닝 방식의 각도 값은 파티션 모드 인덱스 및 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정되고, 타겟 파티셔닝 방식의 거리 값은 파티션 모드 인덱스 및 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정된다.

[0379] 마스크의 세트(예를 들어, $g_sampleWeight_L[]$)를 기초로, 제1 오프셋 및 제2 오프셋(offsetX 및 offsetY)을 사용하여 생성될 수 있는 엣지 서브블록에 대한 복수의 블렌딩 가중치가 결정될 수 있다.

[0380] 일부 실시예에서, 마스크의 세트를 사용하지 않고 제1 오프셋(예를 들어, offsetX) 및 제2 오프셋(예를 들어, offsetY)이 결정될 수 있고, 블렌딩 가중치가 즉각적으로 결정될 수 있다. 예를 들어, 각 엣지 블록에 대한 가중치는 다음의 식을 사용하여 계산될 수 있다:

$$offsetX = \begin{cases} (-W) \gg 1 & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W\text{)} \\ ((-W) \gg 1) + A < N \ ? \ (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{그렇지 않은 경우} \end{cases}$$

$$offsetY = \begin{cases} ((-H) \gg 1) + A < N \ ? \ (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{수평 분할 또는 (수직이 아닌 분할 및 } H \geq W\text{)} \\ (-H) \gg 1 & \text{그렇지 않은 경우} \end{cases}$$

[0381] $displacementX = angleIdx$

[0382] $displacementY = (displancementX + NumAngles \gg 2)\%NumAngles$, 여기서 $NumAngles$는 32로 설정된다.

[0383] 위치 (x, y)에 위치된 루마 샘플에 대한 가중치(예를 들어, $sampleWeight_L$ [x][y])는 다음과 같이 계산할 수 있다:

$$weightIdx = (((x + offsetX) << 1) + 1) * disLut[displacementX] + (((y + offsetY) << 1) + 1)) * disLut[displacementY]$$

$$partFlip = (angleIdx >= 13 \&\& angleIdx <= 27) \ ? \ 0 : 1$$

$$weightIdxL = partFlip \ ? \ 32 + weightIdx \ : \ 32 - weightIdx$$

$$sampleWeight_L[x][y] = Clip3(0, 8, (weightIdxL + 4) \gg 3)$$

[0384]

[0385] 제1 블록이 크로마 블록 및 루마 블록을 포함하기 때문에, 복수의 블렌딩 가중치는 엣지 블록에 대한 복수의 루마 가중치 및 엣지 블록에 대한 복수의 크로마 가중치를 포함할 수 있다. 복수의 크로마 가중치 중에서, 크로마 가중치는 크로마 가중치에 대응하는 2×2 서브블록의 좌측-상단 코너에 대한 루마 가중치를 기초로 결정된다. 예를 들어, 도 7을 참조로, 2x2 루마 서브블록의 좌측-상단 코너에 대한 루마 가중치는 크로마 블록에 대한 크로마 가중치로 사용될 수 있다.

[0386] 따라서, 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계는: 엣지 블록에 대한 복수의 루마 가중치에 따라 엣지 블록의 루마 값을 결정하기 위해 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계; 및 엣지 블록에 대한 복수의 크로마 가중치에 따라 엣지 블록의 크로마 값을 결정하기 위해 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계를 더 포함할 수 있다.

[0387] 일부 실시예에서, 명령어를 포함하는 비일시적 컴퓨터 판독가능 저장 매체가 또한 제공되고, 명령어는 상술한 방법을 수행하기 위한 (개시된 인코더 및 디코더와 같은) 디바이스에 의해 실행될 수 있다. 비일시적 매체의 일반적인 형태는 예를 들어, 플로피 디스크, 플렉서블 디스크(flexible disk), 하드 디스크, 솔리드 스테이트 드라이브(solid state drive), 자기 테이프 또는 임의의 다른 자기 데이터 저장 매체, CD-ROM, 임의의 다른 광학 데이터 저장 매체, 구멍의 패턴을 갖는 임의의 물리적 매체, RAM, PROM 및 EPROM, FLASH-EPROM 또는 임의의 다른 플래시 메모리, NVRAM, 캐시, 레지스터, 임의의 다른 메모리 칩 또는 카트리지 및 이들의 네트워크화된 버전을 포함한다. 디바이스는 하나 이상의 프로세서(CPU), 입/출력 인터페이스, 네트워크 인터페이스 및/또는 메모리를 포함할 수 있다.

[0388] 실시예는 다음 조항을 사용하여 더 설명될 수 있다.

[0389] 1. 비디오 콘텐츠를 처리하기 위한 방법으로서,

[0390] 파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계;

[0391] 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측을 수행하는 단계; 및

[0392] 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계를 포함하는, 방법.

[0393] 2. 조항 1에 있어서, 파티셔닝 엣지를 따라 복수의 블록을 파티셔닝하는 단계는:

[0394] 복수의 블록의 파티션 모드를 결정하는 단계; 및

[0395] 파티션 모드를 기초로 복수의 블록을 파티셔닝하는 단계를 더 포함하는, 방법.

[0396] 3. 조항 2에 있어서, 복수의 블록은 제1 블록의 서브블록이고, 복수의 블록에 대한 파티션 모드를 결정하는 단계는:

[0397] 제1 표시 신호에 따라, 제1 블록이 서브블록 병합 모드를 사용하여 코딩되는지를 결정하는 단계;

[0398] 제1 블록이 서브블록 병합 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 제2 표시 신호에 따라, 제1 블록이 정규 모드(regular mode) 또는 움직임 벡터 차분(motion vector differences)을 갖는 병합 모드(MMVD) 중 하나를 사용하여 코딩되는지를 결정하는 단계;

[0399] 제1 블록이 정규 모드 또는 MMVD 중 하나를 사용하여 코딩되지 않는다는 결정에 응답하여, 제3 표시 신호에 따라, 제1 블록이 결합된 인터 및 인트라 예측(CIIP) 모드를 사용하여 코딩되는지를 결정하는 단계; 및

[0400] 제1 블록이 CIIP 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 복수의 블록의 파티션 모드가 삼각형 파티션 모드 또는 기하학적 파티션 모드 중 하나라고 결정하는 단계를 더 포함하는, 방법.

[0401] 4. 조항 2 또는 3에 있어서, 파티셔닝 엣지를 따라, 복수의 블록을 파티셔닝하는 단계는:

[0402] 파티션 모드 인덱스, 각도 인덱스 또는 거리 인덱스에 따라 타겟 파티셔닝 방식을 결정하는 단계; 및

[0403] 타겟 파티셔닝 방식에 대응하는 파티셔닝 엣지를 결정하는 단계를 더 포함하는, 방법.

[0404] 5. 조항 4에 있어서,

[0405] 마스크의 세트를 생성하는 단계;

[0406] 제1 블록의 크기, 타겟 파티셔닝 방식의 각도 값, 타겟 파티셔닝 방식의 거리 값을 기초로 제1 오프셋 및 제2 오프셋을 결정하는 단계; 및

[0407] 마스크의 세트를 기초로, 제1 오프셋 및 제2 오프셋을 사용하여 복수의 블렌딩 가중치를 생성하는 단계를 더 포함하는, 방법.

[0408] 6. 조항 5에 있어서,

[0409] 마스크의 세트의 수는 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정되고, 타겟 파티셔닝 방식의 각도 값은 파티션 모드 인덱스 및 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정되며, 타겟 파티셔닝 방식의 거리 값은 파티션 모드 인덱스 및 기하학적 파티션 모드에서 지원되는 각도의 수를 기초로 결정되는, 방법.

[0410] 7. 조항 5 또는 6에 있어서, 파티션 모드 인덱스, 각도 인덱스 또는 거리 인덱스에 따라 타겟 파티셔닝 방식을 결정하는 단계는:

[0411] 룩업 테이블에 따라 타겟 파티셔닝 방식을 결정하는 단계 - 룩업 테이블은 복수의 파티셔닝 방식과 연관된 복수의 파티션 모드 인덱스, 복수의 각도 인덱스 및 복수의 거리 인덱스를 포함함 -를 더 포함하는, 방법.

[0412] 8. 조항 7에 있어서,

[0413] 마스크의 세트의 수는 룩업 테이블의 각도 인덱스의 수를 기초로 결정되고, 타겟 파티셔닝 방식의 각도 값은 타겟 파티셔닝 방식에 대응하는 룩업 테이블의 각도 인덱스를 기초로 결정되고, 및 타겟 파티셔닝 방식의 거리 값은 타겟 파티셔닝 방식에 대응하는 룩업 테이블의 거리 인덱스를 기초로 결정되는, 방법.

[0414] 9. 조항 7 또는 8에 있어서, 룩업 테이블은 복수의 파티셔닝 방식 중에서, 제1 파티션 모드 인덱스와 연관된 제1 파티셔닝 방식 및 제2 파티션 모드 인덱스와 연관된 제2 파티셔닝 방식을 포함하고, 제1 파티셔닝 방식 및 제2 파티셔닝 방식은 삼각형 파티션 모드에 대응하는, 방법.

[0415] 10. 조항 9에 있어서, 제1 파티션 모드 인덱스는 10과 동일하고, 제1 파티션 모드 인덱스와 연관된 제1 파티셔닝 방식은 블록을 블록의 좌측-상단 코너로부터 우측-하단 코너로 분할하는 것에 대응하고, 제2 파티션 모드 인덱스는 24와 동일하고, 제2 파티션 모드 인덱스와 연관된 제2 파티셔닝 방식은 블록을 블록의 우측-상단 코너로부터 좌측-하단 코너로 분할하는 것에 대응하는, 방법.

[0416] 11. 조항 5-10 중 어느 하나에 있어서, 복수의 블렌딩 가중치는 엣지 블록에 대한 복수의 루마 가중치 및 엣지 블록에 대한 복수의 크로마 가중치를 포함하고, 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계는:

[0417] 엣지 서브블록에 대한 복수의 루마 가중치에 따라 엣지 블록의 루마 값을 결정하는 단계; 및

[0418] 엣지 서브블록에 대한 복수의 크로마 가중치에 따라 엣지 서브블록의 크로마 값을 결정하는 단계를 더 포함하는, 방법.

[0419] 12. 조항 11에 있어서, 복수의 크로마 가중치 중에서, 크로마 가중치는 크로마 가중치에 대응하는 2x2 블록의 좌측-상단 코너에 대한 루마 가중치를 기초로 결정되는, 방법.

[0420] 13. 조항 4에 있어서,

[0421] 제1 블록의 크기, 타겟 파티셔닝 방식의 각도 값, 타겟 파티셔닝 방식의 거리 값을 기초로 제1 오프셋 및 제2 오프셋을 결정하는 단계; 및

[0422] 제1 오프셋 및 제2 오프셋을 사용하여 제1 블록에서의 루마 샘플에 대한 복수의 블렌딩 가중치를 생성하는 단계를 더 포함하는, 방법.

[0423] 14. 조항 13에 있어서, 제1 오프셋 및 제2 오프셋은 아래의 식을 사용하여 결정되고:

[0424] 제1 오프셋

[0425] $$= \begin{cases} (-W) \gg 1 & A \,\%\, 16 == 8 \text{ 또는 } (A \,\%\, 16 != 0 \text{ 및 } H \geq W) \\ ((-W) \gg 1) + A < 16 \,?\, (D \times W) \gg 3 : -((D \times W) \gg 3) & \textit{그렇지 않은 경우} \end{cases}$$

[0426] 제2 오프셋

[0427] $$= \begin{cases} ((-H) \gg 1) + A < 16 \,?\, (D \times H) \gg 3 : -((D \times H) \gg 3) & A \,\%\, 16 == 8 \text{ 또는 } (A \,\%\, 16 != 0 \text{ 및 } H \geq W) \\ (-H) \gg 1 & \textit{그렇지 않은 경우} \end{cases}$$

[0428] 여기서 W는 제1 블록의 폭을 나타내고, H는 제1 블록의 높이를 나타내고, A는 타겟팅 파티셔닝 방식의 각도 값

을 나타내고, D는 타겟팅 파티셔닝 방식의 거리 값을 나타내는, 방법.

[0429] 15. 조항 3-14 중 어느 하나에 있어서, 제3 표시 신호에 따라, 제1 블록이 결합된 인터 및 인트라 예측(CIIP) 모드를 사용하여 코딩되는지를 결정하는 단계 이전에, 방법은:

[0430] 제1 블록의 크기가 주어진 조건을 충족하는지를 결정하는 단계; 및

[0431] 제1 블록의 크기가 주어진 조건을 충족한다는 결정에 응답하여, 제3 표시 신호를 생성하는 단계, 또는

[0432] 블록의 크기가 주어진 조건을 충족하지 못한다는 결정에 응답하여, 블록이 CIIP 모드를 사용하여 코딩된다고 결정하는 단계를 더 포함하는, 방법.

[0433] 16. 조항 15에 있어서, 주어진 조건은:

[0434] 제1 블록의 폭 및 높이가 각각 8 이상임; 및

[0435] 폭과 높이 사이의 더 큰 값과 폭과 높이 사이의 더 작은 값의 비율이 4 이하임을 포함하는, 방법.

[0436] 17. 비디오 콘텐츠를 처리하기 위한 시스템으로서,

[0437] 명령어의 세트를 저장하는 메모리; 및

[0438] 적어도 하나의 프로세서를 포함하고, 적어도 하나의 프로세서는 시스템이:

[0439] 파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 것;

[0440] 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측을 수행하는 것; 및

[0441] 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 것을 수행하게 하기 위해 명령어의 세트를 실행하도록 구성되는, 시스템.

