



공개특허 10-2023-0044015



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0044015
(43) 공개일자 2023년03월31일

(51) 국제특허분류(Int. Cl.)

H04N 19/105 (2014.01) *H04N 19/11* (2014.01)
H04N 19/157 (2014.01) *H04N 19/423* (2014.01)
H04N 19/593 (2014.01)

(52) CPC특허분류

H04N 19/105 (2015.01)
H04N 19/11 (2015.01)

(21) 출원번호 10-2023-7007642

(22) 출원일자(국제) 2022년08월24일

심사청구일자 2023년03월03일

(85) 번역문제출일자 2023년03월03일

(86) 국제출원번호 PCT/US2022/075402

(87) 국제공개번호 WO 2023/028522

국제공개일자 2023년03월02일

(30) 우선권주장

63/237,098 2021년08월25일 미국(US)

17/893,877 2022년08월23일 미국(US)

(71) 출원인

텐센트 아메리카 엘엘씨

미국 94306 캘리포니아주 팔로 알토 파크 블러바드 2747

(72) 별명자

자오 신

미국 94306 캘리포니아주 팔로 알토 2747 파크 불러바드 텐센트 아메리카 엘엘씨 내

수 샤오종

미국 94306 캘리포니아주 팔로 알토 2747 파크 불러바드 텐센트 아메리카 엘엘씨 내

리우 산

미국 94306 캘리포니아주 팔로 알토 2747 파크 불러바드 텐센트 아메리카 엘엘씨 내

(74) 대리인

유미특허법인

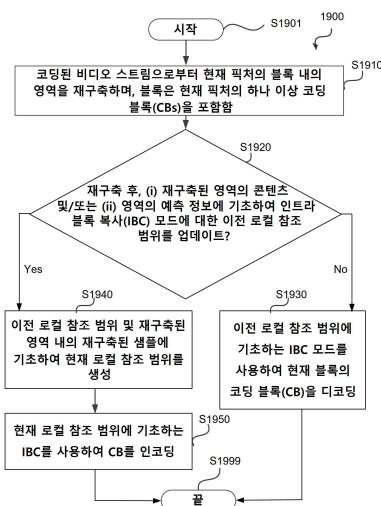
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 적응적 로컬 참조 범위가 있는 인트라 블록 복사

(57) 요 약

본 개시의 측면들은 비디오 디코딩을 위한 방법, 장치 및 컴퓨터로 판독 가능한 비일시적인 저장 매체를 제공한다. 상기 장치는 코딩된 비디오 비트스트림으로부터 현재 픽처의 블록 내의 영역을 재구축하도록 구성된 처리 회로를 포함한다. 상기 블록은 상기 현재 픽처의 하나 이상의 코딩 블록(CBs)을 포함한다. 상기 재구축 후, 상기 처리 회로는 재구축된 영역의 콘텐츠 및/또는 상기 재구축된 영역의 예측 정보에 기초하여 인트라 블록 복사(IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정한다. 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 한 결정에 응답하여, 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않는다. 상기 처리 회로는 상기 재구축된 영역을 포함하지 않는 상기 이전 로컬 참조 범위에 기초하는 상기 IBC 모드를 사용하여 상기 현재 픽처의 코딩 블록(CB)을 디코딩하도록 구성된다.

대 표 도 - 도19



(52) CPC특허분류

H04N 19/157 (2015.01)

H04N 19/423 (2015.01)

H04N 19/593 (2015.01)

명세서

청구범위

청구항 1

비디오 디코더에서의 비디오 디코딩을 위한 방법으로서,

코딩된 비디오 비트스트림으로부터 현재 픽처의 블록 내의 영역을 재구축하는 단계 - 상기 블록은 상기 현재 픽처의 하나 이상의 코딩 블록(coding blocks, CBs)을 포함함 -;

상기 재구축 후, (i) 재구축된 영역의 콘텐츠 및/또는 (ii) 상기 재구축된 영역의 예측 정보에 기초하여 인트라 블록 복사(intra block copy, IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계;

상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 한 결정에 응답하여,

상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않고;

상기 재구축된 영역을 포함하지 않는 상기 이전 로컬 참조 범위에 기초하는 상기 IBC 모드를 사용하여 상기 현재 픽처의 코딩 블록(coding block, CB)을 디코딩하는 단계

를 포함하는 방법.

청구항 2

제1항에 있어서,

상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계는,

상기 재구축된 영역의 재구축된 샘플에 기초하여 상기 재구축된 영역의 콘텐츠를 결정하는 단계 - 상기 재구축된 영역의 콘텐츠는 상기 재구축된 영역의 텍스처 콘텐츠의 백분율 및/또는 스크린 콘텐츠의 백분율을 지시함 -; 및

상기 재구축된 영역의 상기 텍스처 콘텐츠의 백분율 및/또는 상기 스크린 콘텐츠의 백분율에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계를 포함하는, 방법.

청구항 3

제2항에 있어서,

상기 재구축된 영역의 콘텐츠는 스크린 콘텐츠를 포함하지 않고;

상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계는 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트하지 않기로 결정하는 단계를 포함하는, 방법.

청구항 4

제1항에 있어서,

상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계는,

상기 영역의 예측 모드를 지시하는, 상기 영역의 예측 정보를 디코딩하는 단계; 및

상기 예측 모드가 미리 정의된 예측 모드 세트에 있는지에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계를 포함하는, 방법.

청구항 5

제1항에 있어서,

상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계는,

상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지를 지시하는, 상기 영역의 예

측 정보를 디코딩하는 단계; 및

상기 예측 정보 내의, 상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지의 지시에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계를 포함하는, 방법.

청구항 6

제5항에 있어서,

상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지는 상기 영역의 예측 정보에 포함된 플래그에 의해 지시되며;

상기 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계는 상기 플래그에 기초하여 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정하는 단계를 포함하는, 방법.

청구항 7

제6항에 있어서,

상기 플래그는 상기 블록의 임의의 재구축된 샘플이 상기 IBC 모드에서 상기 로컬 참조 범위를 업데이트하는 데 사용되는지를 지시하는, 방법.

청구항 8

제1항에 있어서,

상기 IBC 모드에 대한 이전 로컬 참조 범위는 업데이트되도록 결정되고;

상기 이전 로컬 참조 범위는 참조 샘플 메모리(reference sample memory, RSM)에 저장된 복수의 영역을 포함하며, 상기 복수의 영역은 상기 참조 샘플을 포함하고;

상기 방법은,

상기 복수의 영역의 콘텐츠에 기초하여 상기 복수의 영역에 순위를 매기는 단계 - 각각의 상기 콘텐츠는 상기 복수의 영역 각각에 대응함 -;

상기 복수의 영역의 순위에 기초하여 상기 재구축된 영역의 재구축된 샘플로 대체될 상기 복수의 영역 중의 영역을 선택하는 단계; 및

선택된 영역을 상기 이전 로컬 참조 범위에서 상기 재구축된 영역의 상기 재구축된 샘플로 대체하여 현재 로컬 참조 범위를 생성하는 단계를 더 포함하는 방법.

청구항 9

제8항에 있어서,

상기 복수의 영역 각각에 대응하는 각각의 상기 콘텐츠는 대응하는 영역에서의 스크린 콘텐츠의 백분율 또는 텍스처 콘텐츠의 백분율로 지시되고;

상기 복수의 영역의 순위를 매기는 단계는 상기 복수의 영역의 상기 스크린 콘텐츠의 백분율 각각 또는 상기 텍스처 콘텐츠의 백분율 각각에 기초하여 상기 복수의 영역의 순위를 매기는 단계를 포함하는, 방법.

청구항 10

제1항에 있어서,

상기 IBC 모드에 대한 이전 로컬 참조 범위가 업데이트되도록 결정되고;

상기 이전 로컬 참조 범위는 참조 샘플 메모리(RSM)에 저장된 복수의 영역을 포함하며;

상기 복수의 영역의 각각의 영역은 미리 정의된 예측 모드 세트 중의 하나 이상의 예측 모드로 코딩되는 각각의 영역에 많은 특별히 코딩된 참조 샘플을 포함하고;

상기 방법은,

상기 복수의 영역에서의 상기 특별히 코딩된 참조 샘플 각각의 수에 기초하여 상기 복수의 영역에 순위를 매기는 단계;

상기 복수의 영역의 순위에 기초하여 상기 재구축된 영역의 재구축된 샘플로 대체될 상기 복수의 영역 중의 영역을 선택하는 단계; 및

선택된 영역을 상기 이전 로컬 참조 범위에서 상기 재구축된 영역의 재구축된 샘플로 대체하여 현재 로컬 참조 범위를 생성하는 단계를 더 포함하는 방법.

청구항 11

제1항에 있어서,

상기 IBC 모드에 대한 이전 로컬 참조 범위가 업데이트되도록 결정되고;

상기 이전 로컬 참조 범위는 참조 샘플 메모리(RSM)에 저장된 복수의 영역을 포함하며;

상기 복수의 영역의 각각의 영역과 연관된 참조 빈도는 상기 각각의 영역이 상기 현재 꽂처의 CBs를 예측하는데 사용되는 블록 벡터에 의해 참조되는 횟수를 지시하고;

상기 방법은,

상기 복수의 영역 각각의 참조 빈도에 기초하여 상기 복수의 영역에 순위를 매기는 단계;

상기 복수의 영역의 순위에 기초하여 상기 재구축된 영역의 재구축된 샘플로 대체될 상기 복수의 영역 중의 영역을 선택하는 단계; 및

선택된 영역을 상기 이전 로컬 참조 범위에서 상기 재구축된 영역의 재구축된 샘플로 대체하여 현재 로컬 참조 범위를 생성하는 단계를 더 포함하는 방법.

청구항 12

제1항에 있어서,

상기 이전 로컬 참조 범위는 복수의 영역을 포함하고;

상기 복수의 영역 중 하나 내의 서브영역의 재구축된 샘플을 제외한 상기 복수의 영역의 재구축된 샘플을 참조 샘플 메모리(RSM)에 저장하고;

상기 복수의 영역 중 하나 내의 서브영역의 재구축된 샘플은 상기 RSM 내의 상기 복수 영역 중 하나 내의 서브 영역에 대응하는 패딩된 샘플(padded sample)을 저장하는 패딩 프로세스에 의해 대체되는, 방법.

청구항 13

제1항에 있어서,

상기 재구축된 영역이 재구축된 코딩 블록이거나,

상기 재구축된 영역의 크기는 메모리 업데이트 영역 유닛의 크기인, 방법.

청구항 14

비디오 디코딩을 위한 장치로서,

처리 회로를 포함하고,

상기 처리 회로는,

코딩된 비디오 비트스트림으로부터 현재 꽂처의 블록 내의 영역을 재구축하고 - 상기 블록은 상기 현재 꽂처의 하나 이상의 코딩 블록(CBs)을 포함함 -;

상기 재구축 후, (i) 재구축된 영역의 콘텐츠 및/또는 (ii) 상기 재구축된 영역의 예측 정보에 기초하여 인트라 블록 복사(IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정하고;

상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 한 결정에 응답하여,

상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않고;

상기 재구축된 영역을 포함하지 않는 상기 이전 로컬 참조 범위에 기초하는 상기 IBC 모드를 사용하여 상기 현재 꽉쳐의 코딩 블록(CB)을 디코딩하도록 구성되는,

장치.

청구항 15

제14항에 있어서,

상기 처리 회로는,

상기 재구축된 영역의 재구축된 샘플에 기초하여 상기 재구축된 영역의 콘텐츠를 결정하고 - 상기 재구축된 영역의 콘텐츠는 상기 재구축된 영역의 텍스처 콘텐츠의 백분율 및/또는 스크린 콘텐츠의 백분율을 지시함 -;

상기 재구축된 영역의 상기 텍스처 콘텐츠의 백분율 및/또는 상기 스크린 콘텐츠의 백분율에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정하도록 구성되는, 장치.

청구항 16

제15항에 있어서,

상기 재구축된 영역의 콘텐츠는 스크린 콘텐츠를 포함하지 않고;

상기 처리 회로는 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트하지 않기로 결정하도록 구성되는, 장치.

청구항 17

제14항에 있어서,

상기 처리 회로는,

상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지를 지시하는, 상기 영역의 예측 정보를 디코딩하고;

상기 예측 정보에서, 상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지의 지시에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정하도록 구성되는, 장치.

청구항 18

제17항에 있어서,

상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지는 상기 영역의 예측 정보에 포함된 플래그에 의해 지시되며;

상기 처리 회로는 상기 플래그에 기초하여 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정하도록 구성되는, 장치.

청구항 19

제18항에 있어서,

상기 플래그는 상기 블록의 임의의 재구축된 샘플이 상기 IBC 모드에서 상기 로컬 참조 범위를 업데이트하는 데 사용되는지를 지시하는, 장치.

청구항 20

프로그램을 저장하는, 컴퓨터로 판독 가능한 비일시적 저장 매체로서,

상기 프로그램은,

코딩된 비디오 비트스트림으로부터 현재 픽처의 블록 내의 영역을 재구축하는 것 - 상기 블록은 상기 현재 픽처의 하나 이상의 코딩 블록(CBs)을 포함함 -;

상기 재구축 후, (i) 재구축된 영역의 콘텐츠 및/또는 (ii) 상기 재구축된 영역의 예측 정보에 기초하여 인트라 블록 복사(IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정하는 것;

상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 한 결정에 응답하여,

상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않고;

상기 재구축된 영역을 포함하지 않는 상기 이전 로컬 참조 범위에 기초하는 상기 IBC 모드를 사용하여 상기 현재 픽처의 코딩 블록(CB)을 디코딩하는 것을 수행하도록 적어도 하나의 프로세서에 의해 실행 가능한,

컴퓨터로 판독 가능한 비일시적 저장 매체.

발명의 설명

기술 분야

[0001] 참조에 의한 통합

[0002] 본 출원은 2021년 8월 25일에 출원된 미국 가출원 제63/237,098호, "IntraBC with Adaptive Local Reference Range(적응적 로컬 참조 범위가 있는 인트라BC)"에 대한 우선권의 이익을 주장하는, 2022년 8월 23일에 출원된 미국 특허출원 제17/893,877호, "INTRA BLOCK COPY WITH AN ADAPTIVE LOCAL REFERENCE RANGE(적응적 로컬 참조 범위가 있는 인트라 블록 복사)"에 대한 우선권의 이익을 주장한다. 선행 출원의 개시 내용은 그 전체가 참조에 의해 본 출원에 포함된다.

[0003] 본 개시는 일반적으로 비디오 코딩에 관련된 실시예를 설명한다.

배경 기술

[0004] 본 명세서에 제공되는 배경 설명은 본 개시의 맥락을 일반적으로 제시하기 위한 것이다. 저작물(work)이 이 배경기술 섹션에 설명되는 범위에서, 현재 지명된 발명자들의 저작물은 물론 출원 시에 선행 기술로 인정되지 않을 수 있는 설명의 측면들은 명시적으로도 묵시적으로도 본 개시의 선행 기술로 인정되지 않는다.

[0005] 이미지 및/또는 비디오 코딩 및 디코딩은 움직임 보상과 함께 인터 픽처 예측을 사용하여 수행될 수 있다. 압축되지 않은 디지털 이미지 및/또는 비디오는 일련의 픽처를 포함할 수 있으며, 각각의 픽처는, 예를 들어, 1920 × 1080 휘도 샘플(luminance sample, 루마 샘플이라고도 함) 및 연관된 색차 샘플(chrominance sample, 크로마 샘플이라고도 함)의 공간 차원을 갖는다. 일련의 픽처는 예를 들어 초당 60장의 픽처 또는 60Hz의 고정 또는 가변 픽처 레이트(비공식적으로 프레임 레이트라고 알려져 있음)를 가질 수 있다. 압축되지 않은 이미지 및/또는 비디오는 특정 비트레이트 요건을 갖는다. 예를 들어, 샘플당 8비트의 1080p60 4:2:0 비디오(60Hz 프레임 레이트에서 1920×1080 휘도 샘플 해상도)는 1.5Gbit/s에 가까운 대역폭을 필요로 한다. 한 시간의 이러한 비디오는 600GB 이상의 저장 공간을 필요로 한다.

[0006] 이미지 및/또는 비디오 코딩 및 디코딩의 한 가지 목적은 압축을 통해, 입력 비디오 신호의 중복성(redundancy)을 줄이는 것일 수 있다. 압축은 앞서 언급한 대역폭 및/또는 저장 공간 요건을, 경우에 따라서는 두자릿수 이상 감소시키는 데 도움이 될 수 있다. 본 명세서의 설명은 예시적인 예로서 비디오 인코딩/디코딩을 사용하지만, 동일한 기술이 본 개시의 사상에서 벗어나지 않는 유사한 방식으로 이미지 인코딩/디코딩에 적용될 수 있다. 무손실 압축(lossless compression)과 손실 압축(lossy compress) 둘 다는 물론, 이들의 조합도 사용될 수 있다. 무손실 압축이란 압축된 원본 신호(original signal)에서 원본 신호의 정확한 사본(exact copy)을 재구축할 수 있는 기술을 말한다. 손실 압축을 사용하는 경우, 재구축된 신호는 원본 신호와 동일하지 않을 수 있지만 원본 신호와 재구축된 신호 사이의 왜곡은, 재구축된 신호가 의도된 애플리케이션에 유용할 정도로 충분히 작다. 비디오의 경우, 손실 압축이 널리 채용된다. 용인 가능한 왜곡의 양은 애플리케이션에 따라 다른데, 예를 들어, 특정 소비자 스트리밍 애플리케이션의 사용자는 영화 또는 텔레비전 배급 애플리케이션(television distribution application)의 사용자보다 더 높은 왜곡을 용인할 수 있다. 달성 가능한 압축 비율은 다음을 반영할 수 있다: 허용 가능한/용인 가능한 왜곡이 높을수록 압축 비율이 높을 수 있다.

[0007] 비디오 인코더 및 디코더는, 예를 들어, 움직임 보상(motion compensation), 푸리에 변환 처리, 양자화

(quantization), 및 엔트로피 코딩(entropy coding)을 포함한, 여러 광범위한 범주로부터의 기술을 이용할 수 있다.

[0008] 비디오 코덱 기술은 인트라 코딩(intra coding)으로 알려진 기술을 포함할 수 있다. 인트라 코딩에서, 샘플 값은 이전에 재구축된 참조 픽처의 샘플 또는 다른 데이터에 대한 참조 없이 표현된다. 일부 비디오 코덱에서, 픽처는 샘플 블록들로 공간적으로 세분화된다. 샘플의 모든 블록이 인트라 모드로 코딩되는 경우, 그 픽처는 인트라 픽처(intra picture)일 수 있다. 인트라 픽처 및 독립 디코더 리프레시 픽처(independent decoder refresh picture)와 같은 그 도출물(derivation)은 디코더 상태를 재설정하는 데 사용될 수 있으며, 따라서 코딩된 비디오 비트스트림 및 비디오 세션의 첫 번째 픽처, 또는 스틸 이미지(still image)로 사용될 수 있다. 인트라 블록의 샘플은 변환에 도출될 수 있고, 변환 계수는 엔트로피 코딩 전에 양자화될 수 있다. 인트라 예측(intra prediction)은 변환 전 도메인(pre-transform domain)에서 샘플 값을 최소화하는 기술일 수 있다. 경우에 따라서는, 변환 후 DC 값이 작을수록, 그리고 AC 계수가 작을수록, 엔트로피 코딩 후 블록을 표현하기 위해 주어진 양자화 스텝 크기(quantization step size)에서 필요한 비트는 더 적다.

[0009] 예를 들어, MPEG-2 생성 코딩 기술에 사용되는 전통적인 인트라 코딩은 인트라 예측을 사용하지 않는다. 하지만, 일부 더 새로운 비디오 압축 기술은, 예를 들어 데이터 블록의 인코딩/디코딩 동안 획득되는 메타데이터 및/또는 주변 샘플 데이터에 기초하여 예측을 수행하도록 시도하는 기술을 포함한다. 이러한 기술은 앞으로 "인트라 예측" 기술이라고 한다. 적어도 일부 경우에, 인트라 예측은 참조 픽처로부터가 아닌 재구축중인 현재 픽처로부터의 참조 데이터만 사용한다.