[0442] 18. 조항 17에 있어서, 파티셔닝 엣지를 따라 복수의 블록을 파티셔닝하는 것에서, 적어도 하나의 프로세서는 시스템이:

[0443] 복수의 블록의 파티션 모드를 결정하는 것; 및

[0444] 파티션 모드를 기초로 복수의 블록을 파티셔닝하는 것을 더 수행하게 하기 위해 명령어의 세트를 실행하도록 구성되는, 시스템.

[0445] 19. 조항 18에 있어서, 복수의 블록은 제1 블록의 서브블록이고, 복수의 블록에 대한 파티션 모드를 결정하는 것에서, 적어도 하나의 프로세서는 시스템이:

[0446] 제1 표시 신호에 따라, 제1 블록이 서브블록 병합 모드를 사용하여 코딩되는지를 결정하는 것;

[0447] 제1 블록이 서브블록 병합 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 제2 표시 신호에 따라, 제1 블록이 정규 모드 또는 움직임 벡터 차분을 갖는 병합 모드(MMVD) 중 하나를 사용하여 코딩되는지를 결정하는 것;

[0448] 제1 블록이 정규 모드 또는 MMVD 중 하나를 사용하여 코딩되지 않는다는 결정에 응답하여, 제3 표시 신호에 따라, 제1 블록이 결합된 인터 및 인트라 예측(CIIP) 모드를 사용하여 코딩되는지를 결정하는 것; 및

[0449] 제1 블록이 CIIP 모드를 사용하여 코딩되지 않는다는 결정에 응답하여, 복수의 블록의 파티션 모드가 삼각형 파티션 모드 또는 기하학적 파티션 모드 중 하나라고 결정하는 것을 더 수행하게 하기 위해 명령어의 세트를 실행하도록 구성되는, 시스템.

[0450] 20. 컴퓨터 시스템의 적어도 하나의 프로세서에 의해 실행가능한 명령어를 저장하는 비일시적 컴퓨터 판독가능 매체로서, 명령어의 실행은 컴퓨터 시스템이:

[0451] 파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝하는 단계;

[0452] 제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해, 복수의 블록에 대해 인터 예측을 수행하는 단계; 및

[0453] 파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 예측 신호 및 제2 예측 신호를 블렌딩하는 단계를 포함하는 방법을 수행하게 하는, 비일시적 컴퓨터 판독가능 매체.

[0454] 본원에서 "제1" 및 "제2"와 같은 관계 용어는 엔티티 또는 동작을 다른 엔티티 또는 동작과 구분하기 위해서만 사용되며, 이러한 엔티티 또는 동작 간의 실제 관계 또는 시퀀스를 요구하거나 암시하지 않는다. 더욱이, 포함하는(comprising), "갖는(having)", "포함하는(containing)" 및 "포함하는(including)"이라는 단어 및 다른 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 항목들만으로 제한되는 것을 의미하지 않는다는 점에서, 제약을 두지 않는 것으로(open-ended) 의도된다.

[0455] 본원에서 사용된 바와 같이, 달리 구체적으로 언급되지 않는 한, "또는"이란 용어는 실행 불가능한 경우를 제외하고 모든 가능한 조합을 포함한다. 예를 들어, 데이터베이스가 A 또는 B를 포함할 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 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.

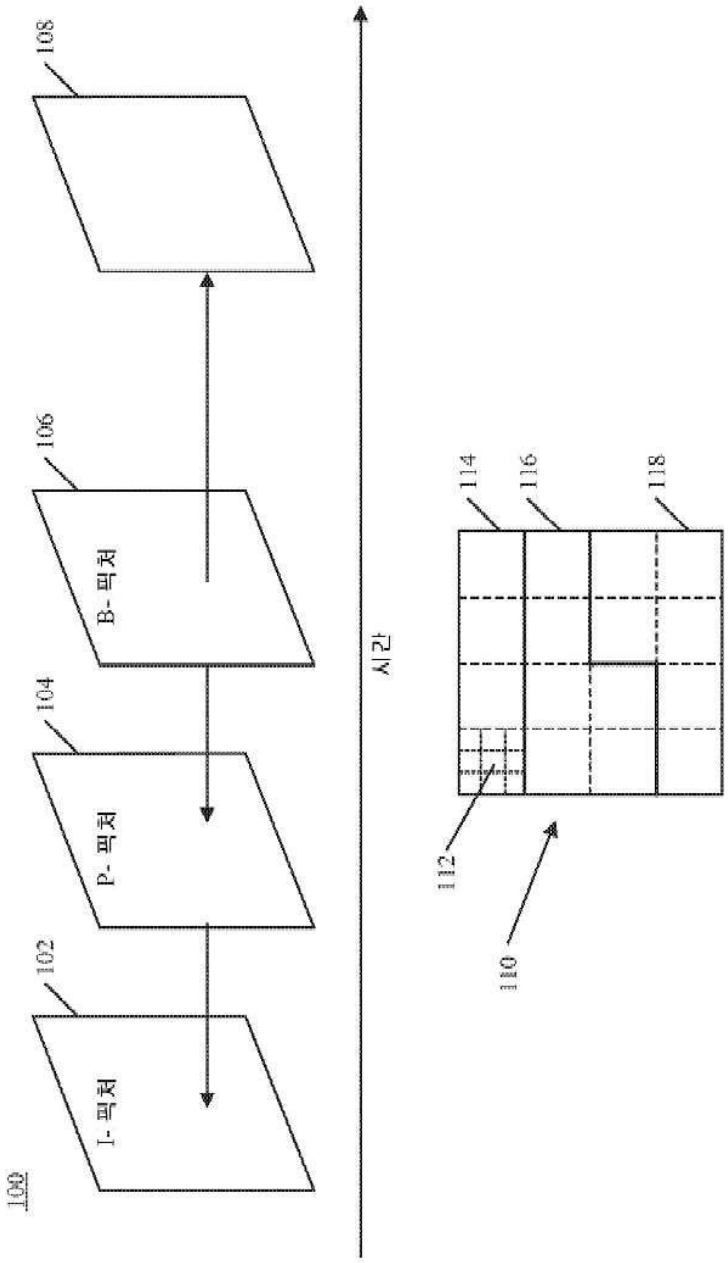
[0456] 상술한 실시예는 하드웨어 또는 소프트웨어(프로그램 코드), 또는 하드웨어 및 소프트웨어의 조합에 의해 구현될 수 있다는 것이 인식된다. 소프트웨어로 구현되는 경우, 이는 상술한 컴퓨터 판독 가능 매체에 저장될 수 있다. 소프트웨어는 프로세서에 의해 실행될 때 개시된 방법을 수행할 수 있다. 본 개시에서 설명된 컴퓨팅 유닛 및 다른 기능 유닛은 하드웨어 또는 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합에 의해 구현될 수 있다. 통상의 기술자는 또한 위에서 설명된 모듈/유닛 중 다수가 하나의 모듈/유닛으로 결합될 수 있고, 위에서 설명된 모듈/유닛 각각이 복수의 서브-모듈/서브-유닛으로 더 분할될 수 있음을 이해할 것이다.

[0457] 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 개시의 명세서 및 실시를 고려하여 통상의 기술자에게 명백할 수 있다. 명세서 및 예시는 단지 예시적인 것으로 간주되어야 하는 것으로 의도되며, 본 개시의 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.

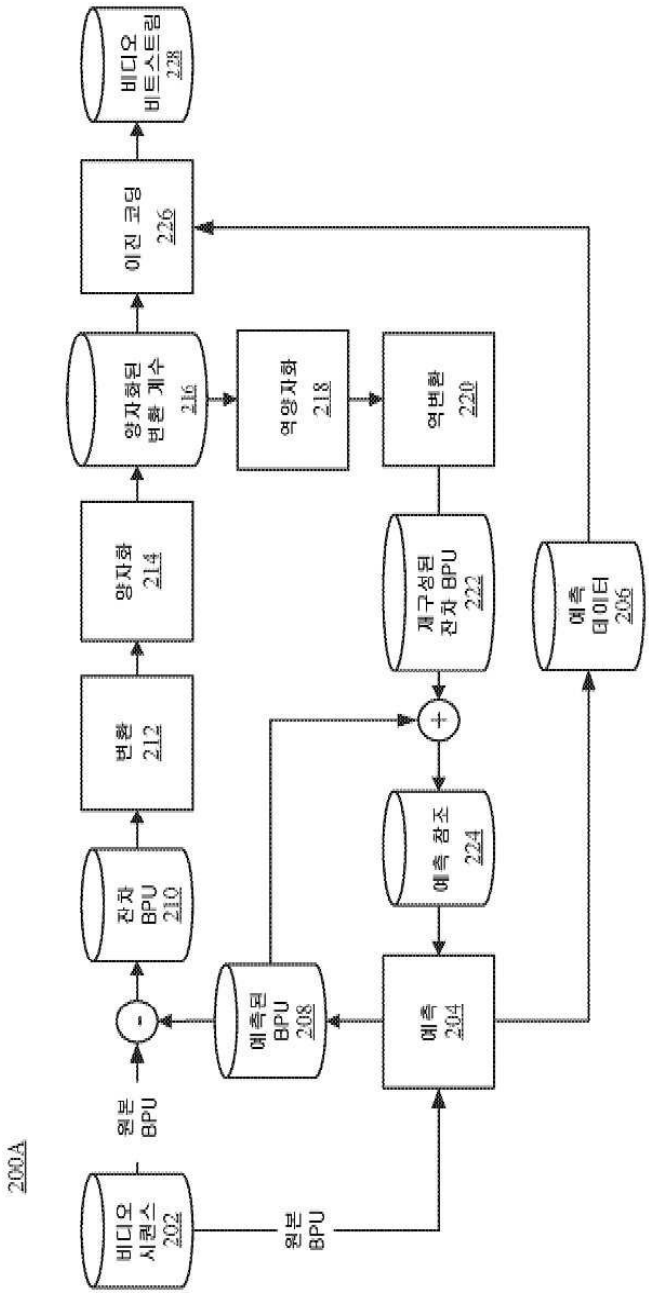
[0458] 도면 및 명세서에서, 예시적인 실시예가 개시된다. 하지만, 이들 실시예에 대해 다수의 변형 및 수정이 이루어질 수 있다. 따라서, 특정한 용어가 이용되더라도, 제한의 목적이 아닌 일반적이고 설명적인 의미로 사용된다.