[0010] 다양한 형태의 인트라 예측이 있을 수 있다. 이러한 기술 중 하나 이상이 주어진 비디오 코딩 기술에서 사용 가능한 경우, 사용 중인 기술은 인트라 예측 모드로 코딩될 수 있다. 어떤 경우에는, 인트라 예측 모드는 서브모드 및/또는 파라미터를 가질 수 있으며, 여기서 서브모드 및/또는 파라미터는 개별적으로 코딩될 수 있거나 집합적으로 모드 코드워드에 포함될 수 있으며, 이는 사용되는 예측 모드를 정의한다. 주어진 모드, 서브모드, 및/또는 파라미터 조합에 어떤 코드워드가 사용되는지는 인트라 예측을 통한 코딩 효율 이득(coding efficiency gain)에 영향을 미칠 수 있어서, 코드워드를 비트스트림으로 변환하는 데 사용되는 엔트로피 코딩 기술에도 영향을 미칠 수 있다.

[0011] 특정 모드의 인트라 예측은 H.264와 함께 도입되었고, H.265에서 개선되었으며, 공동 탐사 모델(joint exploration model, JEM), 다용도 비디오 코딩(versatile video coding, VVC) 및 벤치 마크 세트(benchmark set, BMS)와 같은 더 새로운 코딩 기술에서 더욱 개선되었다. 예측자 블록(predictor block)은 이미 사용 가능한 샘플의 이웃 샘플 값을 사용하여 형성될 수 있다. 이웃 샘플의 샘플 값은 방향에 따라 예측자 블록에 복사될 수 있다. 사용 중의 방향에 대한 참조는 비트스트림에 코딩될 수 있거나 그 자체가 예측될 수 있다.

[0012] 도 1a를 참조하면, H.265에 정의된 33개의 가능한 예측자 방향(35개 인트라 모드의 33개 각도 모드에 대응) 중 알려져 있는 9개의 예측자 방향의 서브세트가 우측 하단에 도시되어 있다. 화살표가 수렴하는 지점(101)은 예측되는 샘플을 나타낸다. 화살표는 샘플이 예측되는 방향을 나타낸다. 예를 들어, 화살표(102)는 샘플(101)이 하나의 샘플 또는 샘플들로부터 수평 방향에서 45도 각도로 우측 상단으로 예측된다는 것을 지시한다. 마찬가지로, 화살표(103)는 샘플(101)이 샘플 또는 샘플들로부터 수평 방향에서 22.5도 각도로 샘플(101)의 좌측 하단으로 예측된다는 것을 지시한다.

[0013] 도 1a를 여전히 참조하면, 좌측 상단에는 (굵은 파선으로 나타낸) 4×4 샘플의 정사각형 블록(104)이 표시되어 있다. 정사각형 블록(104)은 16개의 샘플을 포함하고, 각각은 "S", Y 차원의 위치(예: 행 색인), 및 X 차원의 위치(예: 열 색인)로 레이블이 부여된다. 예를 들어, 샘플 S21은 Y 차원의 (상단에서) 두 번째 샘플이고 X 차원의 (좌측에서) 첫 번째 샘플이다. 마찬가지로, 샘플 S44는 Y 및 X 차원 모두에서 블록(104)의 네 번째 샘플이다. 블록의 크기가 4×4 샘플이므로, S44는 우측 하단에 있다. 유사한 번호 매기기 방식을 따르는 참조 샘플이 추가로 도시되어 있다. 참조 샘플은 R, 블록(104)에 상대적인 Y 위치(예: 행 색인) 및 X 위치(열 색인)로 레이블이 부여된다. H.264 및 H.265 모두에서, 예측 샘플은 재구축중인 블록에 이웃하며, 따라서 음의 값을 사용할 필요는 없다.

[0014] 인트라 픽처 예측(intra picture prediction)은 시그널링된 예측 방향으로 지시되는 이웃 샘플로부터 참조 샘플 값을 복사함으로써 작동할 수 있다. 예를 들어, 코딩된 비디오 비트스트림이, 이 블록에 대해, 화살표(102)와 일치하는 예측 방향을 지시하는 시그널링을 포함한다고 가정한다 - 즉, 샘플들은 수평 방향에서 45도 각도로 우측 상단에 대한 샘플들로부터 예측된다. 이러한 경우, 샘플 S41, S32, S23 및 S14는 동일한 참조 샘플 R05로부터 예측된다. 그런 다음 샘플 S44는 참조 샘플 R08로부터 예측된다.

- [0015] 어떤 경우에는, 참조 샘플을 계산하기 위해, 특히 방향을 45도로 균등하게 나눌 수 없을 때, 예를 들어 보간을 통해 다수의 참조 샘플의 값이 결합될 수 있다.
- [0016] 가능한 방향의 수는 비디오 코딩 기술이 발전됨에 따라 증가하였다. H.264(2003년)에서는, 9개의 서로 다른 방향이 표현될 수 있다. H.265(2013년)에서는 33개로 증가하였다. 현재, JEM/VVC/BMS는 최대 65개의 방향을 지원할 수 있다. 가장 가능성이 있는 방향을 식별하기 위한 실험이 수행되었으며, 엔트로피 코딩에서의 특정 기술이 그러한 가장 가능성이 있는 방향을 적은 수의 비트로 인코딩하는 데 사용되어, 가능성이 적은 방향에 대해 일정한 패널티를 받아들인다. 또한, 방향 자체는 때로 이미 디코딩된 이웃 블록에 사용된 이웃 방향으로부터 예측될 수 있다.
- [0017] 도 1b는 시간 경과에 따라 증가하는 예측 방향의 수를 설명하기 위해 JEM에 따른 65개의 인트라 예측 방향을 나타낸 개략도(110)를 도시한다.
- [0018] 코딩된 비디오 비트스트림에서 방향을 나타내는 인트라 예측 방향 비트의 매핑은 비디오 코딩 기술에 따라 다를 수 있다. 이러한 매핑은 예를 들어 간단한 직접 매핑에서부터 코드워드, 가장 가능성이 높은 모드를 포함하는 복잡한 적응적 방식, 및 유사한 기술에 이르기까지 다양하다. 하지만, 대부분의 경우에, 비디오 콘텐츠에서 다른 특정 방향보다 통계적으로 발생할 가능성이 낮은 예측에 대한 특정 방향이 있을 수 있다. 비디오 압축의 목표는 중복성을 줄이는 것이므로, 잘 설계된 비디오 코딩 기술에서, 가능성이 낮은 방향은 가능성이 더 높은 방향보다 더 많은 수의 비트로 표현될 수 있다.
- [0019] 움직임 보상은 손실 압축 기술일 수 있으며 이전에 재구축된 픽처 또는 그 일부(참조 픽처)로부터의 샘플 데이터가 움직임 벡터(motion vector, 이하 MV)에 의해 지시되는 방향으로 공간적으로 시프트된 후, 새로 재구축되는 픽처 또는 픽처 일부의 예측에 사용되는 기술에 관련될 수 있다. 경우에 따라서는, 참조 픽처는 현재 재구축 중인 픽처와 동일할 수 있다. MV는 X와 Y의 2차원, 또는 3차원일 수 있으며, 세 번째 차원은 사용중인 참조 픽처의 지시이다(후자는 간접적으로 시간 차원일 수 있음).
- [0020] 일부 비디오 압축 기술에서, 샘플 데이터의 특정 영역(area)에 적용할 수 있는 MV는 다른 MV, 예를 들어 재구축 중인 영역에 공간적으로 인접하고 디코딩 순서에서 MV보다 선행하는 샘플 데이터의 다른 영역과 관련되어 있는 다른 MV들로부터 예측될 수 있다. 이렇게 하면 MV를 코딩하는 데 필요한 데이터의 양을 상당히 줄일 수 있으므로, 중복성을 없애고 압축을 높일 수 있다. 예를 들어, 카메라로부터 도출된 입력 비디오 신호(자연 비디오로 알려져 있음)를 코딩할 때 단일 MV가 적용될 수 있는 영역보다 큰 영역은 비디오 시퀀스에서 유사한 방향으로 이동하는 통계적 가능성이 있기 때문에, MV 예측은 효과적으로 작동할 수 있고, 따라서 경우에 따라서는 이웃 영역의 MV로부터 도출된 유사한 움직임 벡터를 사용하여 예측될 수 있다. 그 결과 주어진 영역에 대해 발견된 MV는 주변 MV로부터 예측된 MV와 유사하거나 동일하며, 엔트로피 코딩 후, MV를 직접 코딩한다면 사용되었을 비트 수보다 더 적은 수의 비트로 표시될 수 있다. 경우에 따라서는, MV 예측은 원래 신호(즉, 샘플 스트림)로부터 도출된 신호(즉, MV)의 무손실 압축의 일례일 수 있다. 다른 경우에는 MV 예측 자체가 손실될 수 있는데, 예를 들어 주변의 여러 MV로부터 예측자를 계산할 때 라운딩 오차(rounding error)가 발생하기 때문이다.
- [0021] 다양한 MV 예측 메커니즘이 H.265/HEVC(ITU-T Rec. H.265, "High Efficiency Video Coding(고효율 비디오 코딩)", 2016년 12월)에 기술되어 있다. H.265가 제공하는 많은 MV 예측 메커니즘 중에서 도 2를 참조하여 설명된 기술은 이하 "공간 병합(spatial merge)"이라고 지칭한다.
- [0022] 도 2를 참조하면, 현재 블록(201)은 공간적으로 시프트된 동일한 크기의 이전 블록으로부터 예측 가능하도록 움직임 검색 프로세스 동안 인코더에 의해 발견된 샘플을 포함한다. MV를 직접 코딩하는 대신, MV는 A0, A1 및 B0, B1, B2(각각 202~206)으로 표시된 5개의 주변 샘플 중 하나와 연관된 MV를 사용하여, 하나 이상의 참조 픽처와 연관된 메타데이터로부터, 예를 들어 가장 최근의(디코딩 순서상) 참조 픽처로부터 도출될 수 있다. H.265에서, MV 예측은 이웃 블록이 사용하는 것과 동일한 참조 픽처로부터의 예측자를 사용할 수 있다.

발명의 내용

- [0023] 본 개시의 측면들은 비디오 인코딩 및 디코딩을 위한 방법 및 장치를 제공한다. 일부 예에서, 비디오 디코딩을 위한 장치는 처리 회로를 포함한다. 상기 처리 회로는 코딩된 비디오 비트스트림으로부터 현재 픽처의 블록 내의 영역을 재구축하도록 구성된다. 상기 블록은 상기 현재 픽처의 하나 이상의 코딩 블록(coding blocks, CBs)을 포함한다. 상기 재구축 후, 상기 처리 회로는 (i) 재구축된 영역의 콘텐츠 및/또는 (ii) 상기 재구축된 영역의 예측 정보에 기초하여 인트라 블록 복사(intra block copy, IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정할 수 있다. 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 한 결정에 응답하

여, 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않고 상기 처리 회로는 상기 재구축된 영역을 포함하지 않는 상기 이전 로컬 참조 범위에 기초하는 상기 IBC 모드를 사용하여 상기 현재 꽉쳐의 코딩 블록(CB)을 디코딩한다.

[0024] 상기 처리 회로는 상기 재구축된 영역의 재구축된 샘플에 기초하여 상기 재구축된 영역의 콘텐츠를 결정하며, 상기 재구축된 영역의 콘텐츠는 상기 재구축된 영역에서의 텍스처 콘텐츠의 백분율 및/또는 스크린 콘텐츠의 백분율을 지시한다. 상기 처리 회로는 상기 재구축된 영역에서의 상기 텍스처 콘텐츠의 백분율 및/또는 상기 스크린 콘텐츠의 백분율에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정할 수 있다. 일례에서, 상기 재구축된 영역의 콘텐츠는 스크린 콘텐츠를 포함하지 않고, 상기 처리 회로는 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트하지 않기로 결정한다.

[0025] 일 실시예에서, 상기 처리 회로는 상기 영역의 예측 모드를 지시하는, 상기 영역의 예측 정보를 디코딩하고 상기 예측 모드가 미리 정의된 예측 모드의 세트에 있는지에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정한다.

[0026] 일 실시예에서, 상기 처리 회로는 상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트 할지를 지시하는, 상기 영역의 예측 정보를 디코딩한다. 상기 처리 회로는 상기 예측 정보에서, 상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지의 지시에 기초하여 상기 IBC 모드에 대한 상기 이전 로컬 참조 범위를 업데이트할지를 판정한다.

[0027] 일례에서, 상기 이전 로컬 참조 범위를 상기 재구축된 영역의 재구축된 샘플로 업데이트할지는 상기 영역의 예측 정보에 포함된 플래그에 의해 지시되며, 상기 처리 회로는 상기 플래그에 기초하여 상기 IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정한다. 일례에서, 상기 플래그는 상기 블록 내의 임의의 재구축된 샘플이 상기 IBC 모드에서 상기 로컬 참조 범위를 업데이트에 사용되는지를 지시한다.

[0028] 일 실시예에서, 상기 IBC 모드에 대한 이전 로컬 참조 범위는 업데이트되도록 결정되고, 상기 이전 로컬 참조 범위는 참조 샘플 메모리(reference sample memory, RSM)에 저장된 복수의 영역을 포함한다. 상기 복수의 영역은 참조 샘플을 포함할 수 있다. 상기 처리 회로는 상기 복수의 영역의 콘텐츠에 기초하여 상기 복수의 영역에 순위를 매길 수 있으며, 각각의 상기 콘텐츠는 상기 복수의 영역 각각에 대응한다. 상기 처리 회로는 상기 복수의 영역의 순위에 기초하여 상기 재구축된 영역의 재구축된 샘플로 대체될 상기 복수의 영역 중의 영역을 선택하고 선택된 영역을 상기 이전 로컬 참조 범위에서 상기 재구축된 영역의 상기 재구축된 샘플로 대체하여 현재 로컬 참조 범위를 생성할 수 있다.

[0029] 일례에서, 상기 복수의 영역 각각에 대응하는 각각의 상기 콘텐츠는 대응하는 영역에서의 스크린 콘텐츠의 백분율 또는 텍스처 콘텐츠의 백분율로 지시된다. 상기 처리 회로는 상기 복수의 영역의 상기 스크린 콘텐츠의 백분율 각각 또는 상기 텍스처 콘텐츠의 백분율 각각에 기초하여 상기 복수의 영역의 순위를 매긴다.

[0030] 일 실시예에서, 상기 IBC 모드에 대한 이전 로컬 참조 범위가 업데이트되도록 결정되고, 상기 이전 로컬 참조 범위는 RSM에 저장된 복수의 영역을 포함한다. 상기 복수의 영역의 각각의 영역은 미리 정의된 예측 모드 세트 내의 하나 이상의 예측 모드로 코딩되는 각각의 영역에 많은 수의 특별히 코딩된 참조 샘플을 포함한다. 상기 처리 회로는 상기 복수의 영역에서의 상기 특별히 코딩된 참조 샘플 각각의 수에 기초하여 상기 복수의 영역에 순위를 매길 수 있다. 상기 처리 회로는 상기 복수의 영역의 순위에 기초하여 상기 재구축된 영역의 재구축된 샘플로 대체될 상기 복수의 영역 중의 영역을 선택하고, 선택된 영역을 상기 이전 로컬 참조 범위에서 상기 재구축된 영역의 재구축된 샘플로 대체하여 현재 로컬 참조 범위를 생성할 수 있다.

[0031] 일 실시예에서, 상기 IBC 모드에 대한 이전 로컬 참조 범위가 업데이트되도록 결정되고; 상기 이전 로컬 참조 범위는 RSM에 저장된 복수의 영역을 포함한다. 상기 복수의 영역의 각각의 영역과 연관된 참조 빈도는 상기 각각의 영역이 상기 현재 꽉쳐 내의 CBs를 예측하는 데 사용되는 블록 벡터에 의해 참조되는 횟수를 지시한다. 상기 처리 회로는 상기 복수의 영역 각각의 참조 빈도에 기초하여 상기 복수의 영역에 순위를 매길 수 있다. 상기 처리 회로는 상기 복수의 영역의 순위에 기초하여 상기 재구축된 영역의 재구축된 샘플로 대체될 상기 복수의 영역 중의 영역을 선택하고 선택된 영역을 상기 이전 로컬 참조 범위에서 상기 재구축된 영역의 재구축된 샘플로 대체하여 현재 로컬 참조 범위를 생성할 수 있다.

[0032] 일 실시예에서, 상기 이전 로컬 참조 범위는 복수의 영역을 포함한다. 상기 복수의 영역 중 하나 내의 서브영역의 재구축된 샘플을 제외한 상기 복수의 영역의 재구축된 샘플을 RSM에 저장한다. 상기 복수의 영역 중 하나 내의 서브영역의 재구축된 샘플은 상기 RSM 내의 상기 복수 영역 중 하나 내의 서브영역에 대응하는 패딩된 샘플

(padded sample)을 저장하는 패딩 프로세스에 의해 대체된다.

[0033] 일 실시예에서, 상기 재구축된 영역이 재구축된 코딩 블록이거나, 상기 재구축된 영역의 크기는 메모리 업데이트 영역 유닛의 크기이다.

[0034] 본 개시의 측면은 또한 비디오 디코딩을 위한 방법을 수행하도록 적어도 하나의 프로세서에 의해 실행 가능한 프로그램을 저장하는, 컴퓨터로 판독 가능한 비일시적 저장 매체를 제공한다.

도면의 간단한 설명

[0035] 개시된 주제의 추가 특징, 성질 및 다양한 이점은 이하의 상세한 설명 및 첨부도면으로부터 더욱 명백해질 것이다.

도 1a는 인트라 예측 모드의 예시적인 서브세트의 개략도이다.

도 1b는 예시적인 인트라 예측 방향의 예시이다.

도 2는 현재 블록(201) 및 주변 샘플의 일례를 도시한다.

도 3은 통신 시스템(300)의 예시적인 블록도의 개략도이다.

도 4는 통신 시스템(400)의 예시적인 블록도의 개략도이다.

도 5는 디코더의 예시적인 블록도의 개략도이다.

도 6은 인코더의 예시적인 블록도의 개략도이다.

도 7은 예시적인 인코더의 블록도를 도시한다.

도 8은 예시적인 디코더의 블록도를 도시한다.

도 9는 본 개시의 실시예에 따른 현재 코딩 유닛과 연관된 블록 벡터를 도시한다.

도 10a~도 10d는 본 개시의 실시예에 따른 인트라 블록 복사(IBC) 모드에 대한 참조 영역을 도시한다.

도 11은 참조 샘플 메모리(RSM)의 예시적인 연속 업데이트 프로세스를 공간적으로 도시한다.

도 12는 제한되는 즉시 재구축된 영역의 일례를 도시한다.

도 13은 예시적인 메모리 재사용 메커니즘을 도시한다.

도 14는 현재 SB의 코딩 동안 RSM에서의 예시적인 메모리 업데이트 프로세스를 도시한다.

도 15는 현재 SB가 디코딩되고 있을 때 RSM에서의 예시적인 적응적 메모리 업데이트 프로세스를 도시한다.

도 16은 적응적 메모리 업데이트 프로세스에서 사용되는 예시적인 패딩 프로세스(padding process)를 나타낸다.

도 17은 RSM에서의 예시적인 적응적 메모리 업데이트 프로세스를 도시한다.

도 18은 본 개시의 일 실시예에 따른 인코딩 프로세스를 개략적으로 설명하는 흐름도를 도시한다.

도 19는 본 개시의 일 실시예에 따른 디코딩 프로세스를 개략적으로 설명하는 흐름도를 도시한다.

도 20은 일 실시예에 따른 컴퓨터 시스템의 개략도이다.

발명을 실시하기 위한 구체적인 내용

[0036] I. 비디오/이미지 코딩 기술의 개요

[0037] 도 3은 통신 시스템(300)의 간략화된 블록도를 나타낸다. 통신 시스템(300)은 예를 들어 네트워크(350)를 통해서로 통신할 수 있는 복수의 단말 기기를 포함한다. 예를 들어, 통신 시스템(300)은 네트워크(350)를 통해 상호 연결된 제1의 단말 기기 쌍(310, 320)을 포함한다. 도 3의 예에서, 제1의 단말 기기 쌍(310, 320)은 데이터의 단방향 송신을 수행한다. 예를 들어, 단말 기기(310)는 네트워크(350)를 통해 다른 단말 기기(320)로의 송신을 위해 비디오 데이터(예: 단말 기기(310)에 의해 캡처된 비디오 픽처의 스트림)를 코딩할 수 있다. 인코딩된 비디오 데이터는 하나 이상의 코딩된 비디오 비트스트림의 형태로 송신될 수 있다. 단말 기기(320)는 네트워크(350)로부터 코딩된 비디오 데이터를 수신하고, 코딩된 비디오 데이터를 디코딩하여 비디오 픽처를 복원하고,

복원된 비디오 데이터에 따라 비디오 픽처를 표시할 수 있다. 단방향 데이터 송신은 미디어 서빙 애플리케이션 등에 일반적일 수 있다.