도면

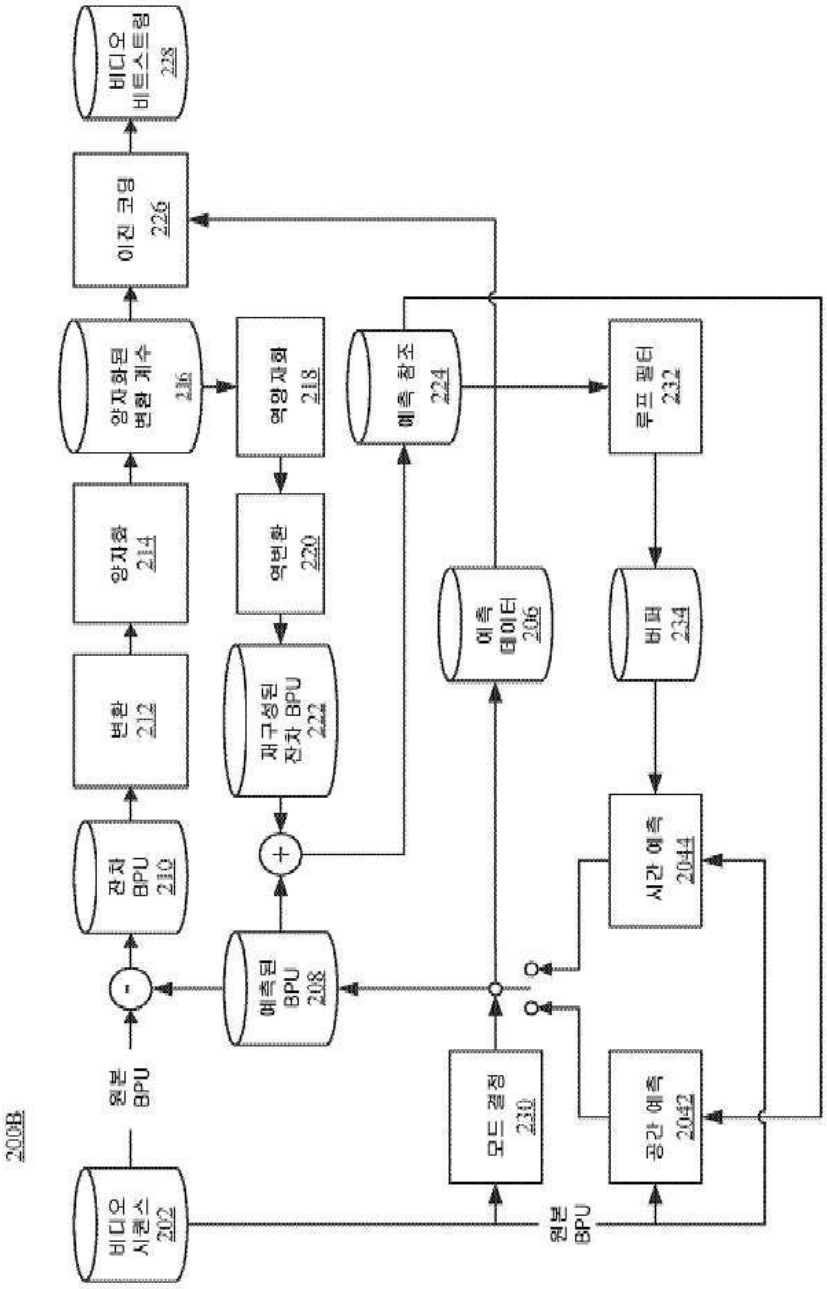
도면1



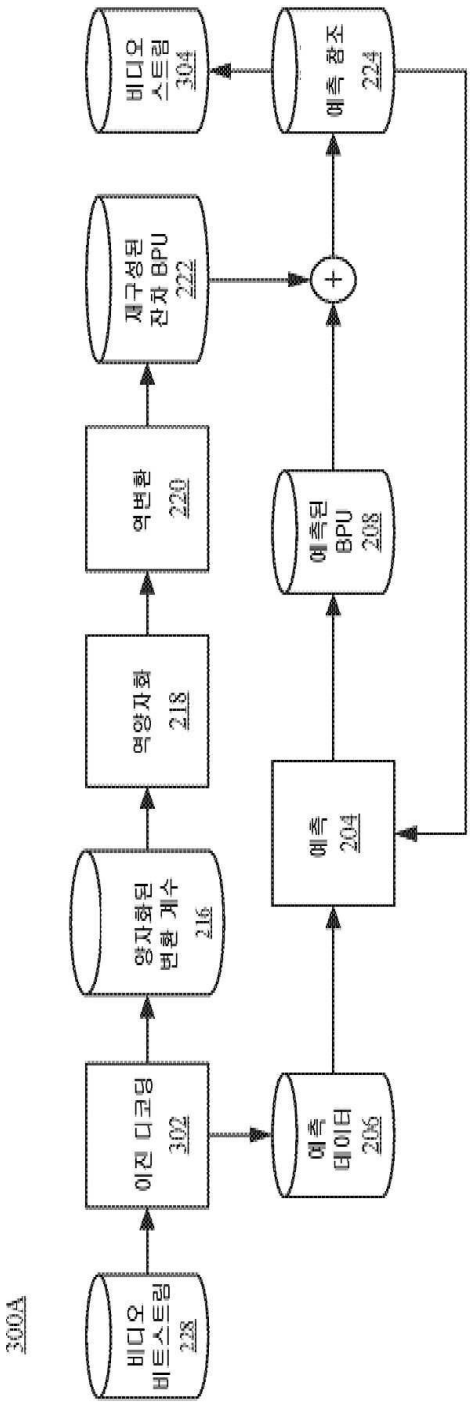
도면2a


200A
비디오 시퀀스 202
원본 BPU
원본 BPU
-
잔차 BPU 210
변환 212
양자화 214
양자화된 변환 계수 216
역양자화 218
역변환 220
+
재구성된 잔차 BPU 222
예측 참조 224
예측 BPU 208
예측 204
이진 코딩 226
비디오 비트스트림 228
예측 데이터 206

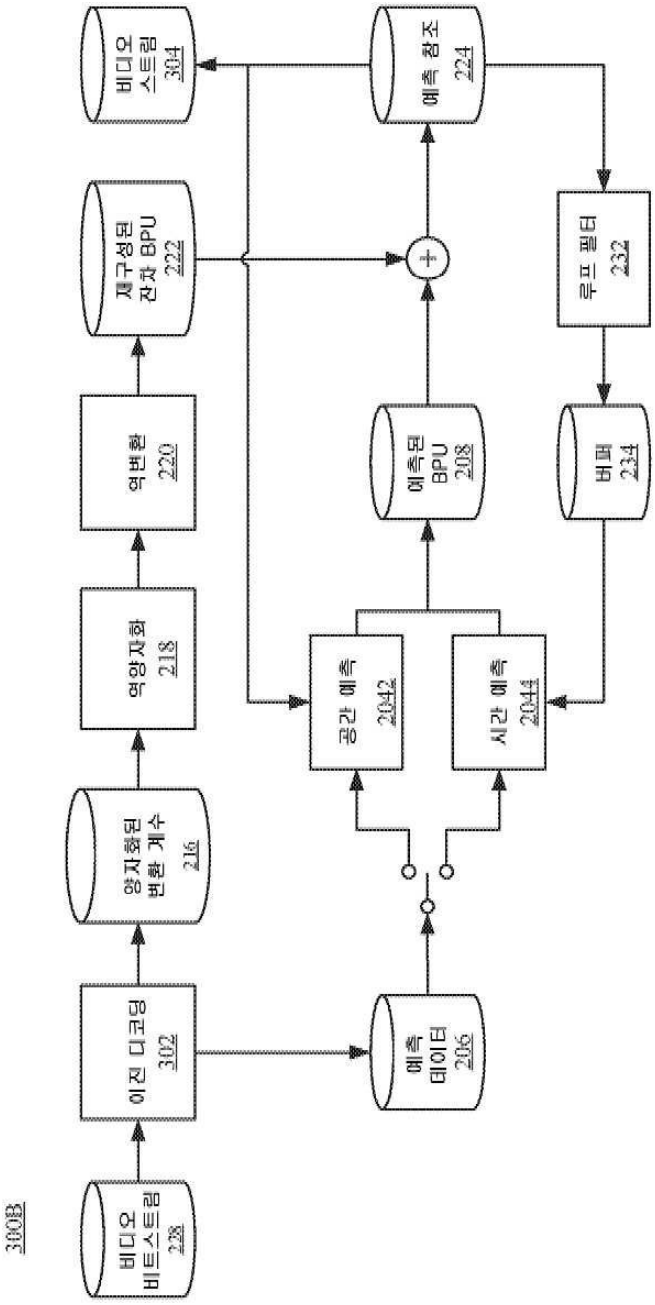
도면2b


200B
비디오 시퀀스 202
원본 BPU
-
+
변환 212
양자화 214
엔트로피 인코딩 226
비디오 비트스트림 228
양자화된 변환 계수 236
역양자화 218
역변환 220
재구성된 잔차 BPU 222
+
예측 참조 224
루프 필터 232
버퍼 234
예측 데이터 206
예측된 BPU 208
관계 BPU 210
모드 결정 230
공간 예측 2042
시간 예측 2044
원본 BPU

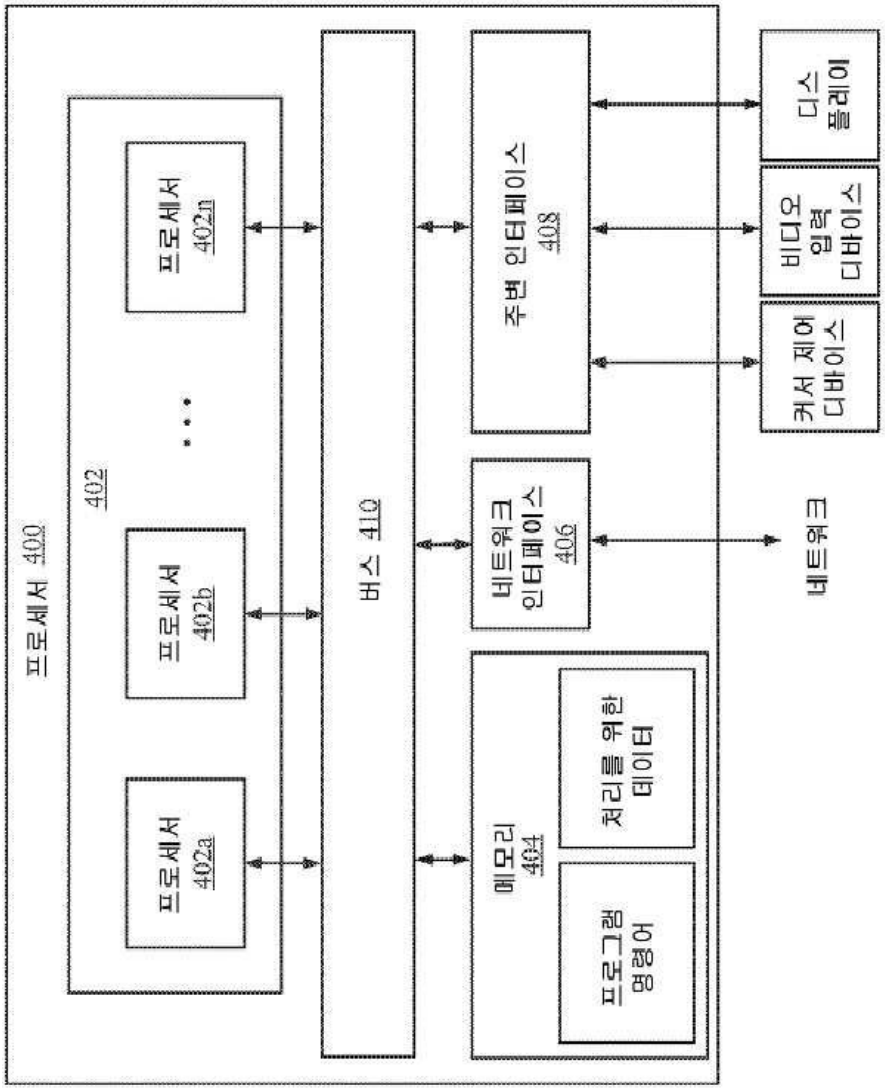
도면3a


300A
비디오 비트스트림 228
이진 디코딩 302
양자화된 변환 계수 216
역양자화 218
역변환 220
재구성된 잔차 BPU 222
비디오 스트림 304
예측 데이터 206
예측 204
예측된 BPU 208
예측 참조 224

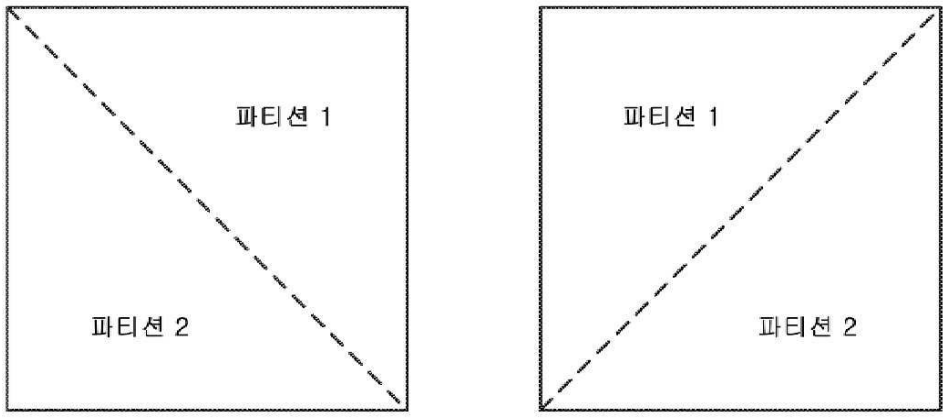
도면3b


300B
비디오 비트스트림 228
이진 디코딩 302
양자화된 변환 계수 216
역양자화 218
역변환 220
재구성된 잔차 BPU 222
비디오 스트림 304
예측 데이터 206
공간 예측 2042
시간 예측 2044
예측된 BPU 208
예측 참조 224
루프 필터 232
버퍼 234

도면4



도면5

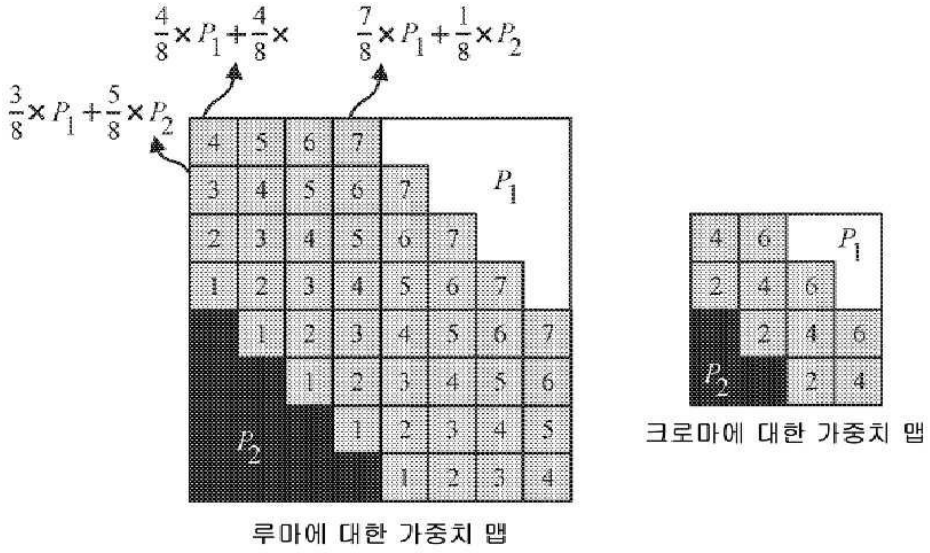


삼각형 파티션 기반 인터 예측

도면6

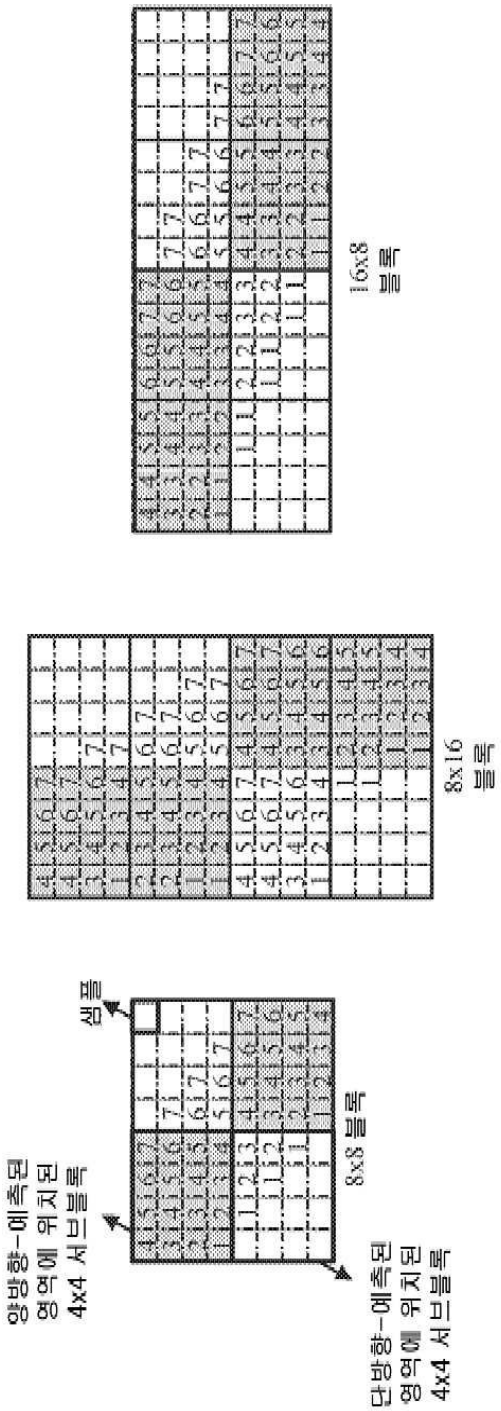
병합 인덱스	L0 MV	L1 MV
0	X	
1		X
2	X	
3		X
4	X	