[0038] 다른 예에서, 통신 시스템(300)은, 예를 들어 화상 회의 동안, 코딩된 비디오 데이터의 양방향 송신을 수행하는 제2의 단말 기기 쌍(330, 340)을 포함한다. 데이터의 양방향 송신의 경우, 일례에서, 단말 기기 쌍(330, 340)의 단말 기기 각각은 네트워크(350)를 통해 단말 기기 쌍(330, 340) 중 다른 단말 기기로의 송신을 위해 비디오 데이터(예: 단말 기기에 의해 캡처된 비디오 픽처의 스트림)를 코딩할 수 있다. 단말 기기 쌍(330, 340)의 단말 기기 각각은 또한 단말 기기 쌍(330, 340)의 다른 단말 기기에 의해 송신되는 코딩된 비디오 데이터를 수신할 수 있으며, 코딩된 비디오 데이터를 디코딩하여 비디오를 복원할 수 있고 복원된 비디오 데이터에 따라 액세스 가능한 디스플레이 기기에 비디오 픽처를 표시할 수 있다.

[0039] 도 3의 예에서, 단말 기기(310, 320, 330, 340)는 각각 서버, 개인용 컴퓨터 및 스마트폰으로 예시될 수 있지만, 본 개시의 원리는 그렇게 한정되지 않을 수 있다. 본 개시의 실시예는 랩톱 컴퓨터, 태블릿 컴퓨터, 미디어 플레이어 및/또는 전용 화상 회의 장비가 있는 애플리케이션을 찾아낸다. 네트워크(350)는, 예를 들어 유선(wired) 및/또는 무선 통신 네트워크를 포함한, 단말 기기(310, 320, 330, 340) 사이에서 코딩된 비디오 데이터를 전달하는 임의의 수의 네트워크를 나타낸다. 통신 네트워크(350)는 회선 교환 및/또는 패킷 교환 채널에서 데이터를 교환할 수 있다. 대표적인 네트워크로는 통신 네트워크, 근거리 통신망, 광역 통신망 및/또는 인터넷을 포함한다. 본 논의를 목적을 위해, 네트워크(350)의 아키텍처 및 토플로지는 이하에서 설명되지 않는 한 본 개시의 동작에 중요하지 않을 수 있다.

[0040] 도 4는 개시된 주제에 대한 애플리케이션의 일례로서, 스트리밍 환경에서 비디오 인코더 및 비디오 디코더를 예시한다. 개시된 주제는 예를 들어, 화상 회의 디지털 TV, 스트리밍 서비스, CD, DVD, 메모리 스틱 등을 포함하는 디지털 미디어에 압축된 비디오의 저장 등을 포함한, 다른 비디오 지원 애플리케이션(video enabled application)에도 동일하게 적용될 수 있다.

[0041] 스트리밍 시스템은 예를 들어 압축되지 않은 비디오 픽처의 스트림(402)을 생성하는 비디오 소스(401), 예를 들어 디지털 카메라를 포함할 수 있는 캡처 서브시스템(413)을 포함할 수 있다. 일례에서, 비디오 픽처의 스트림(402)은 디지털 카메라로 촬영되는 샘플을 포함한다. 인코딩된 비디오 데이터(404)(또는 코딩된 비디오 비트 스트림)와 비교할 때 높은 데이터 볼륨을 강조하기 위해 굵은 선으로 표시된 비디오 픽처의 스트림(402)은 비디오 소스(401)에 결합된 비디오 인코더(403)를 포함하는 전자 기기(420)에 의해 처리될 수 있다. 비디오 인코더(403)는 아래에서 더 상세히 설명되는 바와 같이 개시된 주제의 측면들을 가능하게 하거나 구현할 수 있는 하드웨어, 소프트웨어, 또는 이들의 조합을 포함할 수 있다. 비디오 픽처의 스트림(402)과 비교할 때 더 낮은 데이터 볼륨을 강조하기 위해 가는 선으로 표시된 인코딩된 비디오 데이터(404)(또는 인코딩된 비디오 비트스트림)은 나중에 사용하기 위해 스트리밍 서버(405)에 저장될 수 있다. 도 4에서의 클라이언트 서브시스템(406, 408)과 같은, 하나 이상의 스트리밍 클라이언트 서브시스템은 인코딩된 비디오 데이터(404)의 사본(407, 409)을 검색하기 위해 스트리밍 서버(405)에 액세스할 수 있다. 클라이언트 서브시스템(406)은 예를 들어, 전자 기기(430)에 비디오 디코더(410)를 포함할 수 있다. 비디오 디코더(410)는 인코딩된 비디오 데이터의 인커밍 사본(incoming copy)(407)을 디코딩하고 디스플레이(412)(예: 디스플레이 스크린) 또는 다른 렌더링 기기(도시되지 않음)에서 렌더링될 수 있는 비디오 픽처(411)의 아웃고잉 스트림을 생성한다. 일부 스트리밍 시스템에서, 인코딩된 비디오 데이터(404, 407, 409)(예: 비디오 비트 스트림)은 특정 비디오 코딩/압축 표준에 따라 인코딩될 수 있다. 이러한 표준의 예로는 ITU-T Recommendation H.265를 포함한다. 일례에서, 개발중인 비디오 코딩 표준은 비공식적으로 VVC(Versatile Video Coding)로 알려져 있다. 개시된 주제는 VVC의 컨택스트에서 사용될 수 있다.

[0042] 전자 기기(420, 430)는 다른 구성요소(도시되지 않음)를 포함할 수 있음에 유의한다. 예를 들어, 전자 기기(420)는 비디오 디코더(도시되지 않음)를 포함할 수 있고, 전자 기기(430)도 비디오 인코더(도시되지 않음)를 포함할 수 있다.

[0043] 도 5는 비디오 디코더(510)의 예시적인 블록도를 도시한다. 비디오 디코더(510)는 전자 기기(530)에 포함될 수 있다. 전자 기기(530)는 수신기(531)(예: 수신 회로)를 포함할 수 있다. 비디오 디코더(510)는 도 4의 예에서의 비디오 디코더(410) 대신에 사용될 수 있다.

[0044] 수신기(531)는 비디오 디코더(510)에 의해 디코딩될 하나 이상의 코딩된 비디오 시퀀스를 수신할 수 있다. 일 실시예에서, 하나의 코딩된 비디오 시퀀스가 한 번에 수신되며, 여기서 각각의 코딩된 비디오 시퀀스의 디코딩은 다른 코딩된 비디오 시퀀스의 디코딩과 독립적이다. 코딩된 비디오 시퀀스는 인코딩된 비디오 데이터를 저장

하는 저장 기기에 대한 하드웨어/소프트웨어 링크일 수 있는 채널(501)로부터 수신될 수 있다. 수신기(531)는 다른 데이터, 예를 들어, 코딩된 오디오 데이터 및/또는 보조 데이터 스트림과 함께 인코딩된 비디오 데이터를 수신할 수 있으며, 이는 엔티티(도시되지 않음)를 사용하여 그들 각각에 포워딩될 수 있다. 수신기(531)는 코딩된 비디오 시퀀스를 다른 데이터와 분리할 수 있다. 네트워크 jitter를 방지하기 위해, 버퍼 메모리(515)가 수신기(531)와 엔트로피 디코더/파서(520)(이하 "파서(520)") 사이에 결합될 수 있다. 특정 애플리케이션에서, 버퍼 메모리(515)는 비디오 디코더(510)의 일부이다. 다른 애플리케이션의 경우에는 비디오 디코더(510) 외부에 있을 수 있다(도시되지 않음). 또 다른 애플리케이션의 경우에는 예를 들어 네트워크 지터를 방지하기 위해 비디오 디코더(510) 외부에 버퍼 메모리(도시되지 않음)가 있을 수 있으며, 예를 들어 플레이아웃 타이밍(playout timing)을 처리하기 위해 비디오 디코더(510) 내부에 추가로 다른 버퍼 메모리(515)가 있을 수 있다. 수신기(531)가 충분한 대역폭과 제어성(controllability)을 갖는 저장/포워딩 기기로부터 또는 등시성 네트워크(isosynchronous network)로부터 데이터를 수신하고 있을 때, 버퍼 메모리(515)는 필요하지 않거나 작을 수 있다. 인터넷과 같은 최선형 패킷 네트워크(best effort packet network)에서 사용하기 위해, 버퍼 메모리(515)가 필요할 수 있고, 비교적 클 수 있으며 적응적인 크기가 유리할 수 있고, 운영 체제 또는 비디오 디코더(510)의 외부의 유사한 요소(도시되지 않음)에서 적어도 부분적으로 구현될 수 있다.

[0045] 비디오 디코더(510)는 코딩된 비디오 시퀀스로부터 심볼(521)을 재구축하기 위해 파서(520)를 포함할 수 있다. 이러한 심볼의 카테고리로는 비디오 디코더(510)의 동작을 관리하는 데 사용되는 정보와, 도 5에 도시된 바와 같이, 전자 기기(530)의 일체화된 부분이 아니라 전자 기기(530)에 결합될 수 있는 렌더 기기(512)(예: 디스플레이 스크린)와 같은 렌더링 기기를 제어하기 위한 잠재적인 정보를 포함한다. 렌더링 기기(들)에 대한 제어 정보는 보충 강화 정보(Supplemental Enhancement Information, SEI) 메시지 또는 비디오 유용성 정보(Video Usability Information, VUI) 파라미터 세트 프래그먼트(도시되지 않음)의 형태일 수 있다. 파서(520)는 수신되는 코딩된 비디오 시퀀스를 파싱/엔트로피 디코딩할 수 있다. 코딩된 비디오 시퀀스의 코딩은 비디오 코딩 기술 또는 표준에 따른 것일 수 있으며, 가변 길이 코딩, Huffman 코딩, 컨텍스트 민감도가 있거나 없는 산술 코딩 등을 포함한 다양한 원리를 따를 수 있다. 파서(520)는 비디오 디코더에서의 핵심의 서브그룹 중 적어도 하나에 대한 서브그룹 파라미터의 세트를, 그 그룹에 대응하는 적어도 하나의 파라미터에 기초하여, 코딩된 비디오 시퀀스로부터 추출할 수 있다. 서브그룹은 핵심의 그룹(Group of Picture, GOP), 핵심, 타일(tile), 슬라이스(slice), 매크로블록(macroblock), 코딩 유닛(Coding Unit, CU), 블록, 변환 유닛(Transform Unit, TU), 예측 유닛(Prediction Unit, PU) 등을 포함할 수 있다. 파서(520)는 또한 변환 계수, 양자화기 파라미터 값, 움직임 벡터 등과 같은 코딩된 비디오 시퀀스 정보로부터 추출할 수 있다.

[0046] 파서(520)는 버퍼 메모리(515)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩/파싱 동작을 수행하여 심볼(521)을 생성할 수 있다.

[0047] 심볼(521)의 재구축은 코딩된 비디오 핵심 또는 그 일부(예: 인터 및 인트라 핵심, 인터 및 인트라 블록)의 유형 및 기타 인자에 따라 다수의 다른 유닛과 관련 있을 수 있다. 관련되는 유닛 및 어떻게 관련되는지는 파서(520)에 의해 코딩된 비디오 시퀀스로부터 파싱된 서브그룹 제어 정보에 의해 제어될 수 있다. 파서(520)와 그 아래의 다수의 유닛 사이의 그러한 서브그룹 제어 정보의 흐름은 단순함을 위해 나타내지 않았다.

[0048] 이미 언급된 기능 블록을 넘어, 비디오 디코더(510)는 아래에 설명된 바와 같이 개념적으로 다수의 기능 유닛으로 세분될 수 있다. 상업적 제약 하에서 동작하는 실제 구현에서, 이러한 유닛 중 다수는 서로 밀접하게 상호작용하며, 적어도 부분적으로, 서로 통합될 수 있다. 그러나 개시된 주제를 명확하게 설명하기 위해서는 아래의 기능 유닛들로의 개념적 세분화가 적절하다.

[0049] 제1 유닛은 스케일러/역변환 유닛(551)이다. 스케일러/역변환 유닛(551)은 파서(520)로부터의 심볼(들)(521)을 사용할 역변환, 블록 크기, 양자화 인자, 양자화 스케일링 행렬 등을 포함한 제어 정보뿐만 아니라 양자화된 변환 계수를 수신한다. 스케일러/역변환 유닛(551)은 집성기(aggregator)(555)에 입력될 수 있는, 샘플 값으로 포함하는 블록을 출력할 수 있다.

[0050] 경우에 따라서는, 스케일러/역변환 유닛(551)의 출력 샘플은 인트라 코딩된 블록에 관련될 수 있다. 인트라 코딩된 블록은 이전에 재구축된 핵심로부터의 예측 정보를 사용하지 않고, 현재 핵심의 이전에 재구축된 부분으로부터의 예측 정보를 사용할 수 있는 블록이다. 이러한 예측 정보는 인트라 핵심 예측 유닛(552)에 의해 제공될 수 있다. 경우에 따라서는, 인트라 핵심 예측 유닛(552)은 현재 핵심 버퍼(558)에서 인출되는 이미 재구축된 주변 정보를 사용하여, 재구축중인 블록과 동일한 크기 및 모양의 블록을 생성한다. 현재 핵심 버퍼(558)는, 예를 들어 부분적으로 재구축된 현재 핵심 및/또는 완전히 재구축된 현재 핵심을 버퍼링한다. 집성기(555)는, 경우에

따라서는, 샘플 단위로, 인트라 예측 유닛(552)이 생성한 예측 정보를 스케일러/역변환 유닛(551)에 의해 제공되는 출력 샘플 정보에 추가할 수 있다.

[0051] 다른 경우에, 스케일러/역변환 유닛(551)의 출력 샘플은 인터 코딩되고 잠재적으로 움직임 보상된 블록에 관련될 수 있다. 이러한 경우, 움직임 보상 예측 유닛(553)은 인터 픽처 예측에 사용되는 샘플을 인출하기 위해 참조 픽처 메모리(557)에 액세스할 수 있다. 인출된 샘플을 블록에 관련되는 심볼(521)에 따라 움직임 보상한 후, 이러한 샘플은 집성기(555)에 의해 스케일러/역변환 유닛(551)의 출력(이 경우 잔차 샘플 또는 잔차 신호라고 할 수 있음)에 더해져, 출력 샘플 정보를 생성할 수 있다. 움직임 보상 예측 유닛(553)이 예측 샘플을 인출하는 참조 픽처 메모리(557) 내의 주소는 움직임 벡터에 의해 제어될 수 있으며, 예를 들어 X, Y 및 참조 픽처 성분을 가질 수 있는 심볼(521)의 형태로 움직임 보상 예측 유닛(553)에 사용할 수 있다. 움직임 보상은 또한 서브 샘플의 정확한 움직임 벡터가 사용되고 있을 때에 참조 픽처 메모리(557)로부터 인출된 샘플 값의 보간, 움직임 벡터 예측 메커니즘 등도 포함할 수 있다.

[0052] 집성기(555)의 출력 샘플은 루프 필터 유닛(556)에서 다양한 루프 필터링 기술의 대상이 될 수 있다. 비디오 압축 기술은 코딩된 비디오 시퀀스(코딩된 비디오 비트 스트림이라고도 함)에 포함된 파라미터에 의해 제어되고 파서(520)로부터의 심볼(521)로서 루프 필터 유닛(556)에 사용 가능하게 되는 인 루프(in-loop) 필터 기술을 포함할 수 있다. 비디오 압축은 또한 코딩된 픽처 또는 코딩된 비디오 시퀀스의 이전(디코딩 순서상) 부분을 디코딩하는 동안 획득된 메타 정보에 응답할 수 있을 뿐만 아니라 이전에 재구축되고 루프 필터링된 샘플 값에 응답할 수도 있다.

[0053] 루프 필터 유닛(556)의 출력은 렌더 기기(512)에 출력될 수 있고 미래의 인터 픽처 예측에 사용하기 위해 참조 픽처 메모리(557)에 저장될 수 있는 샘플 스트림일 수 있다.

[0054] 일단 완전히 재구축된 특정 코딩된 픽처는 나중의 예측을 위한 참조 픽처로서 사용할 수 있다. 예를 들어, 현재 픽처에 대응하는 코딩된 픽처가 완전히 재구축되고 코딩된 픽처가 참조 픽처로 식별되면(예: 파서(520)에 의해), 현재 픽처 버퍼(558)는 참조 픽처 메모리(557)의 일부가 될 수 있고, 새로운 현재 픽처 버퍼는 다음 코딩된 픽처의 재구축을 시작하기 전에 재할당될 수 있다.

[0055] 비디오 디코더(510)는 ITU-T Rec. H.265와 같은, 미리 정해진 비디오 압축 기술 또는 표준에 따라 디코딩 동작을 수행할 수 있다. 코딩된 비디오 시퀀스는 비디오 압축 기술 또는 표준의 신팩스와 비디오 압축 기술에 문서화된 프로파일 모두에 준거한다는 점에서, 사용되는 비디오 압축 기술 또는 표준에 의해 지정된 신팩스를 따를 수 있다. 구체적으로, 프로파일은 비디오 압축 기술 또는 표준에서 사용 가능한 모든 도구 중에서 그 프로파일에서 사용할 수 있는 유일한 도구로서 특정 도구를 선택할 수 있다. 또한 준거에 필요한 것은 코딩된 비디오 시퀀스의 복잡도가 비디오 압축 기술 또는 표준의 레벨에 정의된 범위 내에 있는 것이다. 경우에 따라서는, 레벨은 최대 픽처 크기, 최대 프레임 레이트, 최대 재구축 샘플 레이트(예: 초당 메가 샘플로 측정됨), 최대 참조 픽처 크기 등을 제한한다. 레벨에 의해 설정된 제한은, 경우에 따라서는 코딩된 비디오 시퀀스에서 시그널링된 가상 참조 디코더(Hypothetical Reference Decoder, HRD) 버퍼 관리를 위한 HRD 사양 및 메타 데이터를 통해 추가로 제한될 수 있다.

[0056] 일 실시예에서, 수신기(531)는 인코딩된 비디오와 함께 추가(중복) 데이터를 수신할 수 있다. 추가 데이터는 코딩된 비디오 시퀀스(들)의 일부로서 포함될 수 있다. 추가 데이터는 데이터를 적절하게 디코딩하고/하거나 원본 비디오 데이터를 더 정확하게 재구축하기 위해 비디오 디코더(510)에 의해 사용될 수 있다. 추가 데이터는, 예를 들어 시간 계층, 공간 계층 또는 SNR 강화 계층(enhancement layer), 중복 슬라이스(redundant slice), 중복 픽처(redundant picture), 순방향 오류 정정 코드(forward error correction code) 등의 형태일 수 있다.

[0057] 도 6은 비디오 인코더(603)의 예시적인 블록도를 도시한다. 비디오 인코더(603)는 전자 기기(620)에 포함된다. 전자 기기(620)는 송신기(640)(예: 송신 회로)를 포함한다. 비디오 인코더(603)는 도 4의 예에서의 비디오 인코더(403) 대신에 사용될 수 있다.

[0058] 비디오 인코더(603)는 비디오 인코더(603)에 의해 코딩될 비디오 이미지(들)를 캡처할 수 있는 비디오 소스(601)(도 6의 예에서 전자 기기(620)의 일부가 아님)로부터 비디오 샘플을 수신할 수 있다. 다른 예에서, 비디오 소스(601)는 전자 기기(620)의 일부이다.

[0059] 비디오 소스(601)는 임의의 적절한 비트 심도(bit depth)(예: 8비트, 10비트, 12비트, …), 임의의 색 공간(color space)(예: BT.601 Y CrCb, RGB, …) 및 임의의 적절한 샘플링 구조(예: Y CrCb 4:2:0, Y CrCb 4:4:4) 일 수 있는 디지털 비디오 샘플 스트림의 형태로 비디오 인코더(603)에 의해 코딩될 소스 비디오 시퀀스를 제공

할 수 있다. 미디어 서빙 시스템에서, 비디오 소스(601)는 미리 준비된 비디오를 저장할 수 있는 저장 기기일 수 있다. 영상 회의 시스템에서, 비디오 소스(601)는 로컬 이미지 정보를 비디오 시퀀스로서 캡처하는 카메라일 수 있다. 비디오 데이터는 순차적으로 보여질 때 움직임을 부여하는 복수의 개별 픽처로서 제공될 수 있다. 픽처 자체는 픽셀의 공간 배열(spatial array)로 구성될 수 있으며, 여기서 각각의 픽셀은 사용중인 샘플링 구조, 색 공간 등에 따라 하나 이상의 샘플을 포함할 수 있다. 당업자라면 픽셀과 샘플 사이의 관계를 쉽게 이해할 수 있을 것이다. 아래 설명은 샘플에 중점을 둔다.