도면7



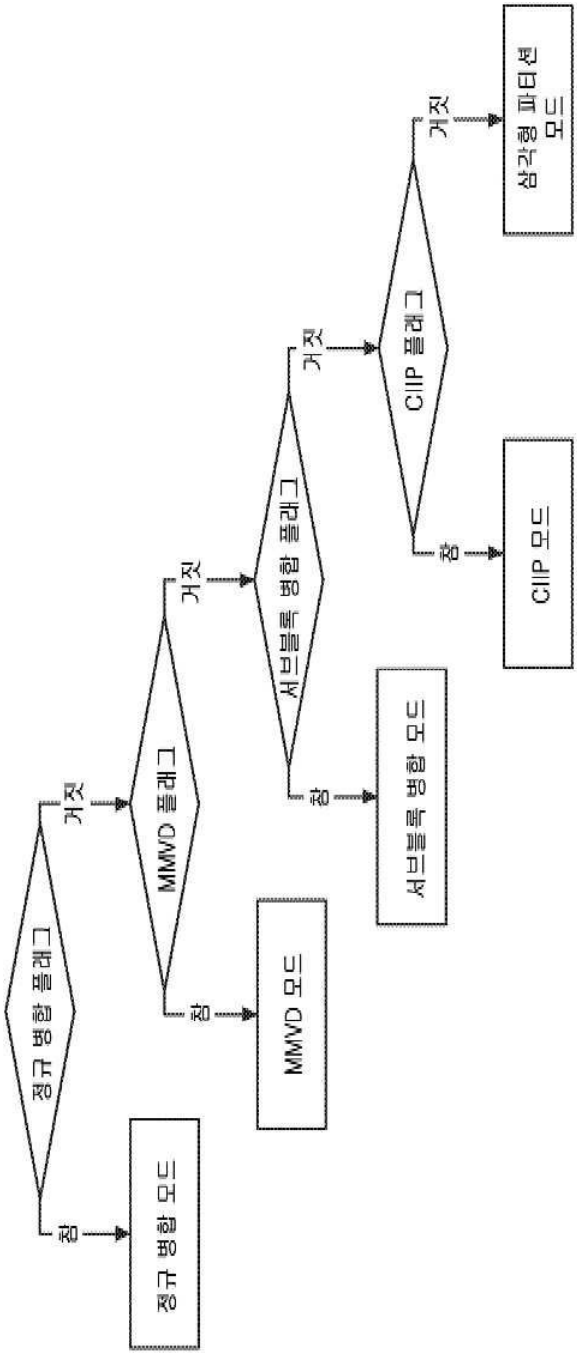
도 7: 블렌딩 프로세스에서 사용되는 가중치

도면8



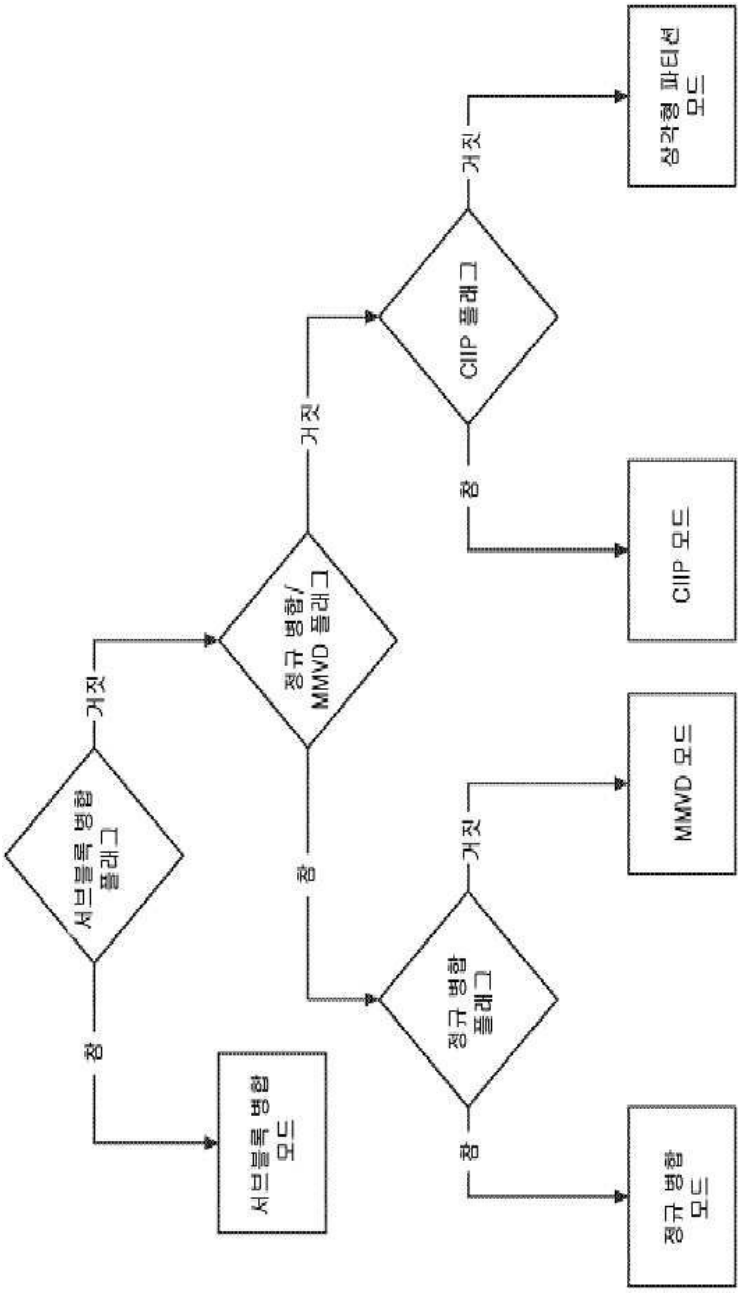
도 8: 단방향-예측된 또는 양방향-예측된 영역에 위치된 4x4 서브블록의 예시

도면9



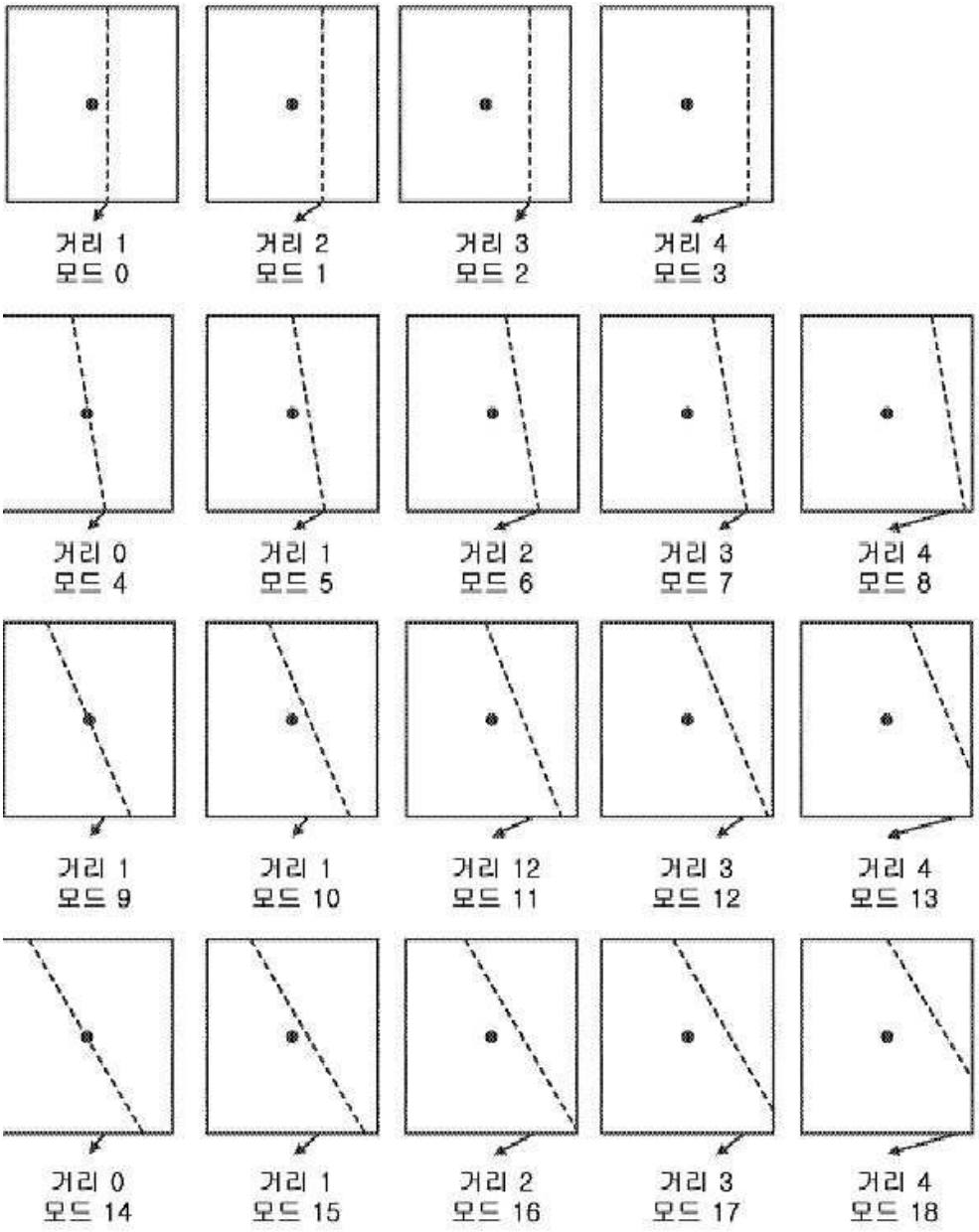
도 9: 예시적인 신택스 구조

도면10



도 10: 다른 예시적인 신택스 구조

도면11



도 11: 기하학적 파티션의 예시

도면12

테이블 12: Dis[]에 대한 룩업 테이블

idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Dis[idx]	64	63	59	53	45	36	24	12	0	-12	-24	-36	-45	-53	-59	-63
idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Dis[idx]	-64	-63	-59	-53	-45	-36	-24	-12	0	12	24	36	45	53	59	63

도면13

테이블 13A: GeoFilter[]에 대한 룩업 테이블

Idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
GeoFilter[idx]	4	4	5	5	5	5	6	6	6	6	7	7	7	7	8

테이블 13B: GeoFilter[]에 대한 룩업 테이블

Idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13
GeoFilter[idx]	4	4	4	4	5	5	5	5	5	5	5	6	6	6
Idx	14	15	16	17	18	19	20	21	22	23	24	25	26	
GeoFilter[idx]	6	6	6	6	7	7	7	7	7	7	7	7	8	

도면14a

테이블 14: angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	0	4
4	1	0
5	1	1
6	1	2
7	1	3
8	1	4
9	2	0
10	2	1
11	2	2
12	2	3
13	2	4
14	3	0
15	3	1
16	3	2
17	3	3
18	3	4
19	4	1
20	4	2
21	4	3
22	4	4
23	5	0
24	5	1
25	5	2
26	5	3
27	5	4
28	6	0
29	6	1
30	6	2
31	6	3
32	6	4
33	7	0
34	7	1
35	7	2
36	7	3
37	7	4
38	8	1
39	8	2

도면14b

테이블 14 - 계속: angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
40	8	3
41	8	4
42	9	0
43	9	1
44	9	2
45	9	3
46	9	4
47	10	0
48	10	1
49	10	2
50	10	3
51	10	4
52	11	0
53	11	1
54	11	2
55	11	3
56	11	4
57	12	1
58	12	2
59	12	3
60	12	4
61	13	0
62	13	1
63	13	2
64	13	3
65	13	4
66	14	0
67	14	1
68	14	2
69	14	3
70	14	4
71	15	0
72	15	1
73	15	2
74	15	3
75	15	4
76	16	1
77	16	2
78	16	3
79	16	4
80	17	1

도면14c

테이블 14 - 계속: angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
81	17	2
82	17	3
83	17	4
84	18	1
85	18	2
86	18	3
87	18	4
88	19	1
89	19	2
90	19	3
91	19	4
92	20	1
93	20	2
94	20	3
95	20	4
96	21	1
97	21	2
98	21	3
99	21	4
100	22	1
101	22	2
102	22	3
103	22	4
104	23	1
105	23	2
106	23	3
107	23	4
108	24	1
109	24	2
110	24	3
111	24	4
112	25	1
113	25	2
114	25	3
115	25	4
116	26	1
117	26	2
118	26	3
119	26	4
120	27	1

도면14d

테이블 14 - 계속: angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
121	27	2
122	27	3
123	27	4
124	28	1
125	28	2
126	28	3
127	28	4
128	29	1
129	29	2
130	29	3
131	29	4
132	30	1
133	30	2
134	30	3
135	30	4
136	31	1
137	31	2
138	31	3
139	31	4