[0060] 일 실시예에 따르면, 비디오 인코더(603)는 소스 비디오 시퀀스의 픽처를 실시간으로 또는 요구되는 임의의 다른 시간 제약하에서 코딩된 비디오 시퀀스(643)로 코딩하고 압축할 수 있다. 적절한 코딩 속도를 강제하는 것은 제어기(650)의 한 가지 기능이다. 일부 실시예에서, 제어기(650)는 후술되는 바와 같이 다른 기능 유닛을 제어하고 다른 기능 유닛에 기능적으로 결합된다. 명확함을 위해 그 결합은 표시되지 않는다. 제어기(650)에 의해 설정되는 파라미터로는 레이트 제어 관련 파라미터(픽처 스킵, 양자화기, 레이트 왜곡 최적화 기술의 람다 값, …), 픽처 크기, 픽처 그룹(GOP) 레이아웃, 최대 움직임 벡터 검색 범위, 등을 포함할 수 있다. 제어기(650)는 특정 시스템 설계에 최적화된 비디오 인코더(603)에 관련 있는 다른 적절한 기능을 갖도록 구성될 수 있다.

[0061] 일부 실시예에서, 비디오 인코더(603)는 코딩 루프에서 동작하도록 구성된다. 과도하게 단순화된 설명으로서, 일례에서, 코딩 루프는 소스 코더(630)(예: 코딩될 입력 픽처에 기초하여 심볼 스트림과 같은 심볼 및 참조 픽처(들)의 생성을 담당), 및 비디오 인코더(603)에 내장된 (로컬) 디코더(633)를 포함할 수 있다. 디코더(633)는 (원격) 디코더도 생성하였을 것과 유사한 방식으로 샘플 데이터를 생성하기 위해 샘볼을 재구축한다. 재구축된 샘플 스트림(샘플 데이터)은 참조 픽처 메모리(634)에 입력된다. 심볼 스트림의 디코딩은 디코더 위치(로컬 또는 원격)와 무관하게 비트가 정확한 결과(bit-exact result)로 이어지므로, 참조 픽처 메모리(634)의 내용도 로컬 인코더와 원격 인코더 사이에서 비트가 정확하다(bit exact). 다시 말해, 인코더의 예측 부분은 디코딩 동안에 예측을 사용하는 경우에 디코더가 "인식하는(see)" 것과 정확히 동일한 샘플 값을 참조 픽처 샘플로서 "인식 한다". 참조 픽처 동시성(reference picture synchronicity)(및 예를 들어 채널 오차로 인해 동시성이 유지될 수 없는 경우, 결과 드리프트)의 이러한 기본 원리는 일부 관련 기술에서도 사용된다.

[0062] "로컬" 디코더(633)의 동작은 비디오 디코더(510)와 같은, "원격" 디코더의 동작과 동일할 수 있으며, 이는 이미 도 5와 관련하여 상세하게 설명되었다. 도 5를 간단히 다시 참조하면, 심볼이 이용 가능하고 엔트로피 코더(645) 및 파서(420)에 의해 코딩된 비디오 시퀀스로의 심볼의 인코딩/디코딩이 무손실일 수 있으므로, 비디오 디코더(510)의 엔트로피 디코딩 부분은, 베퍼 메모리(515) 및 파서(520)를 포함하여, 로컬 디코더(633)에 인코더에 전적으로 구현되지 않을 수 있다.

[0063] 일 실시예에서, 디코더에 존재하는 파싱/엔트로피 디코딩을 제외한 디코더 기술이 대응하는 인코더에서, 동일하거나 실질적으로 동일한 기능적 형태로 존재한다. 이에 따라, 개시된 주제는 디코더 동작에 중점을 둔다. 따라서 인코더 기술에 대한 설명은 포괄적으로 설명된 디코더 기술의 반대이므로 생략될 수 있다. 특정 영역에서 더 자세한 설명이 아래에 제공된다.

[0064] 동작하는 동안, 일부 예에서, 소스 코더(630)는 "참조 픽처"로 지정된 비디오 시퀀스로부터 하나 이상의 이전에 코딩된 픽처를 참조하여 입력 픽처를 예측적으로 코딩하는 움직임 보상된 예측 코딩을 수행할 수 있다. 이러한 방식으로, 코딩 엔진(632)은 입력 픽처에 대한 예측 참조(들)로서 선택될 수 있는 입력 픽처의 픽셀 블록과 참조 픽처(들)의 픽셀 블록 간의 차이를 코딩한다.

[0065] 로컬 비디오 디코더(633)는 소스 코더(630)에 의해 생성된 심볼에 기초하여, 참조 픽처로서 지정될 수 있는 픽처의 코딩된 비디오 데이터를 디코딩할 수 있다. 코딩 엔진(632)의 동작은 유리하게 손실 프로세스일 수 있다. 코딩된 비디오 데이터가 비디오 디코더(도 6에 도시되지 않음)에서 디코딩될 수 있을 때, 재구축된 비디오 시퀀스는 일반적으로 약간의 오류가 있는 소스 비디오 시퀀스의 복제본(replica)일 수 있다. 로컬 비디오 디코더(633)는 참조 픽처에 대해 비디오 디코더에 의해 수행될 수 있는 디코딩 프로세스를 복제하고, 재구축된 참조 픽처가 참조 픽처 메모리(634)에 저장되도록 할 수 있다. 이러한 방식으로, 비디오 인코더(603)는 원단(far-end) 비디오 디코더(송신 오류 없음)에 의해 획득될 재구축된 참조 픽처로서 공통 콘텐츠를 갖는 재구축된 참조 픽처의 사본을 로컬에 저장할 수 있다.

[0066] 예측기(predictor)(635)는 코딩 엔진(632)에 대한 예측 검색을 수행할 수 있다. 즉, 코딩될 새로운 픽처를 위해, 예측기(635)는 참조 픽처 메모리(634)에서 새로운 픽처에 대한 적절한 예측 참조의 역할을 할 수 있는 샘플 데이터(후보 참조 픽셀 블록임), 또는 참조 픽처 움직임 벡터, 블록 모양 등과 같은 특정 메타데이터를 검색할 수 있다. 예측기(635)는 적절한 예측 참조를 찾기 위해 샘플 블록별 픽셀 블록(sample block-by-pixel

block) 단위로 동작할 수 있다. 경우에 따라서는, 예측기(635)에 의해 획득된 검색 결과에 의해 결정되는 바와 같이, 입력 픽처는 참조 픽처 메모리(634)에 저장된 다수의 참조 픽처로부터 끌어낸 예측 참조를 가질 수 있다.

[0067] 제어기(650)는, 예를 들어 비디오 데이터를 인코딩하기 위해 사용되는 파라미터 및 서브그룹 파라미터의 설정을 포함하는 소스 코더(630)의 코딩 동작을 관리할 수 있다.

[0068] 전술한 모든 기능 유닛의 출력은 엔트로피 코더(645)에서의 엔트로피 코딩 대상일 수 있다. 엔트로피 코더(645)는 Huffman 코딩, 가변 길이 코딩, 산술 코딩 등과 같은 기술에 따라 심볼에 무손실 압축을 적용함으로써, 다양한 기능 유닛에 의해 생성된 심볼을 코딩된 비디오 시퀀스로 변환한다.

[0069] 송신기(640)는 엔트로피 코더(645)에 의해 생성된 코딩된 비디오 시퀀스(들)를 버퍼링하여, 인코딩된 비디오 데이터를 저장할 저장 기기에 대한 하드웨어/소프트웨어 링크일 수 있는 통신 채널(660)을 통한 송신을 준비할 수 있다. 송신기(640)는 비디오 인코더(603)로부터 코딩된 비디오 데이터를 송신될 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림(소스는 도시되지 않음)과 병합할 수 있다.

[0070] 제어기(650)는 비디오 인코더(603)의 동작을 관리할 수 있다. 코딩하는 동안, 제어기(650)는 각각의 코딩된 픽처에 특정 코딩된 픽처 유형을 할당할 수 있으며, 이는 각각의 픽처에 적용될 수 있는 코딩 기술에 영향을 미칠 수 있다. 예를 들어, 픽처는 종종 다음 픽처 유형 중 하나로서 지정될 수 있다:

[0071] 인트라 픽처(Intra Picture: I 픽처)는 예측 소스로서 시퀀스 내의 어떤 다른 픽처도 사용하지 않고 코딩 및 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱은, 예를 들어 독립 디코더 리프레시 픽처(Independent Decoder Refresh Picture, "IDR")을 포함한, 상이한 유형의 인트라 픽처를 허용한다. 당업자는 I 픽처의 이러한 변형 및 각각의 적용 및 특징을 알고 있다.

[0072] 예측 픽처(Predictive picture: P 픽처)는 각각의 블록의 샘플 값을 예측하기 위해 최대 하나의 움직임 벡터 및 참조 색인을 사용하는 인트라 예측 또는 인터 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다.

[0073] 양 방향 예측 픽처(Bi-directionally Predictive Picture: B 픽처)는 각각의 블록의 샘플 값을 예측하기 위해 최대 두 개의 움직임 벡터 및 참조 색인을 사용하는 인트라 예측 또는 인터 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다. 유사하게, 다중 예측 픽처(multiple-predictive picture)는 단일 블록의 재구축을 위해 두 개보다 많은 참조 픽처와 연관 메타데이터를 사용할 수 있다.

[0074] 소스 픽처는 일반적으로 공간적으로 복수의 샘플 블록(예: 4×4 , 8×8 , 4×8 또는 16×16 샘플 블록)으로 세분화되고 블록 단위로 코딩될 수 있다. 블록은 블록의 픽처 각각에 적용된 코딩 할당에 의해 결정된 대로 다른 (이미 코딩된) 블록을 참조하여 예측적으로 코딩될 수 있다. 예를 들어, I 픽처의 블록은 비 예측적으로 코딩되거나, 동일한 픽처의 이미 코딩된 블록(공간 예측(spatial prediction) 또는 인트라 예측)을 참조하여 예측적으로 코딩될 수 있다. P 픽처의 픽셀 블록은 이전에 코딩된 하나의 참조 픽처를 참조하여 공간 예측을 통해 또는 시간 예측(temporal prediction)을 통해 예측적으로 코딩될 수 있다. B 픽처의 블록은 이전에 코딩된 하나 또는 두 개의 참조 픽처를 참조하여 공간 예측을 통해 또는 시간 예측을 통해 예측적으로 코딩될 수 있다.

[0075] 비디오 인코더(603)는 ITU-T Rec. H.265.와 같은 미리 결정된 비디오 코딩 기술 또는 표준에 따라 코딩 동작을 수행할 수 있다. 그 동작에서, 비디오 인코더(603)는 입력 비디오 시퀀스에서의 시간적 및 공간적 중복성을 이용하는 예측 코딩 동작을 포함한, 다양한 압축 동작을 수행할 수 있다. 따라서, 코딩된 비디오 데이터는 사용되는 비디오 코딩 기술 또는 표준에 의해 지정된 신택스를 따를 수 있다.

[0076] 일 실시예에서, 송신기(640)는 인코딩된 비디오와 함께 추가 데이터를 송신할 수 있다. 소스 코더(630)는 코딩된 비디오 시퀀스의 일부로서 이러한 데이터를 포함할 수 있다. 추가 데이터는 시간/공간/SNR 강화 계층, 중복 픽처 및 슬라이스와 같은 다른 형태의 중복 데이터, SEI 메시지, VUI 파라미터 세트 프래그먼트 등을 포함할 수 있다.

[0077] 비디오는 시간적 시퀀스로 복수의 소스 픽처(비디오 픽처)로서 캡처될 수 있다. 인트라 픽처 예측(종종 인트라 예측으로 축약됨)은 주어진 픽처에서 공간적 상관관계를 이용하고, 인터 픽처 예측(inter-picture prediction)은 픽처 사이의 (시간적 또는 기타) 상관관계를 이용한다. 일례에서, 현재 픽처라고 하는 인코딩/디코딩 중인 특정 픽처는 블록으로 파티셔닝될 수 있다. 현재 픽처 내의 블록이 이전에 코딩되고 비디오에서 여전히 버퍼링된 참조 픽처의 참조 블록과 유사한 경우, 현재 픽처 내의 블록은 움직임 벡터라고 하는 벡터에 의해 코딩될 수 있다. 움직임 벡터는 참조 픽처 내의 참조 블록을 가리키며, 다수의 참조 픽처가 사용 중인 경우, 참조 픽처를 식별하는 3차원을 가질 수 있다.

- [0078] 일부 실시예에서, 양-예측 기술(bi-prediction technique)이 인터 픽처 예측에 사용될 수 있다. 양-예측 기술에 따르면, 비디오에서 디코딩 순서상 현재 픽처에 둘다 선행하는 (하지만 각각 표시 순서상 과거 또는 미래일 수 있는) 제1 참조 픽처와 제2 참조 픽처와 같은, 2개의 참조 픽처가 사용된다. 현재 픽처의 블록은 제1 참조 픽처의 제1 참조 블록을 가리키는 제1 움직임 벡터 및 제2 참조 픽처의 제2 참조 블록을 가리키는 제2 움직임 벡터에 의해 코딩될 수 있다. 블록은 제1 참조 블록과 제2 참조 블록의 조합에 의해 예측될 수 있다.
- [0079] 또한, 인터 픽처 예측에 병합 모드 기술을 사용하여 코딩 효율을 향상시킬 수 있다.
- [0080] 본 개시의 일부 실시예에 따르면, 인터 픽처 예측 및 인트라 픽처 예측과 같은 예측은 블록 단위로 수행된다. 예를 들어, HEVC 표준에 따라, 비디오 픽처의 시퀀스에서의 픽처는 압축을 위해 코딩 트리 유닛(coding tree unit, CTU)로 파티셔닝되고, 픽처 내의 CTU는 64×64 픽셀, 32×32 픽셀 또는 16×16 픽셀과 같이, 동일한 크기를 가질 수 있다. 일반적으로, CTU는 루마 CTB 1개와 크로마 CTB 2개인 코딩 트리 블록(coding tree block, CTB)을 포함할 수 있다. 각각의 CTU는 반복적으로 하나 또는 복수의 코딩 유닛(CU)으로 워드트리 분할될 수 있다. 예를 들어 64×64 픽셀의 CTU는 64×64 픽셀의 CU 1개 또는 32×32 픽셀의 CU 4개 또는 16×16 픽셀의 CU 16개로 분할될 수 있다. 일례에서, 각각의 CU는 인터 예측 유형 또는 인트라 예측 유형과 같은 CU에 대한 예측 유형을 결정하기 위해 분석될 수 있다. CU는 시간적 및/또는 공간 예측 가능성에 따라 하나 이상의 예측 유닛(prediction unit, PU)으로 분할될 수 있다. 일반적으로, 각각의 PU는 루마 예측 블록(PB)과 두 개의 크로마 PB를 포함한다. 일 실시예에서, 코딩(인코딩/디코딩)에서의 예측 동작은 예측 블록의 단위로 수행된다. 루마 예측 블록을 예측 블록의 일례로 사용하면, 예측 블록은 8×8 픽셀, 16×16 픽셀, 8×16 픽셀, 16×8 픽셀 등과 같은, 픽셀에 대한 값(예: 루마 값)의 행렬을 포함한다.
- [0081] 도 7은 비디오 인코더(703)의 예시적인 도면을 도시한다. 비디오 인코더(703)는 비디오 픽처의 시퀀스에서 현재 비디오 픽처 내의 샘플 값의 처리 블록(예: 예측 블록)을 수신하고, 처리 블록을 코딩된 비디오 시퀀스의 일부인 코딩된 픽처로 인코딩하도록 구성된다. 일례에서, 비디오 인코더(703)는 도 4의 예에서의 비디오 인코더(403) 대신에 사용될 수 있다.
- [0082] HEVC의 예에서, 비디오 인코더(703)는 8×8 샘플의 예측 블록 등과 같은 처리 블록에 대한 샘플 값의 행렬을 수신한다. 비디오 인코더(703)는 처리 블록이, 예를 들어 레이트 왜곡 최적화(rate-distortion optimization)를 사용하는 인트라 모드, 인터 모드 또는 양-예측 모드를 사용하여 가장 잘 코딩되는지를 판정한다. 처리 블록이 인트라 모드로 코딩되는 것으로 결정되는 경우, 비디오 인코더(703)는 처리 블록을 코딩된 픽처로 인코딩하기 위해 인트라 예측 기술을 사용할 수 있고; 처리 블록이 인터 모드 또는 양-예측 모드로 코딩되는 것으로 결정되는 경우, 비디오 인코더(703)는 처리 블록을 코딩된 픽처로 인코딩하기 위해 인터 예측 또는 양-예측 기술을 각각 사용할 수 있다. 특정 비디오 코딩 기술에서, 병합 모드는 움직임 벡터가 예측기 외부의 코딩된 움직임 벡터 성분의 이점 없이 하나 이상의 움직임 벡터 예측기로부터 도출되는 인터 픽처 예측의 서브모드일 수 있다. 특정 다른 비디오 코딩 기술에서, 대상 블록(subject block)에 적용 가능한 움직임 벡터 성분이 존재할 수 있다. 일례에서, 비디오 인코더(703)는 처리 블록의 예측 모드를 결정하기 위한 모드 결정 모듈(도시되지 않음)과 같은, 구성요소를 포함할 수 있다.
- [0083] 도 7의 예에서, 비디오 인코더(703)는 인터 인코더(730), 인트라 인코더(722), 잔차 계산기(723), 스위치(726), 잔차 인코더(724), 일반 제어기(721), 및 도 7에 도시된 바와 같이 함께 결합된 엔트로피 인코더(725)를 포함한다.
- [0084] 인터 인코더(730)는 현재 블록(예: 처리 블록)의 샘플을 수신하고, 블록을 참조 픽처 내의 하나 이상의 참조 블록(예: 이전 픽처 및 이후 픽처 내의 블록)과 비교하고, 인터 예측 정보(예: 인터 인코딩 기술에 따른 중복 정보의 설명, 움직임 벡터, 병합 모드 정보)를 생성하고, 임의의 적절한 기술을 사용하여 인터 예측 정보에 기초하여 인터 예측 결과(예: 예측된 블록)를 계산하도록 구성된다. 일부 예에서, 참조 픽처는 인코딩된 비디오 정보에 기초하여 디코딩되는 디코딩된 참조 픽처이다.
- [0085] 인트라 인코더(722)는 현재 블록(예: 처리 블록)의 샘플을 수신하고, 경우에 따라서는 그 블록을 동일한 픽처에서 이미 코딩된 블록과 비교하고, 변환 후 양자화된 계수를, 경우에 따라서는 또한 인트라 예측 정보(예: 하나 이상의 인트라 인코딩 기술에 따른 인트라 예측 방향 정보)를 생성하도록 구성된다. 일례에서, 인트라 인코더(722)는 또한 동일한 픽처의 인트라 예측 정보 및 참조 블록에 기초하여 인트라 예측 결과(예: 예측된 블록)를 계산한다.
- [0086] 일반 제어기(721)는 일반 제어 데이터를 결정하고 일반 제어 데이터에 기초하여 비디오 인코더(703)의 다른 구

성요소를 제어하도록 구성된다. 일례에서, 일반 제어기(721)는 블록의 모드를 결정하고, 모드에 기초하여 제어 신호를 스위치(726)에 제공한다. 예를 들어, 모드가 인트라 모드인 경우, 일반 제어기(721)는 스위치(726)를 제어하여 잔차 계산기(723)에서 사용하기 위한 인트라 모드 결과를 선택하도록 하고, 엔트로피 인코더(725)를 제어하여 인트라 예측 정보를 선택하여 비트 스트림에 인트라 예측 정보를 포함시키도록 하고; 모드가 인터 모드인 경우, 일반 제어기(721)는 스위치(726)를 제어하여 잔차 계산기(723)에서 사용하기 위한 인터 예측 결과를 선택하도록 하고, 엔트로피 인코더(725)를 제어하여 인터 예측 정보를 선택하여 인터 예측 정보를 비트 스트림에 포함시키도록 한다.