도면15aa

테이블 15A: stepDis에 대한 룩업 테이블

whRatio	angleN	stepDis
0	0	0
0	1	18
0	2	31
0	3	40
0	4	42
0	5	40
0	6	31
0	7	18
0	8	0
1	0	0
1	1	38
1	2	71
1	3	97
1	4	115
1	5	125
1	6	126
1	7	118
1	8	102
2	0	0
2	1	78
2	2	149
2	3	210
2	4	260
2	5	295
2	6	315
2	7	319
2	8	307
3	0	0
3	1	158
3	2	306
3	3	438
3	4	549
3	5	636
3	6	694
3	7	721
3	8	717

도면15ab

테이블 15A - 계속: stepDis에 대한 룩업 테이블

whRatio	angleN	stepDis
4	0	0
4	1	318
4	2	619
4	3	893
4	4	1129
4	5	1317
4	6	1450
4	7	1524
4	8	1536

도면15ba

테이블 15B: stepDis에 대한 룩업 테이블

whRatio	angleN	stepDis
0	0	0
0	1	18
0	2	31
0	3	39
0	4	42
0	5	39
0	6	31
0	7	18
0	8	0
1	0	0
1	1	38
1	2	70
1	3	96
1	4	115
1	5	125
1	6	126
1	7	118
1	8	102
2	0	0
2	1	78
2	2	149
2	3	210
2	4	259
2	5	295
2	6	315
2	7	319
2	8	307
3	0	0
3	1	158
3	2	305
3	3	438
3	4	549
3	5	635
3	6	693
3	7	721
3	8	717
4	0	0
4	1	317

도면15bb

테이블 15B - 계속: stepDis에 대한 룩업 테이블

whRatio	angleN	stepDis
4	2	619
4	3	893
4	4	1128
4	5	1317
4	6	1450
4	7	1524
4	8	1536

도면15ca

테이블 15C: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	0	4
4	1	0
5	1	1
6	1	2
7	1	3
8	1	4
9	2	0
10	2	1
11	2	2
12	2	3
13	2	4
14	3	0
15	3	1
16	3	2
17	3	3
18	3	4
19	4	1
20	4	2
21	4	3
22	4	4
23	5	0
24	5	1
25	5	2
26	5	3
27	5	4
28	6	0
29	6	1
30	6	2
31	6	3
32	6	4
33	7	0
34	7	1
35	7	2

도면15cb

테이블 15C - 계속: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
35	7	2
36	7	3
37	7	4
38	8	1
39	8	2
40	8	3
41	8	4
42	9	0
43	9	1
44	9	2
45	9	3
46	9	4
47	10	0
48	10	1
49	10	2
50	10	3
51	10	4
52	11	0
53	11	1
54	11	2
55	11	3
56	11	4
57	12	1
58	12	2
59	12	3
60	12	4
61	13	0
62	13	1
63	13	2
64	13	3
65	13	4
66	14	0
67	14	1
68	14	2
69	14	3
70	14	4

도면15cc

테이블 15C - 계속: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
71	15	0
72	15	1
73	15	2
74	15	3
75	15	4
76	16	1
77	16	2
78	16	3
79	16	4
80	17	1
81	17	2
82	17	3
83	17	4
84	18	1
85	18	2
86	18	3
87	18	4
88	19	1
89	19	2
90	19	3
91	19	4
92	20	1
93	20	2
94	20	3
95	20	4
96	21	1
97	21	2
98	21	3
99	21	4
100	22	1
101	22	2
102	22	3
103	22	4
104	23	1
105	23	2

도면15cd

테이블 15C - 계속: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
106	23	3
107	23	4
108	24	1
109	24	2
110	24	3
111	24	4
112	25	1
113	25	2
114	25	3
115	25	4
116	26	1
117	26	2
118	26	3
119	26	4
120	27	1
121	27	2
122	27	3
123	27	4
124	28	1
125	28	2
126	28	3
127	28	4
128	29	1
129	29	2
130	29	3
131	29	4
132	30	1
133	30	2
134	30	3
135	30	4
136	31	1
137	31	2
138	31	3
139	31	4

도면15da

테이블 15D: 기하학적 파티셔닝 서브모드의 총 수가 108로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	1	0
4	1	1
5	1	2
6	1	3
7	2	0
8	2	1
9	2	2
10	2	3
11	3	0
12	3	1
13	3	2
14	3	3
15	4	1
16	4	2
17	4	3
18	5	0
19	5	1
20	5	2
21	5	3
22	6	0
23	6	1
24	6	2
25	6	3
26	7	0
27	7	1
28	7	2
29	7	3
30	8	1
31	8	2
32	8	3
33	9	0
34	9	1
35	9	2

도면15db

테이블 15D - 계속: 기하학적 파티셔닝 서브모드의 총 수가 108로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
36	9	3
37	10	0
38	10	1
39	10	2
40	10	3
41	11	0
42	11	1
43	11	2
44	11	3
45	12	1
46	12	2
47	12	3
48	13	0
49	13	1
50	13	2
51	13	3
52	14	0
53	14	1
54	14	2
55	14	3
56	15	0
57	15	1
58	15	2
59	15	3
60	16	1
61	16	2
62	16	3
63	17	1
64	17	2
65	17	3
66	18	1
67	18	2
68	18	3
69	19	1
70	19	2

도면15dc

테이블 15D - 계속: 기하학적 파티셔닝 서브모드의 총 수가 108로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
71	19	3
72	20	1
73	20	2
74	20	3
75	21	1
76	21	2
77	21	3
78	22	1
79	22	2
80	22	3
81	23	1
82	23	2
83	23	3
84	24	1
85	24	2
86	24	3
87	25	1
88	25	2
89	25	3
90	26	1
91	26	2
92	26	3
93	27	1
94	27	2
95	27	3
96	28	1
97	28	2
98	28	3
99	29	1
100	29	2
101	29	3
102	30	1
103	30	2
104	30	3
105	31	1
106	31	2
107	31	3

도면15ea

테이블 15E: 기하학적 파티셔닝 서브모드의 총 수가 80으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	1	0
4	1	1
5	1	2
6	1	3
7	2	0
8	2	1
9	2	2
10	2	3
11	3	0
12	3	1
13	3	2
14	3	3
15	4	1
16	4	2
17	4	3
18	5	0
19	5	1
20	5	2
21	5	3
22	8	1
23	8	2
24	8	3
25	11	0
26	11	1
27	11	2
28	11	3
29	12	1
30	12	2
31	12	3
32	13	0
33	13	1
34	13	2
35	13	3
36	14	0
37	14	1
38	14	2
39	14	3

도면15eb

테이블 15E - 계속: 기하학적 파티셔닝 서브모드의 총 수가 80으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
40	15	0
41	15	1
42	15	2
43	15	3
44	16	1
45	16	2
46	16	3
47	17	1
48	17	2
49	17	3
50	18	1
51	18	2
52	18	3
53	19	1
54	19	2
55	19	3
56	20	1
57	20	2
58	20	3
59	21	1
60	21	2
61	21	3
62	24	1
63	24	2
64	24	3
65	27	1
66	27	2
67	27	3
68	28	1
69	28	2
70	28	3
71	29	1
72	29	2
73	29	3
74	30	1
75	30	2
76	30	3
77	31	1
78	31	2
79	31	3

도면15fa

테이블 15F: 기하학적 파티셔닝 서브모드의 총 수가 64로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

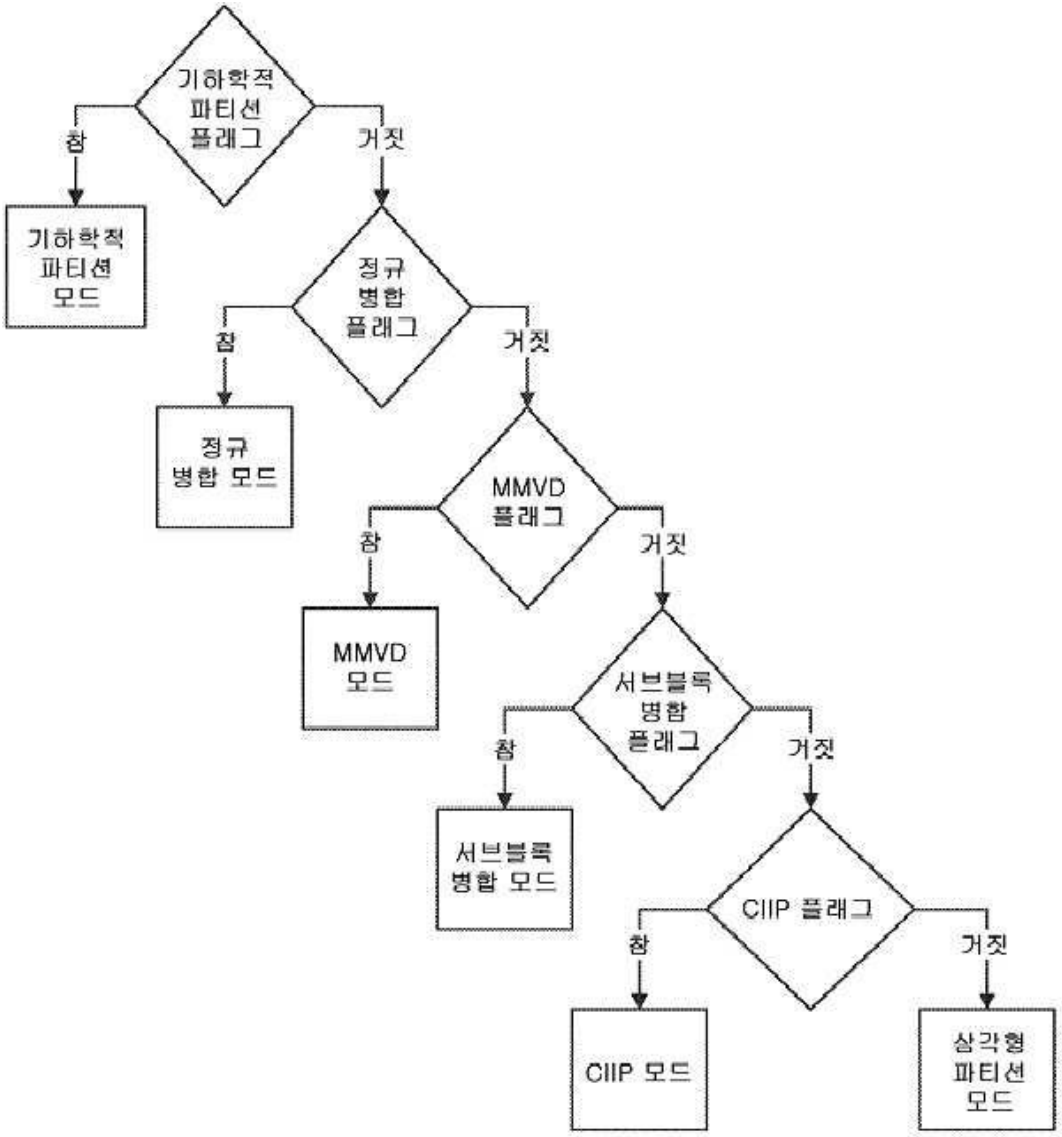
geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	16	2
3	16	1
4	17	3
5	17	2
6	17	1
7	1	0
8	1	1
9	1	2
10	1	3
11	31	3
12	31	2
13	31	1
14	15	0
15	15	1
16	15	2
17	15	3
18	18	3
19	18	2
20	18	1
21	2	0
22	2	1
23	2	2
24	2	3
25	30	3
26	30	2
27	30	1
28	14	0
29	14	1
30	14	2
31	14	3
32	19	3
33	19	2
34	19	1
35	3	0

도면15fb

테이블 15F - 계속: 기하학적 파티셔닝 서브모드의 총 수가 64로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

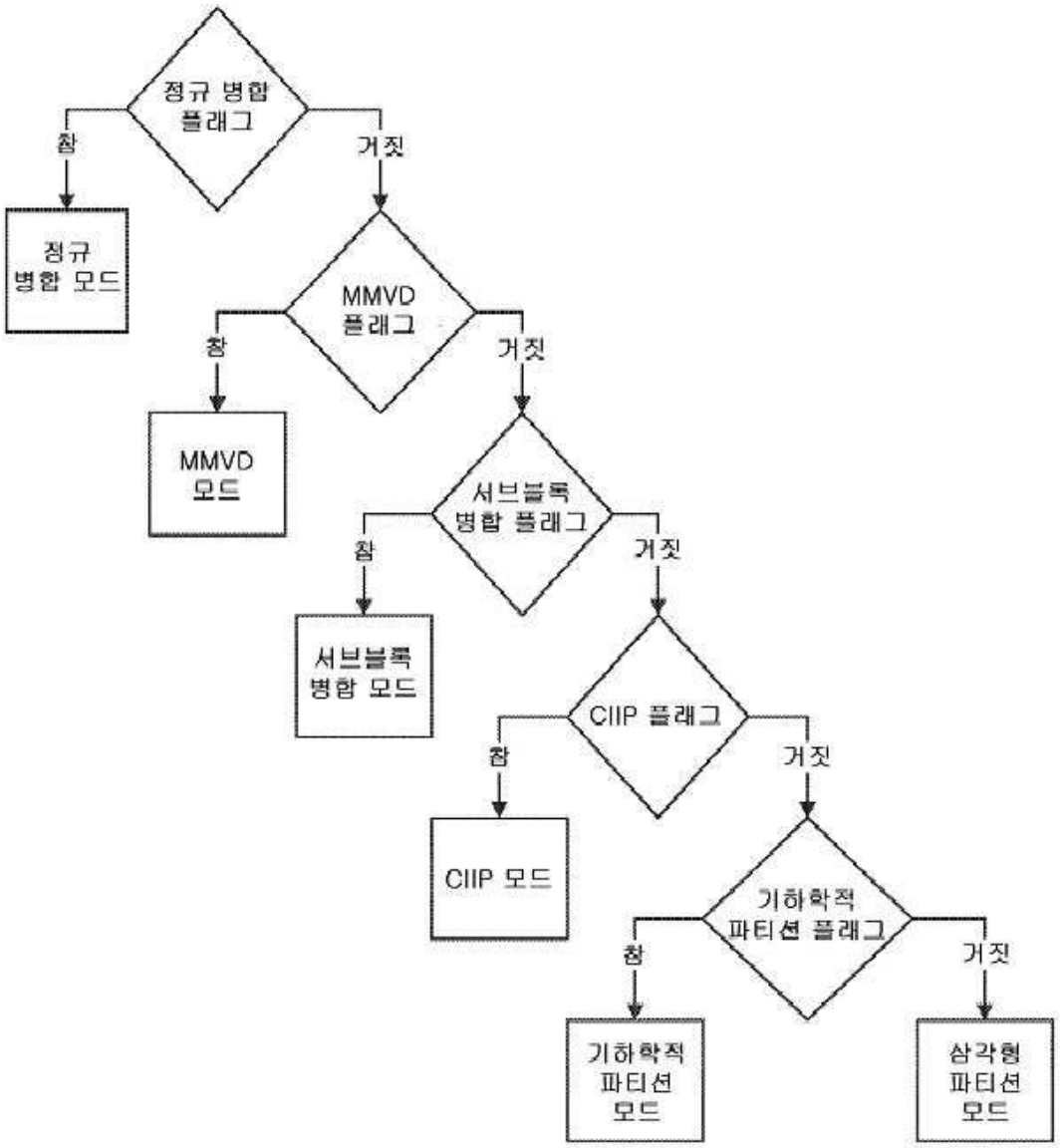
geo_partition_idx	angleIdx	distanceIdx
36	3	1
37	3	2
38	3	3
39	29	3
40	29	2
41	29	1
42	13	0
43	13	1
44	13	2
45	13	3
46	21	3
47	21	2
48	21	1
49	5	0
50	5	1
51	5	2
52	5	3
53	27	3
54	27	2
55	27	1
56	11	0
57	11	1
58	11	2
59	11	3
60	8	1
61	8	2
62	24	2
63	24	1