[0087] 잔차 계산기(723)는 수신된 블록과 인트라 인코더(722) 또는 인터 인코더(730)로부터 선택된 예측 결과 사이의 차이(잔차 데이터)를 계산하도록 구성된다. 잔차 인코더(724)는 잔차 데이터에 기초하여 동작하여 잔차 데이터를 인코딩하여 변환 계수를 생성하도록 구성된다. 일례에서, 잔차 인코더(724)는 잔차 데이터를 공간 도메인에서 주파수 도메인으로 변환하고, 변환 계수를 생성하도록 구성된다. 그런 다음 변환 계수는 양자화된 변환 계수를 얻기 위한 양자화 처리를 거친다. 다양한 실시예에서, 비디오 인코더(703)는 또한 잔차 디코더(728)를 포함한다. 잔차 디코더(728)는 역변환을 수행하고, 디코딩된 잔차 데이터를 생성하도록 구성된다. 디코딩된 잔차 데이터는 인트라 인코더(722) 및 인터 인코더(730)에 의해 적절하게 사용될 수 있다. 예를 들어, 인터 인코더(730)는 디코딩된 잔차 데이터 및 인터 예측 정보에 기초하여 디코딩된 블록을 생성할 수 있고, 인트라 인코더(722)는 디코딩된 잔차 데이터 및 인트라 예측 정보에 기초하여 디코딩된 블록을 생성할 수 있다. 디코딩된 블록은 디코딩된 픽처를 생성하기 위해 적절하게 처리되고 디코딩된 픽처는 메모리 회로(도시되지 않음)에서 버퍼링될 수 있고 일부 예에서 참조 픽처로서 사용될 수 있다.

[0088] 엔트로피 인코더(725)는 인코딩된 블록을 포함하도록 비트 스트림을 포맷하도록 구성된다. 엔트로피 인코더(725)는 HEVC 표준과 같은 적절한 표준에 따라 비트스트림에 다양한 정보를 포함하도록 구성된다. 일례에서, 엔트로피 인코더(725)는 일반 제어 데이터, 선택된 예측 정보(예: 인트라 예측 정보 또는 인터 예측 정보), 잔차 정보 및 기타 적절한 정보를 비트스트림에 포함하도록 구성된다. 개시된 주제에 따르면, 인터 모드 또는 양-예측 모드의 병합 서브모드에서 블록을 코딩하는 경우, 잔차 정보가 없을 수 있음에 유의한다.

[0089] 도 8은 비디오 디코더(810)의 예시적인 도면을 도시한다. 비디오 디코더(810)는 코딩된 비디오 시퀀스의 일부인 코딩된 픽처를 수신하고, 코딩된 픽처를 디코딩하여 재구축된 픽처를 생성하도록 구성된다. 일례에서, 비디오 디코더(810)는 도 4의 예에서 비디오 디코더(410) 대신에 사용될 수 있다.

[0090] 도 8의 예에서, 비디오 디코더(810)는 도 8의 도시된 바와 같이 함께 결합된 엔트로피 디코더(871), 인터 디코더(880), 잔차 디코더(873), 재구축 모듈(874) 및 인트라 디코더(872)를 포함한다.

[0091] 엔트로피 디코더(871)는 코딩된 픽처로부터, 코딩된 픽처를 구성하는 선택스 요소를 나타내는 특정 심볼을 재구축하도록 구성될 수 있다. 이러한 심볼은, 예를 들어 블록이 코딩되는 모드(예: 인트라 모드, 인터 모드, 양-예측 모드, 두 후자는 병합 서브모드(merge submode) 또는 다른 서브모드) 및 인트라 디코더(872) 또는 인터 디코더(880) 각각에 의해 예측에 사용되는 특정 샘플 또는 메타 데이터를 식별할 수 있는 예측 정보(예: 인트라 예측 정보 또는 인터 예측 정보)를 포함할 수 있다. 심볼은 또한 예를 들어 양자화된 변환 계수 등의 형태로 잔차 정보 등을 포함할 수 있다. 일례에서, 예측 모드가 인터 모드 또는 양-예측 모드인 경우, 인터 예측 정보는 인터 디코더(880)에 제공되고; 예측 유형이 인트라 예측 유형인 경우, 인트라 예측 정보는 인트라 디코더(872)에 제공된다. 잔차 정보는 역양자화될 수 있으며 잔차 디코더(873)에 제공된다.

[0092] 인터 디코더(880)는 인터 예측 정보를 수신하고 인터 예측 정보에 기초하여 인터 예측 결과를 생성하도록 구성된다.

[0093] 인트라 디코더(872)는 인트라 예측 정보를 수신하고 인트라 예측 정보에 기초하여 예측 결과를 생성하도록 구성된다.

[0094] 잔차 디코더(873)는 역양자화를 수행하여 역양자화된 변환 계수를 추출하고 역양자화된 변환 계수를 처리하여 잔차 정보를 주파수 도메인에서 공간 도메인으로 변환하도록 구성된다. 잔차 디코더(873)는 또한 (양자화기 파라미터(Quantizer Parameter, QP)를 포함하기 위해) 특정 제어 정보를 필요로 할 수 있고, 그 정보는 엔트로피 디코더(871)에 의해 제공될 수 있다(데이터 경로는 소량의 제어 정보뿐일 수 있으므로 표시되지 않음).

[0095] 재구축 모듈(874)은 공간 도메인에서, 잔차 디코더(873)에 의해 출력되는 잔차 정보와 예측 결과(경우에 따라 인터 예측 모듈 또는 인트라 예측 모듈에 의해 출력됨)를 결합하여 재구축된 블록을 형성하도록 구성되고, 재구축된 블록은 재구축된 픽처의 일부일 수 있으며, 재구축된 픽처는 결과적으로 재구축된 비디오의 일부일 수 있

다. 디블로킹(deblocking) 동작 등과 같은, 다른 적절한 동작이 시각적 품질을 향상시키기 위해 수행될 수도 있다는 점에 유의한다.

[0096] 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 임의의 적절한 기술을 사용하여 구현될 수 있음에 유의한다. 일 실시예에서, 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 하나 이상의 접적 회로를 사용하여 구현될 수 있다. 다른 실시예에서, 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 소프트웨어 명령어를 실행하는 하나 이상의 프로세서를 사용하여 구현될 수 있다.

[0097] II. 인트라 블록 복사(IBC 또는 IntraBC) 모드

[0098] 본 개시는 로컬 참조 범위를 갖는 인트라 블록 복사 모드와 관련된 어드밴스드 비디오/이미지 코딩 기술을 설명한다.

[0099] II.1 HEVC 및 VVC에서의 IBC 모드

[0100] II.1.1 HEVC에서의 현재 픽처 참조(CPR)

[0101] IBC 코딩 도구는 CPR로서 HEVC 스크린 콘텐츠 코딩(Screen Content Coding, SCC) 확장에 사용될 수 있다. IBC 모드는 IBC 모드에서 현재 픽처가 참조 픽처로 사용되는 인터 예측에 사용되는 코딩 기술을 사용할 수 있다. IBC 모드 사용의 이점은 2차원(2D) 공간 벡터가 참조 샘플에 대한 주소지정 메커니즘의 표현으로 사용될 수 있는 IBC 모드의 참조 구조이다. IBC 모드 아키텍처의 이점은 IBC의 통합이 사양에 대한 비교적 사소한 변경을 필요로 하고 제조사가 HEVC 버전 1과 같은 특정 인터 예측 기술을 이미 구현한 경우 구현 부담을 완화할 수 있다는 것이다. HEVC SCC 확장에서의 CPR은 인터 예측 모드의 신택스 구조와 동일한 신택스 구조 및 인터 예측 모드의 디코딩 프로세스와 유사한 디코딩 프로세스를 초래하는 특별한 인터 예측 모드이다.

[0102] IBC 모드는 인터 예측 프로세스에 통합될 수 있다. 일부 예에서, IBC 모드(또는 CPR)는 인터 예측 모드이고, 인트라만의(intra-only) 예측된 슬라이스는 IBC 모드의 사용을 허용하기 위해 예측된 슬라이스가 될 것이다. IBC 모드가 적용 가용한 경우, 코더는 포인터가 현재 픽처를 가리키도록 하나의 엔트리만큼 참조 픽처 목록을 확장할 수 있다. 예를 들어, 현재 픽처는 공유된 디코딩된 픽처 버퍼(decoded picture buffer, DPB) 중 픽처 하나 크기의 버퍼를 사용한다. IBC 모드 시그널링은 암묵적일 수 있다. 예를 들어, 선택된 참조 픽처가 현재 픽처를 가리키는 경우, CU는 IBC 모드를 채용할 수 있다. 다양한 실시예에서, IBC 프로세스에서 사용되는 참조 샘플은 필터링되지 않으며, 이는 정규 인터 예측(regular inter prediction)과 다르다. IBC 프로세스에서 사용되는 대응하는 참조 픽처는 장기 참조 픽처이다. 메모리 요구사항을 최소화하기 위해, 코더는 현재 픽처를 재구축한 후 버퍼를 릴리스(release), 즉 사용할 수 있도록 제한을 풀 수 있다. 예를 들어 코더는 현재 픽처를 재구축한 후 버퍼를 즉시 릴리스한다. 재구축된 픽처의 필터링된 버전은 재구축된 픽처가 참조 픽처인 경우 코더에 의해 단기 참조로서 DPB에 다시 되돌려질 수 있다.

[0103] 블록 벡터(block vector, BV) 코딩에서, 재구축된 영역에 대한 참조는 인터 예측과 유사한 2D BV를 통해 수행될 수 있다. BV의 예측 및 코딩은 인터 예측 프로세스에서 MV 예측 및 코딩을 재사용할 수 있다. 일례에서, 루마 BV는 정규 인터 코딩된 CTU에 사용되는 MV의 1/4 정밀도가 아닌 정수 해상도이다.

[0104] 도 9는 본 개시의 일 실시예에 따른 현재 CU(901)와 연관된 BV를 도시한다. 각각의 정사각형(900)은 CTU를 나타낼 수 있다. 회색 음영 영역은 이미 코딩된 영역(예: 이미 인코딩된 영역)을 나타내고, 음영이 없는 흰색 영역은 코딩될 영역(예: 인코딩될 영역)을 나타낸다. 재구축 중인 현재 CTU(900(4))는 현재 CU(901), 코딩 영역(902) 및 코딩될 영역(903)을 포함한다. 일례에서, 영역(903)은 현재 CU(901)를 코딩한 후에 코딩될 것이다.

[0105] HEVC와 같은 예에서, 현재 CTU(900(4)) 바로 위에 있는 두 개의 CTU(900(1)~900(2))를 제외한 회색 음영 영역은 과면 병렬 처리(Wavefront Parallel Processing, WPP)를 허용하기 위해 IBC 모드에서 참조 영역으로 사용된다. HEVC에서 허용되는 BV는 참조 영역(예: 두 CTU(900(1)~900(2))를 제외한 회색 음영 영역) 내에 있는 블록을 가리킬 수 있다. 예를 들어, HEVC에서 허용되는 BV(905)는 참조 블록(911)을 가리킨다.

[0106] VVC에서와 같은 예에서, 현재 CTU(900(4))에 더하여, 현재 CTU(900(4))의 왼쪽에 있는 좌측 이웃 CTU(900(3))만이 IBC 모드에서 참조 영역으로서 허용된다. 일례에서, VVC에서 IBC 모드에서 사용되는 참조 영역은 접선 영역(915) 내에 있고 코딩된 샘플을 포함한다. 예를 들어, VVC에서 허용되는 BV(906)는 참조 블록(912)을 가리킨다.

[0107] 일부 예에서, BV의 디코딩된 움직임 벡터 차이(motion vector difference, MVD)(BV 차이(BV difference, BVD)

라고도 함)는 최종 BV를 재구축하기 위해 대응하는 BV 예측기에 더하기 전에 2만큼 좌측 시프트될 수 있다.

[0108] 일부 실시예에서, 구현 및 성능상의 이유로 IBC 모드의 특별한 처리가 필요할 수 있고, IBC 모드와 인터 예측 모드(예: 정규 인터 예측 모드)는 이하에서 설명되는 바와 같이 상이할 수 있다. 일례에서, IBC 모드에서 사용되는 참조 샘플은 필터링되지 않는다(예컨대, DBF 및 샘플 적응형 오프셋(sample adaptive offset, SAO) 필터와 같은 인루프 필터링 프로세스 이전에 재구축된 샘플이 적용됨). HEVC의 다른 인터 예측 모드(예: 정규 인터 예측 모드)는 필터링된 샘플, 예를 들어 루프 내 필터링 프로세스에 의해 필터링되는 참조 샘플을 사용할 수 있다.

[0109] 일부 예에서, 루마 샘플 보간은 IBC 모드에서 수행되지 않는다. 크로마 샘플 보간은 IBC 모드에서 수행될 수 있다. 일부 예에서, 크로마 샘플 보간은 크로마 BV가 대응하는 루마 BV로부터 도출될 때 크로마 BV가 정수가 아닌 경우에만 필요하다. 일부 예에서, 루마 샘플 보간 및 크로마 샘플 보간은 정규 인터 예측 모드에서 수행될 수 있다.

[0110] IBC 모드에서, 크로마 BV가 정수가 아닌 BV이고 참조 블록이 사용 가능한 영역(예: 참조 영역)의 경계 근처에 있는 경우에 특수한 경우가 발생할 수 있다. 예를 들어, 주변 재구축된 샘플은 크로마 보간을 수행하기 위해 경계 외부에 있을 수 있다. 일례에서, 경계선에 인접한 단일의 선을 가리키는 BV는 주변의 재구축된 샘플이 경계 외부에 있을 수 있다.

II.1.2 VVC에서의 IBC 아키텍처

[0112] HEVC SCC 확장에서 IBC 모드에 대한 유효 참조 영역은 도 9에 설명된 것과 같은 별별 처리 목적을 위한 일부 예외를 제외하고는 현재 뷍처의 이미 재구축된 전체 영역을 포함할 수 있다. HEVC에서 사용되는 참조 영역의 단점으로는 하드웨어 구현이 외부 메모리를 채용할 수 있는 DPB의 추가 메모리에 대한 요구사항을 포함할 수 있다. 외부 메모리에 대한 추가 액세스는 메모리 대역폭을 증가시킬 수 있으며, DPB 사용은 덜 매력적일 수 있다. 일부 실시예에서, VVC에서는 IBC 모드를 위한 온칩(on-chip)으로 구현될 수 있는 고정 메모리(fixed memory)(예: 고정된 크기를 갖는 메모리)가 사용될 수 있다. IBC 모드의 온칩 고정 메모리는 하드웨어 아키텍처에서 IBC 모드를 구현하는 복잡도를 크게 줄일 수 있다. 예를 들어, IBC 모드에서의 온칩 고정 메모리는 레이턴시(latency)를 감소시킬 수 있다. 일부 예에서, 수정은 HEVC SCC 확장에서와 같이 인터 예측 프로세스 내의 통합에서 벗어난 시그널링 개념을 다룬다.

[0113] 도 10a~도 10d에 도시된 예에서, 고정 메모리는 IBC 모드에서 사용되는 참조 영역을 저장하기 위해 할당될 수 있다. 고정 메모리는 참조 샘플 메모리(reference sample memory, RSM)이라고 지칭될 수 있다. RSM의 일부는 코딩 프로세스(예: 인코딩 프로세스 또는 재구축 프로세스) 동안 다른 중간 시간에 업데이트될 수 있다. 도 10a~도 10d는 본 개시의 실시예에 따른 코딩 프로세스(예: 인코딩 프로세스 또는 재구축 프로세스) 동안 다양한 중간 시간에서의 RSM 업데이트 프로세스를 도시한다. 도 10a~도 10dd는 VVC에서의 IBC 모드 및 VVC에서의 구성에 대한 참조 영역을 도시한다.

[0114] 도 10a~도 10d를 참조하면. 현재 CTU(1020)는 현재 CTU(1020)의 좌측에 있는 CTU(예: 좌측 이웃 CTU)(1010)에 인접한다. 일부 예에서, 현재 CTU(1020)는 4개의 영역(1021~1024)을 포함한다. 좌측 이웃 CTU(1010)는 영역(1021~1024)에 각각 대응하는 4개의 영역(1011~1014)을 포함할 수 있다. 영역(1011~1014)의 위치는 영역(1021~1024)의 위치에서 각각 CTU(1020)의 폭만큼 왼쪽으로 시프트된다. RSM은 현재 CTU(1020)의 일부 및/또는 좌측 이웃 CTU(1010)의 일부를 포함할 수 있다. 도 10a~도 10d에 도시된 예에서. RSM의 크기는 CTU의 크기와 동일하다. 밝은 회색 음영 영역은 좌측 이웃 CTU(1010)의 참조 샘플을 포함할 수 있고, 어두운 회색 음영 영역은 현재 CTU(1020)의 참조 샘플을 포함할 수 있으며, 음영이 없는 흰색 영역은 코딩될 영역(예: 다가오는 코딩 영역)을 나타낼 수 있다.

[0115] 도 10a를 참조하면, 현재 CTU(1020)의 코딩 프로세스의 시작인 코딩 프로세스의 제1 중간 시간에서, RSM은 전체 좌측 이웃 CTU(1010)를 포함하고, 전체 좌측 이웃 CTU(1010)가 현재 CTU(1020)의 코딩 프로세스 시작 시 IBC 모드에서 참조 영역의 역할을 할 수 있다. 현재 CTU(1020)의 코딩 프로세스 시작 시 RSM은 영역(1021~1024)를 포함하지 않는다.

[0116] 도 10b를 참조하면, 영역(1021)은 서브영역(1031~1033)을 포함한다. 서브영역(1031)은 이미 코딩되어 있고(예: 인코딩 또는 재구축됨), 서브영역(1032)은 코딩 중(예: 인코딩 중이거나 재축 중)인 현재 CU이고, 서브영역(1033)은 이후에 코딩될 것이다. 현재 CTU(1020)의 서브영역(1032)이 코딩되고 있는 현재 CTU(1020)의 코딩 프로세스의 제2 중간 시간에, RSM은 좌측 이웃 CTU(1010)의 일부 및 현재 CTU(1020)를 포함하도록 업데이

트된다. 예를 들어, RSM은 좌측 이웃 CTU(1010)의 영역(1012~1014)과 현재 CTU(1020)의 서브영역(1031)을 포함한다. 제2 중간 시간에서의 참조 영역은 좌측 이웃 CTU(1010)의 영역(1012~1014) 및 현재 CTU(1020)의 서브 영역(1031)을 포함할 수 있다.

[0117] 도 10c를 참조하면, 영역(1022)은 서브영역(1041~1043)을 포함한다. 서브영역(1041)(어두운 회색 음영)은 이미 코딩되어 있고(예: 인코딩 또는 재구축됨), 서브영역(1042)은 코딩 중(예: 인코딩 중이거나 재축 중)인 현재 CU이고, 서브영역(1043)(흰색)은 이후에 코딩될 것이다. 현재 CTU(1020)의 서브영역(1042)이 코딩되고 있는 현재 CTU(1020)의 코딩 프로세스의 제3 중간 시간에, RSM은 (i) 좌측 이웃 CTU(1010)의 영역(1013~1014) 및 (ii) 현재 CTU(1020)의 영역(1021) 및 서브영역(1041)을 포함하도록 업데이트된다. RSM에서, 영역(1012)은 서브영역(1041)으로 대체된다. 제3 중간 시간에서의 참조 영역은 (i) 좌측 이웃 CTU(1010)의 영역(1013~1014) 및 (ii) 현재 CTU의 영역(1021) 및 서브영역(1041)을 포함할 수 있다. CTU(1020).

[0118] 도 10d를 참조하면, 영역(1024)은 서브영역(1051~1053)을 포함한다. 서브영역(1051)(어두운 회색 음영)은 이미 코딩되어 있고(예: 인코딩 또는 재구축됨), 서브영역(1052)은 코딩 중(예: 인코딩 중이거나 재축 중)인 현재 CU이고, 서브영역(1053)(흰색)은 이후에 코딩될 것이다. 현재 CTU(1020)의 서브영역(1052)이 코딩되고 있는 현재 CTU(1020)의 코딩 프로세스의 제4 중간 시간에, RSM은 영역(1021~1023) 및 현재 CTU(1020)의 서브영역(1051)을 포함하도록 업데이트된다. 제4 중간 시간에서의 RSM은 좌측 이웃 CTU(1010)에 영역을 포함하지 않는다. 제4 중간 시간의 참조 영역은 현재 CTU(1020)의 영역(1021~1023) 및 서브영역(1051)을 포함할 수 있다.

II.1.3 VVC에서 IBC 모드의 선택스 및 시맨틱스

[0119] VVC에서 IBC 아키텍처는 인트라 예측 모드 및 인터 예측 모드(예: 정규 인터 예측 모드)에 더하여 IBC 모드가 세 번째 예측 모드인 전용 코딩 모드를 형성할 수 있다. 비트스트림은 예를 들어 CU의 크기가 64×64 이하인 경우 CU에 대한 IBC 모드를 지시하는 IBC 선택스를 포함할 수 있다. 일부 예에서, IBC 모드를 이용할 수 있는 최대 CU 크기는 도 10a~도 10d를 참조하여 설명한 바와 같이, RSM의 연속적인 메모리 업데이트 메커니즘을 구현하기 위해 64×64 이다. 일례에서, 참조 샘플 주소지정 메커니즘은 2D 오프셋을 표시하고 인터 예측 모드의 벡터(예: MV) 코딩 프로세스를 재사용함으로써 HEVC SCC 확장에서 사용되는 것과 동일하게 유지된다. 일례에서, CST가 활성화되는 경우, 코더는 대응하는 루마 BV로부터 크로마 BV를 도출할 수 없으므로, 그 결과 루마 CB에 대해서만 IBC 모드를 사용하게 된다.

II.1.4 VVC에서 IBC 모드의 참조 영역 및 샘플 메모리

[0120] VVC에서의 IBC 설계는 참조 샘플을 저장하기 위해 각 색상 성분에 대해 고정 메모리 크기(예: 128×128)를 채용할 수 있다. 전술한 바와 같이, 고정 메모리 크기는 하드웨어 구현에 있어 메모리(예: RSM)의 온칩 배치를 가능하게 할 수 있다. VVC와 같은 예에서, IBC 모드에 대한 최대 CTU 크기 및 고정 메모리 크기는 128×128 이다. 일례에서, RSM은 최대 CTU 크기 구성이 IBC 모드의 고정 메모리 크기(예: 128×128)와 동일한 경우 단일 CTU의 샘플을 포함한다.

[0121] RSM의 특징은 도 10a~도 10d에 기재된 바와 같이 좌측 이웃 CTU의 재구축된 샘플을 현재 CTU의 재구축된 샘플로 대체하는 연속 업데이트 메커니즘이다. 도 10a~도 10d는 코딩 프로세스(예: 재구축 프로세스) 동안 4개의 중간 시간에서의 업데이트 메커니즘에 대한 단순화된 RSM 예를 도시한다. 도 10a~도 10c에서 밝은 회색 음영 영역은 좌측 이웃 CTU(1010)의 참조 샘플을 포함하고, 도 10a~도 10d에서 어둔 회색 음영 영역은 현재 CTU(1020)의 참조 샘플을 포함할 수 있다. 도 10a를 참조하면, 현재 CTU(1020)의 코딩(예: 인코딩 또는 재구축)의 시작을 나타내는 제1 중간 시간에서, RSM은 좌측 이웃 CTU(1010)의 참조 샘플로만 구성된다. 도 10a~도 10d에 도시된 다른 3개의 중간 시간에서, 코딩 프로세스(예: 인코딩 프로세스 또는 재구축 프로세스)는 좌측 이웃 CTU(1010)의 샘플을 현재 CTU(1020)의 샘플로 대체되었다.

[0122] 일부 예에서, RSM은 암묵적으로 64×64 의 4개의 해체된 영역과 같은, 4개의 영역으로 나눈다. 코더가 현재 CTU에서 대응하는 영역에 있는 첫 번째 CU를 처리하는 경우 RSM에서 영역의 재설정이 발생할 수 있으므로, 하드웨어 구현 노력을 경감한다. 예를 들어, RSM은 CTU(들)(예: 좌측 이웃 CTU 및 현재 CTU)의 영역에 매핑된다. 도 11은 RSM의 연속적인 업데이트 프로세스(1100)을 공간적으로 도시한다. 좌측 이웃 CTU(1010) 및 현재 CTU(1020)는 도 10a~도 10d에 기재되어 있다. 좌측 이웃 CTU(1010)는 영역(1011~1014)을 포함할 수 있다. 현재 CTU(1020)는 영역(1021~1024)을 포함할 수 있다. 현재 CTU(1020) 내의 영역(1023)은 코딩 중인 현재 CU(1152), 이미 코딩된 서브영역(1151), 코딩될 서브영역(1153)을 포함한다. 회색 음영 영역은 RSM에 저장된 샘플을 포함할 수 있으며, 음영이 없는 흰색 영역은 대체된 샘플 또는 코딩되지 않은 샘플(예: 재구축되지 않은

샘플)을 포함할 수 있다.

[0125] 도 11에 도시된 코딩 시간(예: 재구축 시간)에서, RSM 업데이트 프로세스는 좌측 이웃 CTU(1010)에서 음영이 없는 흰색 영역(예: 영역(1011~1013))으로 덮인 샘플을 현재 CTU(1020)의 회색 음영 영역(예: 영역(1021~1022) 및 서브영역(1151))으로 대체하였다. 도 11에서, RSM은 (i) 좌측 이웃 CTU(1010) 내의 영역(1014) 및 (ii) 현재 CTU(1020)의 영역(1021~1022) 및 서브영역(1051)을 포함할 수 있다.

[0126] 일부 예에서, RSM은 최대 CTU 크기가 RSM 크기(예: 128×128)보다 작은 경우 하나 이상의 좌측 이웃 CTU를 포함할 수 있고, 다수의 이웃 CTU가 IBC 모드에서 참조 영역으로서 사용될 수 있다. 예를 들어, 최대 CTU 크기가 32×32인 경우, 128×128 크기의 RSM은 15개의 이웃 CTU 샘플을 포함할 수 있다.

II.1.5 VVC에서 IBC 모드의 BV 코딩

[0128] BV 코딩은 인터 예측(예: 정규 인터 예측)을 위해 지정된 프로세스를 채용할 수 있다. BV 코딩은 후보 목록을 구축하기 위해 인터 예측(예: 정규 인터 예측)에 사용되는 규칙보다 간단한 규칙을 사용할 수 있다.

[0129] 예를 들어, 인터 예측을 위한 후보 목록은 5개의 공간적 후보, 1개의 시간적 후보 및 6개의 이력 기반(history-based) 후보를 포함한다. 인터 예측을 위한 최종 후보 목록에서 중복 엔트리를 방지하기 위해 이력 기반 후보에 대해 여러 번의 후보 비교를 사용할 수 있다. 인터 예측을 위한 후보 목록은 쌍으로 평균화된 후보를 포함할 수 있다.

[0130] IBC 모드에 대한 후보 목록은 각각의 공간적 이웃으로부터의 2개의 BV 및 5개의 이력 기반 BV(HBVP)를 포함할 수 있다. 일례에서, IBC 모드에 대한 후보 목록은 각각의 공간적 이웃으로부터의 2개의 BV 및 5개의 이력 기반 BV(HBVP)로 제한된다. 일 실시예에서, IBC 모드에서, 첫 번째 HBVP가 후보 목록에 추가되는 경우 첫 번째 HBVP 만이 공간적 후보와 비교된다.

[0131] 정규 인터 예측 모드는 서로 다른 2개의 후보 목록, 예를 들어 병합 모드를 위한 하나의 후보 목록과 정규 모드(예: 병합 모드가 아닌 인터 예측 모드)를 위한 다른 하나의 후보 목록을 사용할 수 있다. IBC 모드의 후보 목록은 두 IBC 모드(예: 병합 IBC 모드 및 일반 IBC 모드)에 대해 동일할 수 있다. IBC 모드에서, 병합 모드는 후보 목록의 후보를 최대 6개까지 사용할 수 있으며 일반 모드는 후보 목록의 처음 두 후보만 사용한다.

[0132] 블록 벡터 차이(block vector difference, BVD) 코딩은 정규 인터 예측 모드에서 사용되는 MVD 프로세스를 채용할 수 있으며, 최종 BV는 임의의 크기를 가질 수 있다. 결정된 BV(예: 재구축된 BV)는 참조 샘플 영역 밖의 영역을 가리킬 수 있다. 일례에서, 각 방향에 대한 절대 오프셋에 대한 보정은 RSM의 너비 및/또는 높이에 기초한 모듈로 연산을 사용하여 적용될 수 있다.

II.2 AV1에서 IBC 모드

[0134] AV1과 같은 일부 예에서, IBC 모드(또는 IntraBC 모드)는 BV를 사용하여 현재 블록의 동일한 픽처에서 예측 블록을 찾는다. BV는 비트스트림으로 시그널링될 수 있으며 시그널링된 BV의 정밀도는 정수점(integer-point)일 수 있다. IBC 모드에서의 예측 프로세스는 인터 예측 모드에서의 예측 프로세스(예: 인터 픽처 예측)과 유사할 수 있다. IBC 모드와 인터 픽처 예측의 차이점은 이하에 설명된다. IBC 모드에서, 현재 픽처의 (예컨대, 루프 필터링을 적용하기 전) 재구축된 샘플로부터 예측자 블록이 형성될 수 있다. IBC 모드는 BV를 MV로 사용하는 현재 픽처 내에서 "움직임 보상"으로 간주될 수 있다.

[0135] 현재 블록에 대한 IBC 모드의 인에이블 여부를 지시시하는 플래그가 비트스트림에 송신될 수 있다. 현재 블록에 대해 IBC 모드가 인에이블되면, 현재 BV에서 예측 BV를 빼서 BV 차이를 도출할 수 있으며, BV 차이는 BV 차이의 값의 수평 성분과 수직 성분에 따라 4가지 유형으로 분류될 수 있다. 유형 정보는 비트스트림으로 시그널링될 수 있으며, 유형 정보 다음에 두 성분(예: 수평 성분 및 수직 성분)의 BV 차이 값이 시그널링될 수 있다.

[0136] IBC 모드는 스크린 콘텐츠를 코딩하는 데 효과적일 수 있다. IBC 모드는 하드웨어 설계에 문제를 야기할 수 있다. 하드웨어 설계를 용이하게 하기 위해 IBC 모드에서 다음 수정사항을 채택할 수 있다.

[0137] (i) IBC 모드가 허용되는 경우, 루프 필터는 디스에이블될 수 있다. 루프 필터는 디블록킹 필터, 제약된 방향성 강화 필터(constrained directional enhancement filter, CDEF), 및 루프 복원(loop restoration, LR) 필터를 포함할 수 있다. 루프 필터를 디스에이블함으로써, IBC 모드를 인에이블하기 위한 전용의 제2 픽처 버퍼를 그림 버퍼를 회피할 수 있다.

[0138] (ii) 병렬 디코딩을 용이하게 하기 위해, 예측은 제한된 영역을 초과할 수 없다. 슈퍼블록(superblock)의 좌측

상단 좌표는 (x_0, y_0) 이다. 슈퍼블록의 경우, 수직 좌표가 y_0 보다 작고 수평 좌표가 $(x_0 + 2(y_0 - y))$ 보다 작으면 위치 (x, y) 의 예측이 IBC 모드에 의해 액세스될 할 수 있다. 일례에서, 위치 (x, y) 에서의 예측은 수직 좌표가 y_0 보다 작고 수평 좌표가 $(x_0 + 2(y_0 - y))$ 보다 작은 경우에만 IBC 모드에 의해 액세스될 수 있다. 일례에서, 위치 (x, y) 에서의 예측은 수직 좌표가 y_0 보다 작거나 같고 수평 좌표가 $(x_0 + 2(y_0 - y))$ 보다 작은 경우에만 IBC 모드에 의해 액세스될 수 있다.

[0139] (iii) 하드웨어 되쓰기 지연(hardware writing back delay)을 허용하기 위해, 즉시 재구축된 영역은 IBC 모드에 의해 액세스될 수 없다. 제한된 즉시 재구축 영역은 1 내지 N개의 슈퍼블록을 포함할 수 있으며, N은 양의 정수이다. 전술한 수정 (ii)에 더하여, 재구축 중인 슈퍼블록(1210)의 좌측 상단 위치의 좌표가 (x_0, y_0) 이면, 위치 (x, y) 에서의 예측은 수직 좌표가 y_0 보다 작거나 같고 수평 좌표가 $(x_0 + 2(y_0 - y) - D)$ 보다 작으면 IBC 모드에 의해 액세스될 수 있다. D는 IBC 모드에 대해 제한되는 즉시 재구축된 영역(들)의 크기를 지시할 수 있다. 도 12는 제한되는 즉시 재구축된 영역의 일례를 도시한다. 회색 음영 영역은 재구축 중인 현재 슈퍼블록(1210) 각각에 대해 IBC 모드에서 액세스 가능한 허용된 검색 영역을 포함한다. 검은색 음영 영역은 각각의 현재 슈퍼블록(1210)에 대해 IBC 모드에서 액세스할 수 없는 허용되지 않은 검색 영역을 포함한다. 음영이 없는 흰색 영역은 코딩(예: 재구축)될 슈퍼블록을 포함한다. 현재 슈퍼블록(1210(1))에 대해, 즉시 재구축된 영역은 현재 슈퍼블록(1210(1))의 왼쪽에 있는 두 개의 슈퍼블록(1221~1222)을 포함한다(예: N은 2). D는 2W의 크기를 지시하며, 여기서 W는 각 슈퍼블록의 폭이다. 슈퍼블록(1221~1222)은 현재 슈퍼블록(1210(1))에 대해 액세스 불가능하다. 영역(1230)은 현재 슈퍼블록(1210(1))에 대해 액세스 가능하다.

[0140] II.3 AV1에서 로컬 참조 범위를 갖는 IBC 모드

[0141] $M \times M$ 크기(예를 들어, 128×128)의 온 칩 메모리(RSM이라 함)는 IBC 모드에서 사용되는 참조 샘플을 저장하기 위해 할당될 수 있다. 일 실시예에서, RSM의 크기는 슈퍼블록의 크기와 동일하며, 슈퍼블록의 크기는 RSM에 할당된다. 메모리 재사용 메커니즘은 $L \times L$ (예: 64×64) 단위로 RSM에 적용될 수 있다. RSM은 I RSM 단위로 나눌 수 있으며 여기서 I는 $L \times L$ 에 대한 $M \times M$ 의 비율과 같다. 예를 들어, $M \times M$ 이 128×128 이고 $L \times L$ 이 64×64 이면, I는 4($128 \times 128 / (64 \times 64)$)이다. 로컬 참조 범위는 II.2에서 설명한 IBC 모드에서 사용할 수 있다. II.2에서 설명한 IBC 모드에 대해 다음과 같은 변경이 가능하다.

[0142] (i) IBC 모드에서 최대 블록 크기는 $L \times L$ (예: 64×64)로 제한될 수 있다.

[0143] (ii) 현재 슈퍼블록(SB)에서 참조 블록 및 대응하는 현재 블록은 동일한 SB 행에 있을 수 있다. 일례에서, 참조 블록은 현재 SB 또는 현재 SB의 좌측에 있는 좌측 이웃 SB에만 위치한다.

[0144] (iii) RSM 유닛의 크기가 $L \times L$ (예: 64×64)인 유닛이 현재 SB의 재구축된 샘플로 업데이트를 시작하면, 이전에 저장된 참조 샘플(예: 좌측 이웃 SB의 참조 샘플)은 IBC 모드에서 사용되는 예측 샘플을 생성하기 위해 사용 불가능으로 표시될 수 있다.

[0145] 도 13은 본 개시의 일 실시예에 따라 현재 픽처에서 현재 SB(1301)의 코딩(예: 인코딩 또는 디코딩) 동안 메모리(예: RSM(1310))가 업데이트되는 예시적인 메모리 재사용 메커니즘(1300)을 도시한다. 상단 블록은 상태(0)의 RSM(1310)을 보여준다. 상단 행은 상태(1)~(4)의 RSM(1310)을 도시한다. 하단 행은 코딩되고 있는 현재 SB(1301) 및 상태(0)~(4)의 현재 픽처에서의 좌측 이웃 SB(1302)를 보여준다. 좌측 이웃 SB(1302)는 현재 SB(1301)의 좌측에 있을 수 있다. 도 13의 예에서, SB 루트에서 쿼드트리 분할(quad-tree split)이 사용되며 SB는 4개의 구역(region)을 포함할 수 있다. 일례에서, 4개의 구역 각각의 크기는 64×64 이다. 일례에서, 현재 SB(1301)는 4개의 구역 4~7을 포함하고, 좌측 이웃 SB(1302)는 4개의 구역 0~3을 포함한다.

[0146] 현재 SB(1301)와 같이, 각각 SB의 코딩이 시작되는 상태(0)에서, RSM(1310)은 이전에 코딩된 SB(예: 좌측 이웃 SB(1302))의 샘플을 저장할 수 있다. 현재 블록이 현재 SB(1301)에서 4개의 구역(예: 4개의 64×64 구역) 중 하나에 위치하는 경우, RSM(1310) 내의 대응하는 구역은 비워져 현재 코딩 구역(예: 현재 64×64 코딩 구역)의 샘플을 저장하는 데 사용될 수 있다. RSM(1310)의 샘플은 현재 SB(1301)의 샘플에 의해 점진적으로 업데이트될 수 있다.

[0147] 상태 (1)을 참조하면, 현재 블록(1311)은 현재 SB(1301)에서 구역 4에 위치하며, RSM(1310) 내의 대응하는 구역(예: 좌측 상단 구역)은 비워져 현재 코딩 중인 영역인 구역 4의 샘플을 저장하는 데 사용될 수 있다. 하단 행을 참조하면, BV(예: 인코딩된 BV 또는 디코딩된 BV)(1321)는 현재 블록(1311)에서 현재 블록(1311)에 대한 검색 범위(1341)(검색 범위(1341)의 경계는 파선으로 표시됨) 내에 있는 참조 블록(1331)을 가리킬 수 있다. 상단 행을 참조하면, RSM(1310)에서의 대응하는 오프셋(1351)은 현재 블록(1311)으로부터 RSM(1310)의 참조 블록

(1331)을 가리킬 수 있다. 상태 (1)을 참조하면, 검색 범위(1341)는 좌측 이웃 SB(1302)의 구역 1~3을 포함하고, 구역 4의 코딩된 서브구역(1361)을 포함한다. 검색 범위(1341)는 좌측 이웃 SB(1302)의 구역 0을 포함하지 않는다.

[0148] 상태 (2)를 참조하면, 현재 블록(1312)은 현재 SB(1301)에서 구역 5에 위치하며, RSM(1310)에서 대응하는 구역 (예: 우측 상단 구역)은 코딩 중인 현재 구역인 구역 5의 샘플을 저장하는 데 사용된다. BV(예: 인코딩된 BV 또는 디코딩된 BV)(1322)는 현재 블록(1312)으로부터 현재 블록에 대한 검색 범위(1342)(검색 범위(1342)의 경계는 파선으로 표시됨) 내에 있는 참조 블록(1332)을 가리킬 수 있다. RSM(1310)에서의 대응하는 오프셋(1352)은 현재 블록(1312)으로부터 RSM(1310) 내의 참조 블록(1332)을 가리킬 수 있다. 상태 (2)를 참조하면, 검색 범위(1342)는 (i) 좌측 이웃 SB(1302)의 구역 2~3 및 (ii) 현재 SB(1301)에 있는 구역 4, 및 구역 5 내의 코딩된 서브구역(1362)을 포함한다. 검색 범위(1342)는 좌측 이웃 SB(1302)의 구역 0~1을 포함하지 않는다.

[0149] 상태 (3)을 참조하면, 현재 블록(1313)은 현재 SB(1301)에서 구역 6에 위치하며, RSM(1310)에서 대응하는 구역 (예: 좌측 하단 구역)은 비워져 코딩 중인 현재 구역인 구역 6의 샘플을 저장하는 데 사용된다. BV(예컨대, 인코딩된 BV 또는 디코딩된 BV)(1323)는 현재 블록(1313)으로부터 현재 블록(1313)에 대한 검색 범위(1343)(검색 범위(1343)의 경계는 파선으로 표시됨) 내에 있는 참조 블록(1333)을 가리킬 수 있다. RSM(1310)에서의 대응하는 오프셋(1353)은 현재 블록(1313)으로부터 RSM(1310)에서의 참조 블록(1333)을 가리킬 수 있다. 상태 (3)을 참조하면, 검색 범위(1343)는 (i) 좌측 이웃 SB(1302)의 구역 3 및 (ii) 현재 SB(1301)에 있는 구역 4~5 및 구역 6의 코딩된 서브영역(1363)을 포함한다. 검색 범위(1343)는 좌측 이웃 SB(1302)의 구역 0~2를 포함하지 않는다.