도면16



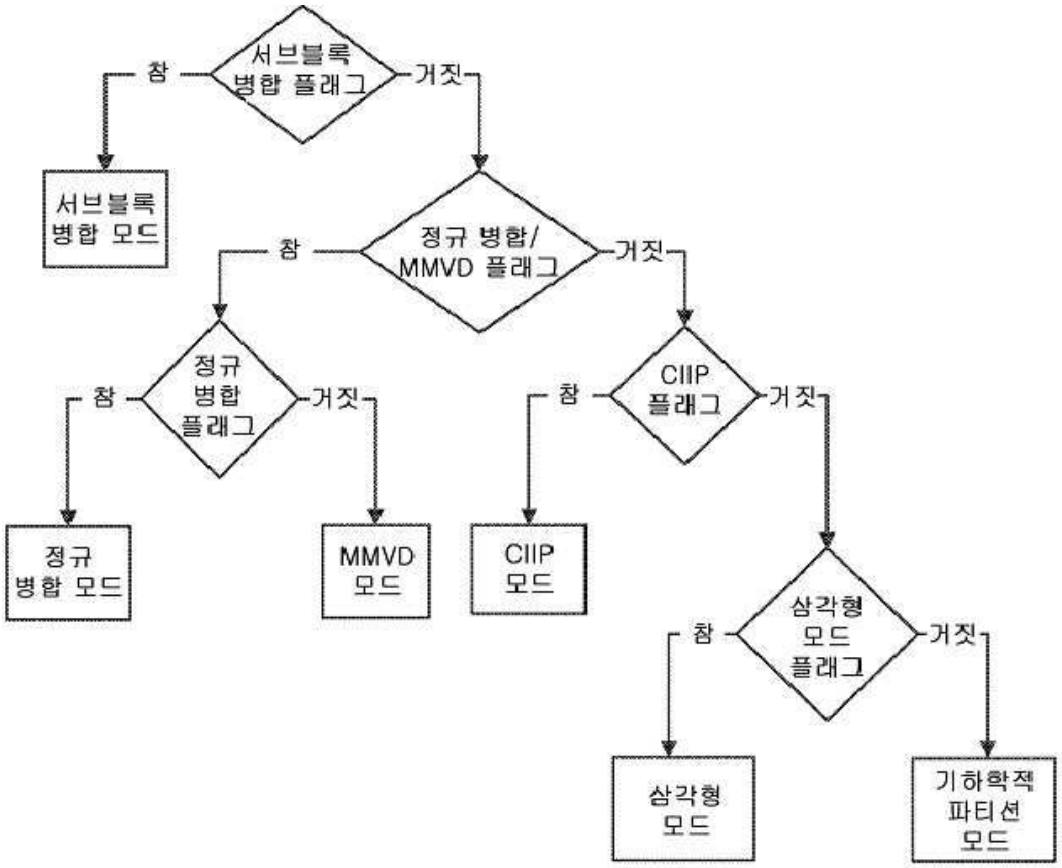
도 16: 기하학적 파티션 모드에 대한 예시적인 신택스 구조

도면17a



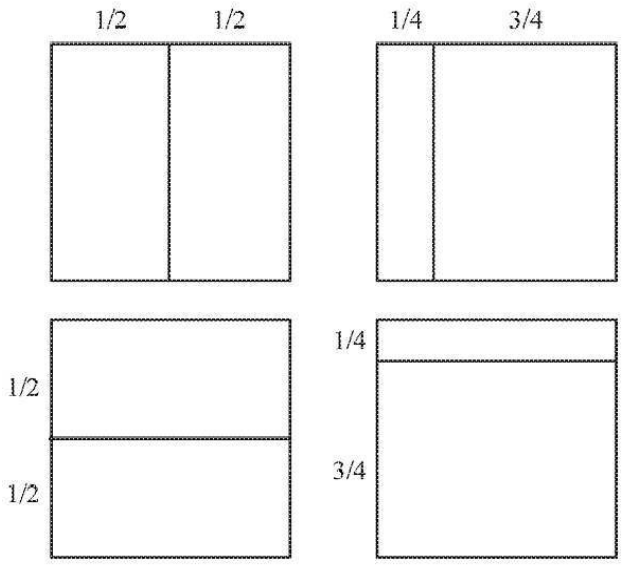
도 17a: 기하학적 파티션 모드에 대한 예시적인 신택스 구조

도면17b



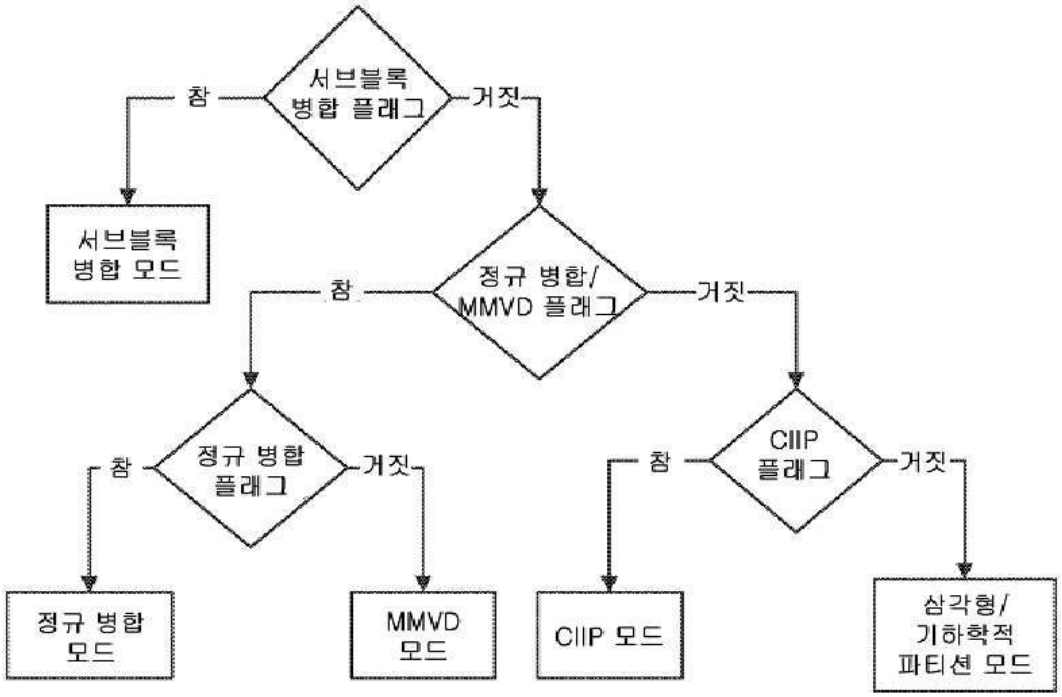
도 17b: 기하학적 파티션 모드에 대한 예시적인 파티션 구조

도면18



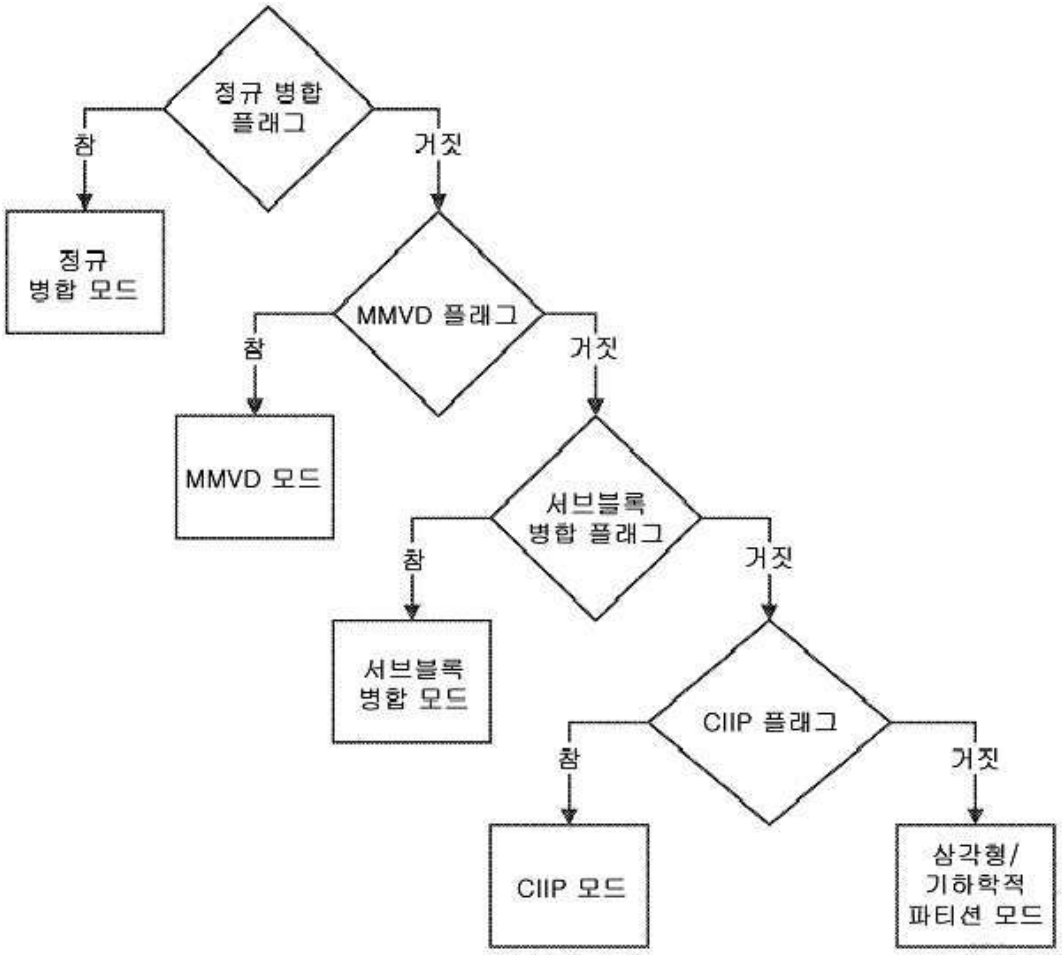
도 18: 인터 예측 블록에 대한 서브블록 변환

도면19



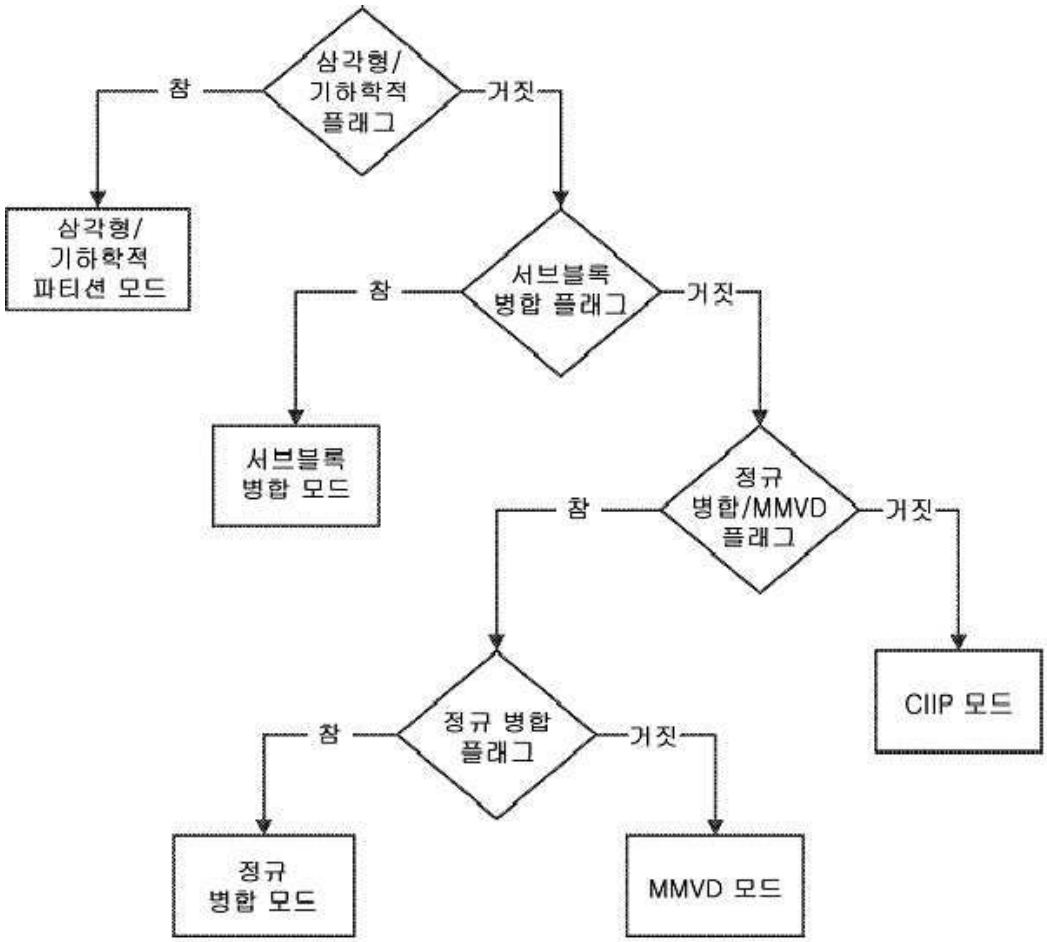
도 19: 통합된 신택스 구조의 예시

도면20



도 20: 통합된 신택스 구조의 예시

도면21



도 21: 통합된 신택스 구조의 예시

도면22aa

테이블 22A: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	0	4
4	1	0
5	1	1
6	1	2
7	1	3
8	1	4
9	2	0
10	2	1
11	2	2
12	2	3
13	2	4
14	3	0
15	3	1
16	3	2
17	3	3
18	3	4
19	4	0
20	4	1
21	4	2
22	4	3
23	4	4
24	5	0
25	5	1
26	5	2
27	5	3
28	5	4
29	6	0
30	6	1
31	6	2
32	6	3
33	6	4
34	7	0
35	7	1

도면22ab

테이블 22A - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
36	7	2
37	7	3
38	7	4
39	8	1
40	8	2
41	8	3
42	8	4
43	9	0
44	9	1
45	9	2
46	9	3
47	9	4
48	10	0
49	10	1
50	10	2
51	10	3
52	10	4
53	11	0
54	11	1
55	11	2
56	11	3
57	11	4
58	12	0
59	12	1
60	12	2
61	12	3
62	12	4
63	13	0
64	13	1
65	13	2
66	13	3
67	13	4
68	14	0
69	14	1
70	14	2

도면22ac

테이블 22A - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
71	14	3
72	14	4
73	15	0
74	15	1
75	15	2
76	15	3
77	15	4
78	16	1
79	16	2
80	16	3
81	16	4
82	17	1
83	17	2
84	17	3
85	17	4
86	18	1
87	18	2
88	18	3
89	18	4
90	19	1
91	19	2
92	19	3
93	19	4
94	20	1
95	20	2
96	20	3
97	20	4
98	21	1
99	21	2
100	21	3
101	21	4
102	22	1
103	22	2
104	22	3
105	22	4
106	23	1
107	23	2
108	23	3
109	23	4
110	24	1

도면22ad

테이블 22A - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

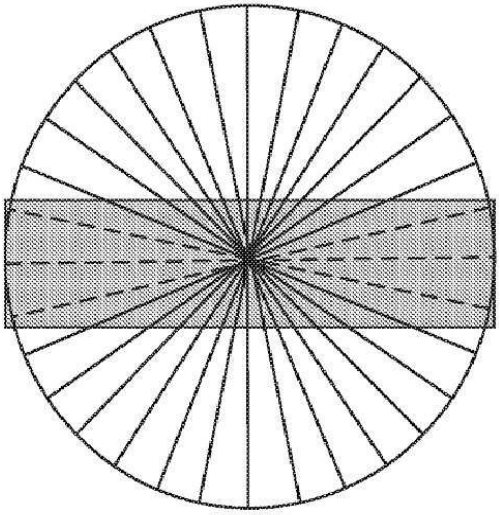
geo_partition_idx	angleIdx	distanceIdx
111	24	2
112	24	3
113	24	4
114	25	1
115	25	2
116	25	3
117	25	4
118	26	1
119	26	2
120	26	3
121	26	4
122	27	1
123	27	2
124	27	3
125	27	4
126	28	1
127	28	2
128	28	3
129	28	4
130	29	1
131	29	2
132	29	3
133	29	4
134	30	1
135	30	2
136	30	3
137	30	4
138	31	1
139	31	2
140	31	3
141	31	4

도면22b

테이블 22B: 상각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
angleIdx	0	0	2	2	2	2	3	3	3	3	4	4	4	4	5	5
distanceIdx	1	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1
geo_partition_idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
angleIdx	5	5	8	8	11	11	11	11	12	12	12	12	13	13	13	13
distanceIdx	2	3	1	3	0	1	2	3	0	1	2	3	0	1	2	3
geo_partition_idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
angleIdx	14	14	14	14	16	16	18	18	18	19	19	19	20	20	20	21
distanceIdx	0	1	2	3	1	3	1	2	3	1	2	3	1	2	3	1
geo_partition_idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
angleIdx	21	21	24	24	27	27	27	28	28	28	29	29	29	30	30	30
distanceIdx	2	3	1	3	1	2	3	1	2	3	1	2	3	1	2	3

도면23



도 23: 더 큰 블록 치수만을 파티셔닝하는 각도를 허용하는 예시

도면24a

테이블 24: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	1	0
1	2	0
2	3	0
3	4	0
4	5	0
5	6	0
6	7	0
7	9	0
8	10	0
9	11	0
10	12	0
11	13	0
12	14	0
13	15	0
14	0	1
15	1	1
16	2	1
17	3	1
18	4	1
19	5	1
20	6	1
21	7	1
22	8	1
23	9	1
24	10	1
25	11	1
26	12	1
27	13	1
28	14	1
29	15	1
30	16	1
31	17	1
32	18	1
33	19	1
34	20	1
35	21	1

도면24b

테이블 24 - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
36	22	1
37	23	1
38	24	1
39	25	1
40	26	1
41	27	1
42	28	1
43	29	1
44	30	1
45	31	1
46	0	2
47	1	2
48	2	2
49	3	2
50	4	2
51	5	2
52	6	2
53	7	2
54	8	2
55	9	2
56	10	2
57	11	2
58	12	2
59	13	2
60	14	2
61	15	2
62	16	2
63	17	2
64	18	2
65	19	2
66	20	2
67	21	2
68	22	2
69	23	2
70	24	2

도면24c

테이블 24 - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
71	25	2
72	26	2
73	27	2
74	28	2
75	29	2
76	30	2
77	31	2
78	0	3
79	1	3
80	2	3
81	3	3
82	4	3
83	5	3
84	6	3
85	7	3
86	8	3
87	9	3
88	10	3
89	11	3
90	12	3
91	13	3
92	14	3
93	15	3
94	16	3
95	17	3
96	18	3
97	19	3
98	20	3
99	21	3
100	22	3
101	23	3
102	24	3
103	25	3
104	26	3
105	27	3

도면24d

테이블 24 - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
106	28	3
107	29	3
108	30	3
109	31	3
110	0	4
111	1	4
112	2	4
113	3	4
114	4	4
115	5	4
116	6	4
117	7	4
118	8	4
119	9	4
120	10	4
121	11	4
122	12	4
123	13	4
124	14	4
125	15	4
126	16	4
127	17	4
128	18	4
129	19	4
130	20	4
131	21	4
132	22	4
133	23	4
134	24	4
135	25	4
136	26	4
137	27	4
138	28	4
139	29	4
140	30	4
141	31	4

도면25a

테이블 25: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	4	0
1	12	0
2	0	1
3	8	1
4	4	1
5	12	1
6	0	2
7	8	2
8	4	2
9	12	2
10	0	3
11	8	3
12	4	3
13	12	3
14	0	4
15	8	4
16	4	4
17	12	4
18	1	0
19	2	0
20	3	0
21	5	0
22	6	0
23	7	0
24	9	0
25	10	0
26	11	0
27	13	0
28	14	0
29	15	0
30	1	1
31	2	1
32	3	1
33	5	1
34	6	1
35	7	1

도면25b

테이블 25 - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
36	9	1
37	10	1
38	11	1
39	13	1
40	14	1
41	15	1
42	16	1
43	17	1
44	18	1
45	19	1
46	20	1
47	21	1
48	22	1
49	23	1
50	24	1
51	25	1
52	26	1
53	27	1
54	28	1
55	29	1
56	30	1
57	31	1
58	1	2
59	2	2
60	3	2
61	5	2
62	6	2
63	7	2
64	9	2
65	10	2
66	11	2
67	13	2
68	14	2
69	15	2
70	16	2

도면25c

테이블 25 - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
71	17	2
72	18	2
73	19	2
74	20	2
75	21	2
76	22	2
77	23	2
78	24	2
79	25	2
80	26	2
81	27	2
82	28	2
83	29	2
84	30	2
85	31	2
86	1	3
87	2	3
88	3	3
89	5	3
90	6	3
91	7	3
92	9	3
93	10	3
94	11	3
95	13	3
96	14	3
97	15	3
98	16	3
99	17	3
100	18	3
101	19	3
102	20	3
103	21	3
104	22	3
105	23	3

도면25d

테이블 25 - 계속: 삼각형 및 기하학적 파티션을 포함하는 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
106	24	3
107	25	3
108	26	3
109	27	3
110	28	3
111	29	3
112	30	3
113	31	3
114	1	4
115	2	4
116	3	4
117	5	4
118	6	4
119	7	4
120	9	4
121	10	4
122	11	4
123	13	4
124	14	4
125	15	4
126	16	4
127	17	4
128	18	4
129	19	4
130	20	4
131	21	4
132	22	4
133	23	4
134	24	4
135	25	4
136	26	4
137	27	4
138	28	4
139	29	4
140	30	4
141	31	4

도면26

테이블 26. Dis[]에 대한 룩업 테이블

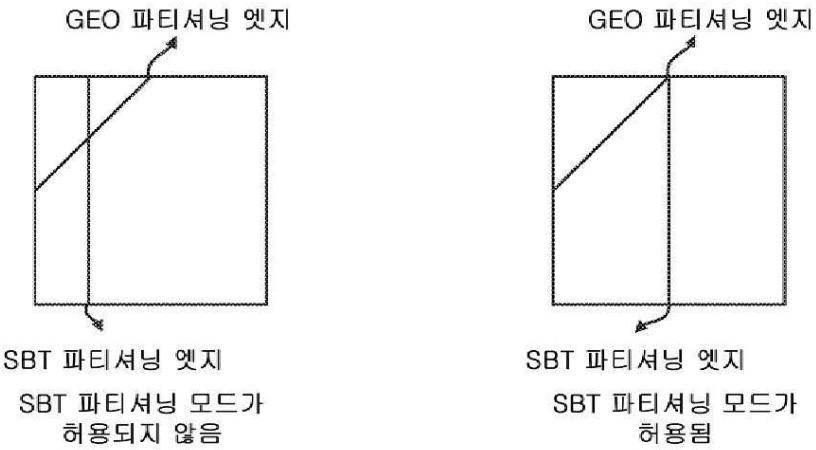
Idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Dis[idx]	128	126	118	106	90	72	48	24	0	-24	-48	-72	-90	-106	-118	-126
Idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Dis[idx]	-128	-126	-118	-106	-90	-72	-48	-24	0	24	48	72	90	106	118	126

도면27

테이블 27, 코딩 유닛 신택스 구조

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
...	
if(!pcm_flag[x0][y0]) {	
if(CuPredMode[x0][y0] != MODE_INTRA && merge_flag[x0][y0] == 0)	
cu_cbf	ae(v)
if(cu_cbf) {	
if(CuPredMode[x0][y0] == MODE_INTER && sps_sbt_enabled_flag && !ciip_flag[x0][y0] && !Merge_triangle_flag[x0][y0] && !geo_flag[x0][y0]) {	
if(cbWidth <= MaxSbtSize && cbHeight <= MaxSbtSize) {	
allowSbtVerH = cbWidth >= 8	
allowSbtVerQ = cbWidth >= 16	
allowSbtHorH = cbHeight >= 8	
allowSbtHorQ = cbHeight >= 16	
if(allowSbtVerH || allowSbtHorH || allowSbtVerQ || allowSbtHorQ)	
cu_sbt_flag	ae(v)
}	
...	
}	

도면28



도 28: SBT 및 GEO 파티셔닝의 예시

도면29

테이블 29. 코딩 유닛 신택스 구조

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
...	
if(!pcm_flag[x0][y0]) {	
if(CuPredMode[x0][y0] != MODE_INTRA && merge_flag[x0][y0] == 0)	
cu_cbf	ae(v)
if(cu_cbf) {	
if(CuPredMode[x0][y0] == MODE_INTER && sps_sbt_enabled_flag && !ciip_flag[x0][y0] && !Merge_triangle_flag[x0][y0]) {	
if(cbWidth <= MaxSbtSize && cbHeight <= MaxSbtSize) {	
allowSbtVerH = cbWidth >= 8	
allowSbtVerQ = cbWidth >= 16	
allowSbtHorH = cbHeight >= 8	
allowSbtHorQ = cbHeight >= 16	
if(allowSbtVerH || allowSbtHorH || allowSbtVerQ || allowSbtHorQ)	
cu_sbt_flag	ae(v)
}	
if(cu_sbt_flag) {	
if((allowSbtVerH || allowSbtHorH) && (allowSbtVerQ || allowSbtHorQ))	
cu_sbt_quad_flag	ae(v)
if(((cu_sbt_quad_flag && allowSbtVerQ && allowSbtHorQ) || (!cu_sbt_quad_flag && allowSbtVerH && allowSbtHorH)) && *geo_angle_idx != 0*)	
cu_sbt_horizontal_flag	ae(v)
cu_sbt_pos_flag	ae(v)
}	
}	
...	
}	