[0150] 상태 (4)를 참조하면, 현재 블록(1314)은 현재 SB(1301)에서 구역 7에 위치하며, RSM(1310)에서 대응하는 구역 (예: 우측 하단 구역)은 비워져 코딩 중인 현재 구역인 구역 7의 샘플을 저장하는 데 사용된다. BV(예: 인코딩된 BV 또는 디코딩된 BV)(1324)는 현재 블록(1314)으로부터 현재 블록(1314)에 대한 검색 범위(1344)(검색 범위(1344)의 경계는 파선으로 표시됨) 내에 있는 참조 블록(1334)을 가리킬 수 있다. RSM(1310)에서 대응하는 오프셋(1354)은 현재 블록(1314)으로부터 RSM(1310)에서의 참조 블록(1334)을 가리킬 수 있다. 상태(4)를 참조하면, 검색 범위(1344)는 현재 SB(1301)에 있는 구역 4~6 및 구역 7 내의 코딩된 서브영역(1364)을 포함한다. 검색 범위(1344)는 좌측 이웃 SB(1302)의 구역 0~3을 포함하지 않는다.

[0151] 현재 SB(1301)의 코딩이 완료되었을 때, 전체 RSM(1310)은 현재 SB(1301)의 모든 샘플로 채워질 수 있다.

[0152] 도 13에 도시된 예에서, 현재 SB(1301)는 큐드트리 분할을 사용하여 파티셔닝된다. 현재 SB(1301)에서 4개 구역의 코딩 순서는 좌측 상단 구역(예: 구역 4), 우측 상단 구역(예: 구역 5), 좌측 하단 구역(예: 구역 6) 그리고 우측 하단 구역(예: 구역 7)일 수 있다. 도 14a 및 도 14b에 도시된 것과 같은 다른 블록 분할 결정에서, RSM 업데이트 프로세스는 예를 들어, RSM의 각 구역을 현재 SB에서의 재구축된 샘플을 사용하여 대체함으로써, 도 13에 도시된 것과 유사할 수 있다.

[0153] 도 14는 현재 SB(1401)의 코딩(예: 인코딩 또는 디코딩) 동안 RSM에서의 예시적인 메모리 업데이트 프로세스를 도시한다. 도 14에서, 좌측 이웃 SB(1402)는 코딩(예: 인코딩 또는 디코딩)되고 있는 현재 SB(1401)의 좌측에 있다. 일례에서, 현재 SB(1401)와 좌측 이웃 SB(1402)의 크기는 각각 128×128 이다. 현재 SB(1401) 및 좌측 이웃 SB(1402) 각각은 64×64 크기의 4개의 구역(예: 4개의 블록)을 포함할 수 있다. 현재 SB(1401)는 블록 4~7을 포함할 수 있고, 좌측 이웃 SB(1402)는 블록 0~3을 포함할 수 있다.

[0154] 도 14의 (a)에서, SB 루트에서 수평 분할이 수행되고 수직 분할이 뒤따른다. SB(예: 현재 SB(1401))는 4개의 블록, 즉 좌측 상단 블록(예: 블록 4), 좌측 하단 블록(예: 블록 6), 우측 상단 블록(예: 블록 5) 및 우측 하단 블록(예: 블록 7)을 포함할 수 있다. 현재 SB(1401)에 대한 코딩 순서는 좌측 상단 블록(상태 1), 우측 상단 블록(상태 2), 좌측 하단 블록(상태 3), 그리고 우측 하단 블록(상태 4)일 수 있다.

[0155] 도 14의 (b)에서, SB 루트에서 수직 분할이 수행되고 수평 분할이 뒤따른다. 현재 SB(1401)에 대한 코딩 순서는 좌측 상단 블록(상태 1), 좌측 하단 블록(상태 2), 우측 상단 블록(상태 3), 그리고 우측 하단 블록(상태 2)일 수 있다.

[0156] 현재 SB(1401)에 대한 현재 블록(예: (1431))의 위치에 따라 다음과 같이 적용될 수 있다.

[0157] (i) 도 14에서 상태 (1)을 참조하면, 현재 블록(1431)은 현재 SB(1401)의 좌측 상단 블록(예: 블록 4)에 있고, 블록 4에 있는 블록(1461) 내의 이미 재구축된 샘플에 더해 좌측 이웃 SB(1402)의 우측 하단 블록(예: 블록 3),

좌측 하단 블록(예: 블록 2) 및 우측 상단 블록(예: 블록 1) 내의 참조 샘플을 포함할 수 있다.

[0158] (ii) 도 14 (a)의 상태 (2) 또는 도 14 (b)의 상태(3)를 참조하면, 현재 블록(1432)은 현재 SB(1401)의 우측 상단 블록(예: 블록 5)에 있다.

[0159] 블록 6의 좌측 상단 코너(예: 현재 SB(1401)에 대해 (0, 64))에 위치한 루마 샘플이 도 14a의 상태(2)에 도시된 바와 같이 아직 재구축되지 않았으면, 블록 4, 및 블록 5에 있는 블록(1462) 내의 이미 재구축된 샘플에 더해, 현재 블록(1432)은 좌측 이웃 SB(1402)의 좌측 하단 블록(예: 블록 2) 및 우측 하단 블록(예: 블록 3) 내의 참조 샘플을 참조할 수 있다. 대응하는 RSM은 블록 4, 및 블록 5에 있는 블록(1462)에 더해 좌측 이웃 SB(1402)의 좌측 하단 블록(예: 블록 2) 및 우측 하단 블록(블록 3) 내의 참조 샘플을 포함할 수 있다.

[0160] 그렇지 않고, 블록 6의 좌측 상단 코너(예: 현재 SB(1401)에 대해 (0, 64))에 위치한 루마 샘플이 도 14 (b)의 상태(3)에 도시된 바와 같이 재구축되었으면, 현재 블록(1432)은 좌측 이웃 SB(1402)의 우측 하단 블록(예: 블록 3) 내의 참조 샘플을 참조할 수 있다. 대응하는 RSM은 블록 4, 블록 6, 및 블록 5에 있는 블록(1462) 내의 이미 재구축된 샘플에 더해 좌측 이웃 SB(1402)의 우측 하단 블록(예: 블록 3)의 참조 샘플을 포함할 수 있다.

[0161] (iii) 도 14 (a)에서 상태(3) 또는 도 14 (b)에서 상태(2)를 참조하면, 현재 블록(1433)은 현재 SB(1401)의 좌측 하단 블록(예: 블록 6)에 있다.

[0162] 블록 5의 좌측 상단 코너(예: 현재 SB(1401)에 대해 (64,0))에 위치한 루마 샘플이 도 14의 (b)에서 상태(2)에 도시된 바와 같이 아직 재구축되지 않았다면, 블록 4 및 현재 SB(1401)에 있는 블록(1463)의 이미 재구축된 샘플에 더하여, 현재 블록(1433)은 우측 상단 블록(예: 블록 1) 및 좌측 이웃 SB(1402)의 우측 하단 블록(예: 블록 3)을 참조할 수 있다. 대응하는 RSM은 블록 4 및 현재 SB(1401)에 있는 블록(1463)에 더하여 좌측 이웃 SB(1402)의 우측 상단 블록(예: 블록 1) 및 좌측 이웃 블록(1402)에 참조 샘플을 포함할 수 있다.

[0163] 그렇지 않고, 도 14 (a)의 상태 (3)에서와 같이, 블록 5의 좌측 상단 코너(예: 현재 SB(1401)에 대해 (64,0))에 위치한 루마 샘플이 재구축되었으면, 현재 블록(1433)은 좌측 이웃 SB(1402)의 우측 하단 블록(예: 블록 3)에서 참조 샘플을 참조할 수 있다. 대응하는 RSM은 블록 4~5 및 현재 SB(1401)에 있는 블록(1463)의 이미 재구축된 샘플들에 더하여, 좌측 이웃 SB(1402)의 우측 하단 블록(예: 블록 3)에 참조 샘플을 포함할 수 있다.

[0164] (iv) 도 14에서 상태(4)를 참조하면, 현재 블록(1434)은 현재 SB(1401)의 우측 하단 블록(예: 블록 7)에 있다. 현재 블록(1434)은 블록 4~6 및 블록(1464)에서 이미 재구축된 샘플과 같은, 현재 SB(1401)에서 이미 재구축된 샘플을 참조할 수 있다. 대응하는 RSM은 블록 4~6 및 블록(1464)에 참조 샘플을 포함할 수 있다. 일례에서, 현재 블록(1434)이 현재 SB(1401)의 우측 하단 블록에 속하면, 현재 블록은 현재 SB(1401)에서 이미 재구축된 샘플만을 참조할 수 있다.

III. 적응형 로컬 참조 범위가 있는 IBC 모드

[0165] IBC 모드의 성능은 IBC 모드에서 사용되는 참조 범위에 크게 의존할 수 있다. VVC의 IBC 모드 및 로컬 참조 범위가 있는 IBC 모드와 같은 관련 기술에서는 미리 정의된 메모리 업데이트 메커니즘이 RSM에서 사용될 수 있다. 일례에서, RSM에서의 메모리 업데이트는 도 13 및 도 14에 설명된 바와 같이, 현재 블록의 블록 위치 및 현재 CTU 또는 현재 SB의 파티셔닝에만 의존한다. 메모리 업데이트는 메모리 업데이트 프로세서 동안 예를 들어 현재 블록의 콘텐츠나 예측 정보에 의존하지 않는다. 일부 실시예에서, 텍스처 콘텐츠가 거의 없거나 전혀 없는 픽처의 영역에 대해, 따라서 참조 범위(예: 로컬 참조 범위)의 영역을 포함하고 픽처의 영역을 RSM에 저장하는 것은 IBC 모드의 예측 효율에 대해 제한된 이점을 가질 수 있다. 본 개시의 일 실시예에 따르면, IBC 모드에서 적응적 로컬 참조 범위와 같은 적응적 참조 범위가 사용될 수 있다.

[0166] 이하의 설명에서 블록이라는 용어는 변환 블록(transform block, TB), 코딩 블록(coding block, CB), 코딩 유닛(coding unit, CU), 픽처 내의 슈퍼블록, CTU 또는 CTB 등으로 해석될 수 있다. 블록 크기는 블록 너비, 블록 높이, 블록 너비와 블록 높이의 최대값, 블록 너비와 블록 높이의 최소값, 영역 크기(예: 블록 너비×블록 높이), 블록의 종횡비(예: 블록 너비/블록 높이, 또는 블록 높이/블록 너비) 등을 의미할 수 있다.

[0167] 일부 예에서, 메모리 업데이트 영역 단위의 예로서 64×64 크기가 사용된다. 메모리 업데이트 영역 단위는 임의의 블록 크기일 수 있으며 크기 64×64 로 제한되지 않는다. 메모리 업데이트 영역 유닛의 예는 도 13의 구역 0~7 및 도 14의 구역 0~7을 포함한다.

[0168] 본 발명의 일 실시예에 따르면, IBC 모드에서 사용되는 로컬 참조 범위의 업데이트 여부는 현재 픽처에서 복원 중인 슈퍼블록에 있는 복원 영역을 기준으로 결정된다. 슈퍼블록은 하나 이상의 CB를 포함할 수 있다. 로컬 참

조 범위를 제한할 수 있다. 일례에서, 로컬 참조 범위의 최대 크기는 도 13 및 도 14에 도시된 2개의 슈퍼블록의 크기와 같이, 다수의 슈퍼 블록의 크기로 제한된다. 일례에서, 로컬 참조 범위의 공간적 범위는 예를 들어, 다수의 슈퍼블록(예: 도 13 및 도 14에 도시된 바와 같이, 슈퍼 블록 및 슈퍼 블록의 좌측 이웃 슈퍼블록) 내에 있도록 제한된다. 일부 실시예에서, 이전 로컬 참조 범위를 업데이트할지는 (i) 재구축된 영역의 콘텐츠 및/또는 (ii) 재구축된 영역의 예측 정보에 기초하여 결정된다.

[0170] 예를 들어, 메모리 업데이트 영역 단위의 크기가 재구축된 CB의 크기보다 작은 경우, 재구축된 영역의 크기는 메모리 업데이트 영역 단위의 크기(예: 64×64)와 동일할 수 있다. 예를 들어, 재구축된 CB의 크기가 메모리 업데이트 영역 단위의 크기보다 작거나 같은 경우, 재구축된 영역은 재구축된 CB일 수 있다.

[0171] 재구축된 영역의 콘텐츠는 텍스처 콘텐츠 및/또는 재구축된 영역의 스크린 콘텐츠를 포함할 수 있다. 재구축된 영역의 콘텐츠는 재구축된 영역의 재구축된 샘플(또는 재구축된 픽셀)에 기초하여 결정될 수 있다.

[0172] 재구축된 영역의 예측 정보는 영역을 재구축하는 데 사용된 예측 모드(들)을 지시할 수 있다. 일 실시예에서, 이전 로컬 참조 범위를 업데이트할지는 예측 모드(들)가 미리 정의된 예측 모드 세트에 있는지 여부에 기초하여 결정된다. 예측 모드(들)가 미리 정의된 예측 모드 세트에 있으면, 이전 로컬 참조 범위는 업데이트될 수 있다. 예측 모드가 미리 정의된 예측 모드 세트에 없으면, 이전 로컬 참조 범위는 업데이트되지 않는다. 일 실시예에서, 미리 정의된 예측 모드 세트는 IBC 모드, 팔레트 모드, 변환 스킁 모드, 블록 기반 델타 풀러스 코드 변조(block based delta pulse code modulation, BDPCM) 모드 등과 같은 스크린 콘텐츠 코딩 모드(들)를 포함한다.

[0173] 일 실시예에서, 예측 정보는, 예를 들어, 영역의 예측 정보에 포함된 플래그를 통해, 이전 로컬 참조 범위를 재구축된 영역에서 재구축된 샘플로 업데이트할지를 지시한다. 따라서, 이전 로컬 참조 범위를 업데이트할지는 예측 정보에서 재구축된 영역의 재구축된 샘플로 이전 로컬 참조 범위를 업데이트할지의 지시(예: 플래그)에 기초할 수 있다.

[0174] 일 실시예에서, IBC 모드의 RSM에서의 메모리 업데이트 프로세스는 현재 블록 또는 영역(예: 64×64 구역)의 콘텐츠(예: 재구축된 샘플의 값) 및/또는 현재 블록 또는 영역의 예측 정보(또는 코딩된 정보)에 의존한다.

[0175] 일례에서, 현재 블록(예: 현재 CB) 또는 영역(예: 64×64 구역)이 재구축된 후, 현재 블록 또는 영역의 콘텐츠(예: 재구축된 샘플의 값)의 평가가 수행된다. 메모리 업데이트의 수행 여부는 평가에 기초하여 결정될 수 있다. 도 15는 현재 SB(1501)가 디코딩되고 있을 때 RSM에서의 예시적인 적응적 메모리 업데이트 프로세스를 도시한다. 현재 SB(1501) 및 현재 SB(1501)의 좌측 이웃 SB(1502) 각각은 4개의 구역을 포함한다. 현재 SB(1501)는 구역 4~7을 포함하고, 좌측 이웃 SB(1502)는 구역 0~3을 포함한다. 일례에서, 구역 0~7 각각은 64×64 의 크기를 갖는다. 구역 0~3을 포함하는 좌측 SB(1502)는 재구축되었다. IBC 모드에서 사용된 이전 로컬 참조 범위는 RSM에 저장되어 있다. 일례에서, 이전 로컬 참조 범위는 구역 1~3을 포함한다. 도 15에 도시된 예에서, 구역 5의 로컬 참조 범위는 구역 0을 포함하지 않는다.

[0176] 현재 SB(1501)의 구역 4는 예를 들어 좌측 SB(1502)의 재구축 후에 재구축되었다. 이전 로컬 참조 범위를 저장하는 RSM을 업데이트하기 위해 메모리 업데이트를 수행할지를 판정하기 위해 구역 4의 콘텐츠(예: 재구축된 샘플)가 평가될 수 있다. 결정이 RSM을 업데이트하지 않는 것이면, RSM은 업데이트되지 않고 이전 로컬 참조 범위는 업데이트되지 않는다. 이전 로컬 참조 범위는 구역 5(예: 재구축 중인 현재 구역)를 재구축하는 데 사용될 수 있다. 일례에서, 이전 로컬 참조 범위는 구역 1~3을 포함하고 구역 5의 재구축 이전에 이미 재구축된 구역 4는 포함하지 않는다. 따라서, 구역 5의 검색 영역은 구역 1~3을 포함하고 구역 4는 포함하지 않는다. 일례에서, 구역 5의 검색 영역은 재구축 중인 구역 5의 이미 재구축된 부분을 포함한다.

[0177] 일 실시예에서, 평가 프로세스는 재구축된 블록 또는 재구축된 영역(예: 64×64 구역)의 텍스처 콘텐츠가 조건을 충족하는지를 판정한다. 도 15를 참조하면, 재구축된 구역 4의 텍스처 콘텐츠가 평가될 수 있다. 예를 들어, 재구축된 블록 또는 재구축된 영역의 텍스처 콘텐츠의 백분율이 결정된다. 재구축된 블록 또는 재구축된 영역의 텍스처 콘텐츠의 백분율이 임계값보다 크거나 같으면, 조건이 충족되는 것으로 결정된다. 그렇지 않으면, 조건은 충족되지 않는 것으로 결정된다.

[0178] 재구축된 블록 또는 재구축된 영역의 콘텐츠는 하나 또는 다수의 색상 성분에서 강도 값의 히스토그램을 기반으로 하는 것과 같은 임의의 적합한 방법(들)에 의해 결정될 수 있다. 일례에서, 히스토그램이 하나 또는 다수의 색상 성분에 N0 초과 N1 미만의 개별 강도 값을 포함하면, 재구축된 블록 또는 재구축된 영역은 스크린 콘텐츠 영역으로서 결정된다. 그렇지 않으면 재구축된 블록 또는 재구축된 영역은 텍스처 영역으로 결정된다. 텍스처 영역은 카메라(들)로 캡처될 수 있으며, 자연 영역 또는 카메라 캡처 영역(camera captured area)이라고도

한다. 일례에서, 스크린 콘텐츠 영역은 카메라에 의해 캡처되지 않는다. 도 15를 참조하면, 구역 4와 연관된 하나 또는 다수의 색상 성분의 강도 값의 히스토그램은 구역 4가 텍스쳐 영역인지 스크린 콘텐츠 영역인지를 판정하는 데 사용될 수 있다.

[0179] 일 실시예에서, 평가 프로세스는 재구축된 블록 또는 재구축된 영역(예: 64×64 구역)이 미리 정의된 예측 모드 세트의 예측 모드(들)에 의해 코딩되는지를 판정한다. 일례에서, 미리 정의된 예측 모드 세트는 IBC 모드, 팔레트 모드, 변환 스kip 모드, BDPCM 모드, 등과 같은 스크린 콘텐츠 코딩 모드(들)를 포함한다.

[0180] 일 실시예에서, 평가 프로세스가 재구축된 블록 또는 재구축된 영역(예: 64×64 구역)의 스크린 콘텐츠가 조건을 충족하는 것으로 결정하면, 예를 들어, 스크린 콘텐츠의 양 또는 백분율은 임계값보다 작으면, 메모리 업데이트 프로세스는 재구축된 블록 또는 재구축된 영역에 대해 생략되며(스킵), 따라서 블록 또는 영역을 코딩(예: 인코딩 또는 디코딩)한 후 수행되지 않는다. 메모리 업데이트 프로세스가 생략되면, RSM에 저장된 이전 로컬 참조 범위가 재구축될 다른 코딩 블록 또는 다른 영역을 디코딩하는 데 사용된다. 일례에서, 재구축된 블록 또는 재구축된 영역은 어떠한 스크린 콘텐츠도 포함하지 않고, 메모리 업데이트 프로세스는 재구축된 블록 또는 재구축된 영역에 대해 생략되므로, 블록 또는 영역을 코딩한 후에 수행되지 않는다.

[0181] 도 15에 도시된 예에서, 구역 0~7의 각 구역의 크기는 메모리 업데이트 영역 단위의 크기와 동일하다. RSM의 크기는 슈퍼블록의 크기와 동일하다. 도 15를 참조하여 설명된 실시예는 슈퍼블록이 임의의 적절한 크기를 갖는 임의의 적절한 수의 구역을 포함할 수 있는 다른 상황에 적합하게 적용될 수 있다. RSM의 크기는 슈퍼블록의 크기와 같거나 크거나 작을 수 있다.

[0182] 도 15에 도시된 예에서, 평가 프로세스는 메모리 업데이트 영역 단위와 동일한 크기를 갖는 구역(예: 구역 4)에 기초한다. 따라서, 그 구역을 재구축한 후, 평가 프로세스가 이전 로컬 참조 범위가 업데이트되어야 함을 지시하면, 전체 구역의 재구축된 샘플을 사용하여 이전 로컬 참조 범위를 업데이트한다.