도면30a

테이블 30: 기하학적 파티셔닝 서브모드의 총 수가 80으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	1	0
4	1	1
5	1	2
6	1	3
7	2	0
8	2	1
9	2	2
10	2	3
11	3	0
12	3	1
13	3	2
14	3	3
15	4	1
16	4	2
17	4	3
18	5	0
19	5	1
20	5	2
21	5	3
22	6	1
23	6	2
24	6	3
25	7	0
26	7	1
27	7	2
28	7	3
29	8	1
30	8	2
31	8	3
32	9	0
33	9	1
34	9	2
35	9	3
36	10	0
37	10	1
38	10	2
39	10	3

도면30b

테이블 30 - 계속: 기하학적 파티셔닝 서브모드의 총 수가 80으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
40	11	0
41	11	1
42	11	2
43	11	3
44	12	1
45	12	2
46	12	3
47	13	1
48	13	2
49	13	3
50	14	1
51	14	2
52	14	3
53	15	1
54	15	2
55	15	3
56	16	1
57	16	2
58	16	3
59	17	1
60	17	2
61	17	3
62	18	1
63	18	2
64	18	3
65	19	1
66	19	2
67	19	3
68	20	1
69	20	2
70	20	3
71	21	1
72	21	2
73	21	3
74	22	1
75	22	2
76	22	3
77	23	1
78	23	2
79	23	3

도면31a

테이블 31: 기하학적 파티셔닝 서브모드의 총 수가 64로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	10	2
3	10	1
4	11	3
5	11	2
6	11	1
7	1	0
8	1	1
9	1	2
10	1	3
11	19	3
12	19	2
13	19	1
14	9	0
15	9	1
16	9	2
17	9	3
18	12	3
19	12	2
20	12	1
21	2	0
22	2	1
23	2	2
24	2	3
25	18	3
26	18	2
27	18	1
28	8	0
29	8	1
30	8	2
31	8	3
32	13	3
33	13	2
34	13	1
35	3	0

도면31b

테이블 31 - 계속: 기하학적 파티셔닝 서브모드의 총 수가 64로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
36	3	1
37	3	2
38	3	3
39	17	3
40	17	2
41	17	1
42	7	0
43	7	1
44	7	2
45	7	3
46	14	3
47	14	2
48	14	1
49	4	0
50	4	1
51	4	2
52	4	3
53	16	3
54	16	2
55	16	1
56	6	0
57	6	1
58	6	2
59	6	3
60	5	1
61	5	2
62	15	2
63	15	1

도면32

테이블 32: Rho[]에 대한 룩업 테이블

Idx	0	1	2	3	4	5	6	7
Rho[idx]	16384	13056	8960	4352	0	-4352	-8960	-13056
Idx	8	9	10	11	12	13	14	15
Rho[idx]	-16384	-19200	-21248	-22784	-23040	-22784	-21248	-19200

도면33

테이블 33: 블록 당 각 연산의 수

식	×	<<	+	비교
블록 당	4	9	7	6
$displacementX = angleIdx \% 16$	–	0	0	0
$displacementY = (displacementX + NumAngles \gg 2) \% NumAngles$	–	–	–	0
$offsetX = \begin{cases} ((256-W) \gg 1) + A < N\ ?\ (D \times W) \gg 3 : -((D \times W) \gg 3) & \text{은 수평 분할되거나 또는 (수직 분할되지 않음)} \\ \text{그렇지 않으면} \end{cases}$ $offsetY = \begin{cases} ((256-H) \gg 1) + A < N\ ?\ (D \times H) \gg 3 : -((D \times H) \gg 3) & \text{은 수평 분할되거나 또는 (수직 분할되지 않음)} \\ (256-H) \gg 1 & \text{그렇지 않으면} \end{cases}$	–	3	3	3
$offsetXmotion = \begin{cases} (64-(W \gg 2)) \gg 1 & \text{은 수평 분할되거나 또는 (수직 분할되지 않음)} \\ ((64-(W \gg 2)) \gg 1) + A < N\ ?\ (D \times W) \gg 5 : -((D \times W) \gg 5) & \text{그렇지 않으면} \end{cases}$ $offsetYmotion = \begin{cases} ((64-(H \gg 2)) \gg 1) + A < N\ ?\ (D \times H) \gg 5 : -((D \times H) \gg 5) & \text{은 수평 분할되거나 또는 (수직 분할되지 않음)} \\ (64-(H \gg 2)) \gg 1 & \text{그렇지 않으면} \end{cases}$	–	5	3	3
루마 샘플 당	2	3	8	2
$distFromLine = (((x + offsetX) \ll 1) + 1) \times Dis[displacementX] + (((y + offsetY) \ll 1) + 1) \times Dis[displacementY] - Rho[displacementX]$	2	2	6	0
$distScaled = Min((abs(distFromLine) + 4) \gg 3, 26)$	0	–	–	1
$sampleWeight_L[x][y] = distFromLine \le 0\ ?\ GeoFilter[distScaled] : 8 - GeoFilter[distScaled]$	0	0	1	1
4x4 서브블록 당	2	2	6	2
$distFromLine_{subblk} = (((x_{subblk} + offsetXmotion) \ll 3) + 1) \times Dis[displacementX] + (((y_{subblk} + offsetYmotion) \ll 3) + 1) \times Dis[displacementY] - rho_{subblk}[displacementX]$	2	2	6	0
$motionMask[x_{subblk}][y_{subblk}] = abs(distFromLine_{subblk}) < 256\ ?\ 2 : (distFromLine_{subblk} \le 0\ ?\ 0 : 1)$	0	0	0	2

도면34

테이블 34: $Rho_{subblk}[]$에 대한 룩업 테이블

Idx	0	1	2	3	4	5	6	7
$Rho_{subblk}[idx]$	16192	12903	8855	4301	0	-4301	-8855	-12903
Idx	8	9	10	11	12	13	14	15
$Rho_{subblk}[idx]$	-16192	-18975	-20999	-22517	-22770	-22517	-20999	-18975

도면35a

4	5	6	7	8	8	8	8
3	4	5	6	7	8	8	8
2	3	4	5	6	7	8	8
1	2	3	4	5	6	7	8
0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6
0	0	0	1	2	3	4	5
0	0	0	0	1	2	3	4

도면35b

8	8	8	8	7	6	5	4
8	8	8	7	6	5	4	3
8	8	7	6	5	4	3	2
8	7	6	5	4	3	2	1
7	6	5	4	3	2	1	0
6	5	4	3	2	1	0	0
5	4	3	2	1	0	0	0
4	3	2	1	0	0	0	0

도면36a

4	5	6	7	8	8	8	8
3	4	5	6	7	8	8	8
2	3	4	5	6	7	8	8
1	2	3	4	5	6	7	8
0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6
0	0	0	1	2	3	4	5
0	0	0	0	1	2	3	4

도면36b

0	0	0	0	1	2	3	4
0	0	0	1	2	3	4	5
0	0	1	2	3	4	5	6
0	1	2	3	4	5	6	7
1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	8
3	4	5	6	7	8	8	8
4	5	6	7	8	8	8	8

도면37

테이블 37: 상이한 블록 형태에 대한 삼각형 파티션 모드의 각도

폭 : 높이 비율	1 : 1	1 : 2	1 : 4	1 : 8	1 : 16
삼각형 각도 1	135°	116.5° ≈ 112.5°	104° ≈ 101.25°	97.1° ≈ 101.25°	93.6° ≈ 90°
삼각형 각도 2	45°	63.5° ≈ 67.5°	76° ≈ 78.75°	82.9° ≈ 78.75°	86.4° ≈ 90°
폭 : 높이 비율		2 : 1	4 : 1	8 : 1	16 : 1
삼각형 각도 1		153.4° ≈ 157.5°	166° ≈ 168.75°	172.9° ≈ 168.75°	176.4° ≈ 0°
삼각형 각도 2		26.6° ≈ 22.5°	14° ≈ 11.25°	7.1° ≈ 11.25°	3.6° ≈ 0°

도면38a

테이블 38: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	3
3	0	4
4	1	0
5	1	1
6	1	2
7	1	3
8	1	4
9	2	0
10	2	1
11	2	2
12	2	3
13	2	4
14	3	0
15	3	1
16	3	2
17	3	3
18	3	4
19	*4*	*0*
~~19~~20	*4*	*1*
~~20~~21	*4*	*2*
~~21~~22	*4*	*3*
~~22~~23	*4*	*4*
~~23~~24	*5*	*0*
~~24~~25	*5*	*1*
~~25~~26	*5*	*2*
~~26~~27	*5*	*3*
~~27~~28	*5*	*4*
~~28~~	~~6~~	~~0~~
29	6	1
30	6	2
31	6	3
32	6	4
33	7	0
34	7	1
35	7	2

도면38b

테이블 38 - 계속: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
35	7	2
36	7	3
37	7	4
38	8	1
39	8	2
40	8	3
41	8	4
42	9	0
43	9	1
44	9	2
45	9	3
46	9	4
~~47~~	~~10~~	~~0~~
~~48~~47	10	1
~~49~~48	10	2
~~50~~49	10	3
~~51~~50	10	4
~~52~~51	11	0
~~53~~52	11	1
~~54~~53	11	2
~~55~~54	11	3
~~56~~55	11	4
57	12	1
58	12	2
59	12	3
60	12	4
61	13	0
62	13	1
63	13	2
64	13	3
65	13	4
66	14	0
67	14	1
68	14	2
69	14	3
70	14	4

도면38c

테이블 38 - 계속: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

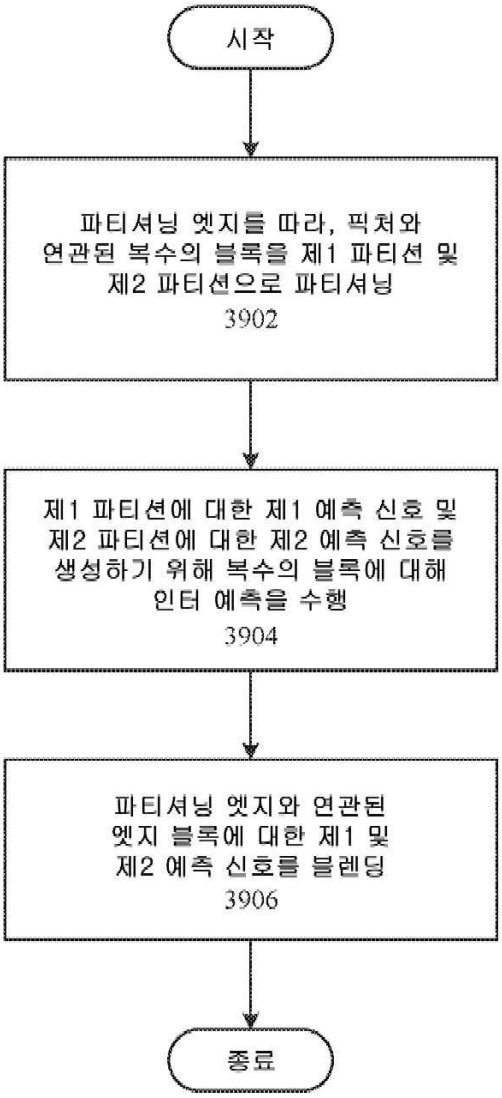
geo_partition_idx	angleIdx	distanceIdx
71	15	0
72	15	1
73	15	2
74	15	3
75	15	4
76	16	1
77	16	2
78	16	3
79	16	4
80	17	1
81	17	2
82	17	3
83	17	4
84	18	1
85	18	2
86	18	3
87	18	4
88	19	1
89	19	2
90	19	3
91	19	4
92	20	1
93	20	2
94	20	3
95	20	4
96	21	1
97	21	2
98	21	3
99	21	4
100	22	1
101	22	2
102	22	3
103	22	4
104	23	1
105	23	2

도면38d

테이블 38 - 계속: 기하학적 파티셔닝 서브모드의 총 수가 140으로 설정될 때 angleIdx 및 distanceIdx에 대한 룩업 테이블

geo_partition_idx	angleIdx	distanceIdx
106	23	3
107	23	4
108	24	1
109	24	2
110	24	3
111	24	4
112	25	1
113	25	2
114	25	3
115	25	4
116	26	1
117	26	2
118	26	3
119	26	4
120	27	1
121	27	2
122	27	3
123	27	4
124	28	1
125	28	2
126	28	3
127	28	4
128	29	1
129	29	2
130	29	3
131	29	4
132	30	1
133	30	2
134	30	3
135	30	4
136	31	1
137	31	2
138	31	3
139	31	4

도면39


시작
파티셔닝 엣지를 따라, 픽처와 연관된 복수의 블록을 제1 파티션 및 제2 파티션으로 파티셔닝
3902
제1 파티션에 대한 제1 예측 신호 및 제2 파티션에 대한 제2 예측 신호를 생성하기 위해 복수의 블록에 대해 인터 예측을 수행
3904
파티셔닝 엣지와 연관된 엣지 블록에 대한 제1 및 제2 예측 신호를 블렌딩
3906
종료

3900