[0183] 일부 예에서, 평가 프로세스는 메모리 업데이트 영역 유닛의 크기와 동일한 크기를 갖는 구역의 일부에 기초한다. 예를 들어, CB의 크기는 메모리 업데이트 영역 단위의 크기보다 작고, 구역의 일부가 CB이다. 구역(예: C B)의 일부를 재구축한 후, 평가 프로세스에서 이전 로컬 참조 범위가 업데이트되어야 함을 지시하면, 구역(예: CB)의 일부에서 재구축된 샘플을 사용하여 이전 로컬 참조 범위를 업데이트한다. 일례에서, 메모리 업데이트 영역 단위의 크기를 갖는 구역에 대해, 그 구역의 제1 부분은 이전 참조 범위를 업데이트하는 데 사용될 수 있고 그 구역의 제2 부분은 이전 참조 범위를 업데이트하는 데 사용되지 않는다. 따라서, 로컬 참조 범위는 구역의 제1 부분의 제1 재구축된 샘플을 포함할 수 있고 구역의 제2 부분의 제2 재구축된 샘플을 포함하지 않는다. 구역의 제2 부분의 제2 재구축된 샘플의 위치(들)가 BV에 의해 참조되면, 도 16에 도시된 바와 같이, IBC 모드에서 사용될 샘플(들)을 생성하기 위해 패딩 프로세스(padding process)가 사용될 수 있다.

[0184] 도 16은 적응적 메모리 업데이트 프로세스에서 사용되는 예시적인 패딩 프로세스를 도시한다. 현재 SB(1601) 및 현재 SB(1601)의 좌측 이웃 SB(1602) 각각은 4개의 구역을 포함한다. 현재 SB(1601)는 구역 4~7을 포함하고, 좌측 이웃 SB(1602)는 구역 0~3을 포함한다. 일례에서, 구역 0~7 각각은 64×64 의 크기를 갖는다. 현재 SB(1601)의 구역 0~3 및 구역 4~5를 포함하는 좌측 SB(1602)는 이미 재구축되어 있다.

[0185] 구역 4는 제1 부분(1611) 및 제2 부분(1612)을 포함한다. 일례에서, 제1 부분(1611)은 제1 CB이고 제2 부분(1612)은 제2 CB이다. 일례에서, IBC 모드에서 사용되는 로컬 참조 범위가 제1 CB(1611)를 포함하도록 업데이트되는 메모리 업데이트가 수행된다. IBC 모드에서 사용되는 로컬 참조 범위가 제2 CB(1612)를 포함하도록 업데이트되지 않는 제2 CB(1612)를 재구축한 후에는 메모리 업데이트가 수행되지 않는다. 일례에서, 구역 5의 재구축 후, 로컬 참조 범위는 구역 3, 구역 4의 제1 부분(1611) 및 구역 5를 포함한다. 도 16에 도시된 예에서, 로컬 참조 범위는 좌측 이웃 SB(1602)에 구역 0~2를 포함하지 않는다. 로컬 참조 범위는 구역 4에 제2 부분(1612)을 포함하지 않는다. 구역 4의 제2 부분(1612)은 RSM에 저장되지 않는다.

[0186] 일 실시예에서, RSM에 저장된 로컬 참조 범위를 사용하여 IBC 예측을 수행하는 경우, 구역(예: 도 16의 구역 4)의 모든 재구축된 샘플이 RSM에 저장되는 것이 아니면 RSM에서 구역(예: 도 16의 구역 4)을 채우기 위해 패딩 프로세스가 적용될 수 있다. 도 16을 참조하면, 구역 6의 CB(1632)는 IBC 모드로 재구축된다. 구역 4의 제2 부분(1612)의 샘플은 CB(1632)에 대한 IBC 모드에서 참조 샘플로 사용되기 전에 패딩 프로세스에 의해 생성될 수 있다.

[0187] 일례에서, RSM에 저장된 로컬 참조 범위의 구역 4에서 제2 부분(1612)에 단일 값이 할당된다. 예를 들어, 제2 부분(1612)의 각 샘플은 단일 값과 같은 동일한 값을 갖는다. 일례에서, 단일 값은 제2 부분(1612)에 사용 가능

하고 이웃하는 샘플(들)에 기초한다. 제2 부분(1612)에 사용 가능하고 이웃하는 샘플(들)은 구역 4 내에 또는 구역 4 외부에 위치할 수 있다. 일례에서, 단일 값은 제2 부분(1612)에 이웃하는 샘플(들) 중 하나의 값과 같다.

- [0188] 일례에서, 구역 4의 제2 부분(1612)에 사용 가능하고 이웃하는 샘플(들)을 사용하여 RSM에서 구역 4의 제2 부분(1612)에 특정 인트라 예측, 예를 들어 DC 예측, 평면 예측 등을 수행함으로써 패딩 프로세스가 구현된다.
- [0189] 일례에서, 패딩 프로세스는 구역 4의 제2 부분(1612)에 사용 가능하고 이웃하는 샘플(들)을 복사함으로써 구현된다.
- [0190] 일부 예에서, 패딩 프로세스는 구역 4 외부의 경계 영역에 대해 수행된다.
- [0191] 일 실시예에서, RSM에서 메모리 업데이트 프로세스를 수행하는 경우, RSM에 저장된 블록(또는 저장된 구역)이 순위가 매겨질 수 있다. 메모리 업데이트 프로세스 동안 RSM에서 저장된 블록 중 어느 블록이 대체될지는 순위에 기초하여 결정될 수 있다. 일례에서, 저장된 블록 각각은 메모리 업데이트 영역 단위의 크기(예: 64×64)를 갖는다.
- [0192] 도 17을 참조하면, 현재 SB(1701) 및 현재 SB(1701)의 좌측 이웃 SB(1702)는 각각 4개의 구역을 포함한다. 현재 SB(1701)는 구역 4~7을 포함하고, 좌측 이웃 SB(1702)는 구역 0~3을 포함한다. 일례에서, 구역 0~7 각각은 64×64 의 크기를 갖는다. 구역 0~3을 포함하는 좌측 SB(1702)는 이미 재구축되었다.
- [0193] 도 17의 왼쪽을 참조하면, 구역 4는 재구축 중이고, 구역 5~7은 아직 재구축되지 않았다. IBC 모드에서 사용되는 로컬 참조 범위는 RSM에 저장되어 있다. 일례에서, 로컬 참조 범위는 구역 1~3을 포함하고 구역 0을 포함하지 않는다.
- [0194] 도 17의 오른쪽을 참조하면, 구역 4를 재구축한 후, 로컬 참조 범위는 재구축된 구역 4에 기초하여 업데이트되어야 한다. 일례에서, RSM에 저장된 구역 1~3에 순위가 매겨진다. 재구축된 구역 4에 의한 메모리 업데이트 프로세스 동안 구역 1~3 중 어느 구역이 대체되는지는 순위에 기초하여 결정될 수 있다. 예를 들어, 순위가 구역 2가 대체되어야 함을 지시하면, 로컬 참조 범위는 구역 2를 재구축된 구역 4로 대체함으로써 업데이트한다. 도 17의 오른쪽의 업데이트된 로컬 참조 범위는 구역 1, 3 및 4를 포함한다.
- [0195] 일례에서, 스크린 콘텐츠에 대응하는 재구축된 샘플의 백분율은 RSM에 저장된 블록(예: 도 17의 구역 1~3) 각각에 대해 결정된다. 각각의 저장된 블록의 스크린 콘텐츠의 백분율 순위에 기초하여, RSM에 저장된 저장된 블록 중 어느 것이 메모리 업데이트 프로세스 동안 대체되는지가 결정된다.
- [0196] 일례에서, 텍스처 영역에 대응하는 재구축된 샘플의 백분율은 RSM에 저장된 블록(예: 도 17의 구역 1~3) 각각에 대해 결정된다. 각각의 저장 블록의 텍스처 영역 백분율의 순위에 기초하여, RSM에 저장되어 있는 저장된 블록 중 어느 것이 메모리 업데이트 프로세스 동안 대체되는지가 결정된다.
- [0197] 일례에서, RSM에 저장된 블록들 각각(예: 도 17의 구역 1~3)과 연관된 빈도(예: 참조 빈도)가 결정된다. 일례에서, 참조 빈도는 현재 픽처 또는 현재 픽처의 일부에서 CB를 예측하기 위해 각각의 저장된 블록이 IBC 모드에서 BV에 의해 참조되는 횟수를 지시한다. 각각의 저장 블록에 대한 참조 빈도의 순위에 기초하여, RSM에 저장되어 있는 저장된 블록 중 어느 것이 메모리 업데이트 프로세스 동안 대체되는지가 결정된다.
- [0198] 일례에서, 저장된 블록들 각각(예: 도 17의 구역 1~3)의 특별히 코딩된 샘플의 수(예: 구역 크기)가 결정되며, 여기서 특별히 코딩된 샘플은 미리 정의된 예측 모드 세트 중의 하나 이상의 예측 모드로 코딩된다. 일례에서, 미리 정의된 예측 모드 세트는 위에서 설명된 바와 같이, IBC 모드, 팔레트 모드, 변환 스윕 모드, BDPCM 모드 등과 같은 스크린 콘텐츠 코딩 모드(들)를 포함한다. 각각의 저장 블록의 특별히 코딩된 샘플의 수의 순위에 기초하여, RSM에 저장된 블록 중 어느 것이 메모리 업데이트 프로세스 동안 대체되는지 결정된다.
- [0199] 재구축된 영역 또는 재구축된 블록의 콘텐츠 또는 예측 모드(들)에 기초하여 로컬 참조 범위를 업데이트할지를 판정하는 것에 더하여 또는 대신에, 로컬 참조 범위를 업데이트할지 여부는 로컬 참조 범위의 업데이트 여부를 지시하는 재구축된 영역 또는 재구축된 블록의 예측 정보에 기초하여 판정될 수 있다. 예를 들어, 예측 정보는 로컬 참조 범위의 업데이트 여부를 명시적으로 지시하는 재구축된 영역 또는 재구축된 블록의 플래그(예: 지시 플래그)를 포함하고, 플래그에 기초하여 로컬 참조 범위의 업데이트 여부를 판정한다. 일부 예에서, 로컬 참조 범위의 업데이트 여부는 재구축된 영역 또는 재구축된 블록의 플래그, 콘텐츠 및/또는 예측 모드(들)에 기초하여 판정된다.

- [0200] 일 실시예에서, 적응적 메모리 업데이트 프로세스는 IBC 모드의 RSM에서 메모리 업데이트 프로세스의 수행 여부가 각각의 메모리 업데이트 영역 단위에 대해 비트스트림으로 전송되는 지시(예: 플래그)에 의존하는 것을 제외하고는 도 15~도 17에 기술된 실시예와 유사하다.
- [0201] 일 실시예에서, 각각의 코딩된 영역에 대해(예: 코딩된 영역의 크기는 메모리 업데이트 영역 단위의 크기와 동일함), 코딩된 영역의 재구축된 샘플이 RSM에 저장되어야 하는지를 지시하기 위해 플래그가 시그널링된다. 플래그가 코딩된 영역의 재구축된 샘플이 RSM에 저장되어야 함을 지시하면, 본 개시에 설명된 바와 같이 메모리 업데이트 프로세스가 수행된다. 그렇지 않고, 플래그가 코딩된 영역의 재구축된 샘플이 RSM에 저장되지 않음을 지시하면, 코딩된 영역의 재구축된 샘플은 RSM에 저장될 필요가 없다. 예를 들어 메모리 업데이트 프로세스는 생략된다.
- [0202] 일 실시예에서, 현재 블록 크기(예: CB의 크기)가 메모리 업데이트 영역 단위의 크기보다 큰 경우, 현재 블록(예: CB)은 다수의 구역 각각이 메모리 업데이트 영역 단위의 크기를 가진 다수의 구역을 포함할 수 있다. 현재 블록의 제1 구역의 제1 플래그가 시그널링될 수 있고, 현재 블록의 다른 영역의 후속 플래그(들)은 추론되거나 시그널링될 수 있다. 일례에서, 후속 플래그(들)는 현재 블록의 제1 영역의 제1 플래그로부터 추론될 수 있다. 예를 들어, 후속 플래그(들)는 제1 영역의 제1 플래그와 동일한 값을 사용한다. 일례에서, 후속 플래그(들)는 시그널링되며 여기서 각각의 플래그는 다수의 구역 각각에 대해 개별적으로 시그널링된다. 각각의 영역의 샘플이 RSM에 저장되어야 하는지는 비트스트림에서 시그널링되는 각각의 플래그에 기초하여 판정될 수 있다. 따라서 현재 블록에 대해 다수의 플래그가 시그널링된다.
- [0203] 일 실시예에서, 고정된 크기의 구역에 대한 플래그는 고정된 크기의 구역 내의 샘플이 RSM에 저장되어야 하는지를 지시하기 위해 시그널링된다. 고정 크기의 구역은 메모리 업데이트 영역 단위보다 클 수 있다. 예를 들어, 고정된 크기의 구역은 메모리 업데이트 영역 단위의 크기가 예를 들어 64×64 인 128×128 크기의 슈퍼 블록, CTU, CTB이다. 플래그는 고정된 크기의 구역(예: 슈퍼블록, CTU 또는 CTB) 내의 임의의 샘플(들)이 RSM에 저장되어야 하는지를 지시하는 데 사용될 수 있다.
- [0204] 본 개시의 일부 실시예에서, 슈퍼블록(들)은 도 15~도 17을 참조한 설명에서와 같이 예로서 사용된다. 설명은 CTU(들) 또는 CTB(들)에 적용 가능하며, 슈퍼 블록(들)을 CTU(들) 또는 CTB(들)로 대체함으로써 적용될 수 있다.
- [0205] 도 18은 본 개시의 실시예에 따른 인코딩 프로세스(1800)를 개략적으로 설명하는 흐름도를 도시한다. 다양한 실시예에서, 인코딩 프로세스(1800)는 단말 기기(310, 320, 330, 340)의 처리 회로, 비디오 인코더(예: (403), (603), (703))의 기능을 수행하는 처리 회로, 등과 같은 처리 회로에 의해 실행된다. 일부 실시예에서, 인코딩 프로세스(1800)는 소프트웨어 명령어로 구현되며, 따라서 처리 회로가 소프트웨어 명령어를 실행할 때, 처리 회로는 인코딩 프로세스(1800)를 수행한다. 프로세스는 (S1801)에서 시작하여 (S1810)으로 진행한다.
- [0206] (S1810)에서, 현재 꽉쳐에서 슈퍼블록 또는 코딩 트리 블록(coding tree block, CTB) 내의 영역이 그 영역의 예측 정보에 기초하여 인코딩될 수 있다. 영역의 예측 정보는 영역의 하나 이상의 예측 모드를 지시할 수 있다. 영역은 영역의 하나 이상의 예측 모드에 기초하여 인코딩될 수 있다. 일례에서, 영역은 슈퍼블록 또는 CTB 내의 CB이다. 일례에서, 영역은 CB의 일부이다. 일례에서, 슈퍼블록 또는 CTB의 크기는 128×128 이다. 일례에서, 영역의 크기는 64×64 와 같이, 메모리 업데이트 영역 단위의 크기이다.
- [0207] (S1820)에서, (i) 영역의 콘텐츠 및/또는 (ii) 예측 정보에 기초하여 인트라 블록 복사(IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정할 수 있다. 일례에서, 영역의 콘텐츠는 샘플이 공간 도메인에서 재구축된 샘플인 대응하는 재구축된 영역의 샘플에 의해 지시된다.
- [0208] 일례에서, 영역의 콘텐츠는 영역에서의 스크린 콘텐츠의 백분율 및/또는 텍스처 콘텐츠의 백분율을 지시한다. IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지는 그 영역에서의 스크린 콘텐츠의 백분율 및/또는 텍스처 콘텐츠의 백분율에 기초하여 결정된다. 예를 들어, 영역의 콘텐츠는 스크린 콘텐츠를 포함하지 않으며, IBC 모드에 대한 이전 로컬 참조 범위는 업데이트되지 않는 것으로 결정된다.
- [0209] 일례에서, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지는 하나 이상의 예측 모드가 미리 정의된 예측 모드 세트에 있는지에 기초하여 결정된다.
- [0210] IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 결정한 것에 대한 응답으로, 인코딩 프로세스(1800)는 (S1830)로 진행한다. 그렇지 않은 경우, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하기로 결정

한 것에 대한 응답으로, 인코딩 프로세스(1800)는 (S1840) 및 (S1850)을 포함하는 분기로 진행한다.

[0211] (S1830)에서, 현재 픽처의 코딩 블록(CB)은 그 영역의 샘플을 포함하지 않는 이전 로컬 참조 범위에 기초하는 IBC 모드를 사용하여 인코딩될 수 있다. (S1830) 후, 인코딩 프로세스(1800)는 (S1899)로 진행하여 끝난된다.

[0212] (S1840) 및 (S1850)을 포함하는 분기에서, IBC 모드에 대한 이전 로컬 참조 범위가 업데이트되도록 결정된다.

[0213] (S1840)에서, 현재 로컬 참조 범위는 이전 로컬 참조 범위와 대응하는 재구축된 영역 내의 샘플(예: 재구축된 샘플)에 기초하여 생성될 수 있다.

[0214] 이전 로컬 참조 범위는 참조 샘플 메모리(RSM)에 저장된 다수의 영역을 포함할 수 있다. 현재 로컬 참조 범위는 이전 로컬 참조 범위의 다수의 영역 중 하나 내의 참조 샘플을 그 영역 내의 샘플로 대체하여 생성할 수 있다.

[0215] 일부 실시예에서, 이전 로컬 참조 범위의 다수의 영역 중 대체될 영역은 다수의 영역의 순위에 기초하여 결정된다. 따라서, 다수의 영역의 순위에 기초하여 복수의 영역 중에서 그 영역 내의 샘플로 대체될 영역이 선택될 수 있다. 현재 로컬 참조 범위는 이전 로컬 참조 범위 내의 선택된 영역을 해당 영역의 샘플로 대체하여 생성할 수 있다.

[0216] 일부 예에서, 다수의 영역은 다수의 영역의 콘텐츠에 기초하여 순위가 매겨진다. 각각의 콘텐츠는 다수의 영역 중 각각에 대응할 수 있다. 일례에서, 다수의 영역 각각에 대응하는 콘텐츠 각각은 대응하는 영역에서 스크린 콘텐츠의 백분율 또는 텍스트 콘텐츠의 백분율로 지시되고, 다수의 영역은 스크린 콘텐츠의 백분율 각각 또는 텍스트 콘텐츠의 백분율 각각에 기초하여 순위가 매겨진다.

[0217] 일부 예에서, 다수의 영역의 각각의 영역은 미리 정의된 예측 모드 세트 중의 하나 이상의 예측 모드로 코딩되는 각각의 영역에서 특별히 코딩된 다수의 참조 샘플을 포함한다. 다수의 영역은 다수의 영역에서 특별히 코딩된 샘플 각각의 수에 기초하여 순위가 매겨질 수 있다.

[0218] 일부 예에서, 다수의 영역의 각각의 영역과 연관된 참조 빈도는 현재 픽처 또는 현재 픽처의 일부에서 CB를 예측하는 데 사용되는 블록 벡터에 의해 각각의 영역이 참조되는 횟수를 지시한다. 다수의 영역은 다수의 영역의 각각의 참조 빈도에 기초하여 순위가 매겨질 수 있다.

[0219] (S1840) 후, 인코딩 프로세스(1800)는 (S1850)으로 진행한다. (S1850)에서, CB는 현재 로컬 참조 범위를 기초로 하는 IBC 모드를 사용하여 인코딩될 수 있다. (S1850) 후, 인코딩 프로세스(1800)는 (S1899)로 진행하여 끝난다.

[0220] 인코딩 프로세스(1800)는 다양한 시나리오에 적합하게 적응될 수 있고 프로세스(1800)의 단계는 그에 따라 조정될 수 있다. 인코딩 프로세스(1800)의 단계 중 하나 이상이 적응, 생략, 반복 및/또는 결합될 수 있다. 인코딩 프로세스(1800)를 구현하기 위해 임의의 적절한 순서가 사용될 수 있다. 추가 단계(들)가 추가될 수 있다.

[0221] 일부 예에서, 예측 정보는 이전 로컬 참조 범위를 영역 내의 샘플로 업데이트할지를 지시하는 플래그를 포함한다. 플래그는 인코딩되어 비디오 비트스트림에 포함될 수 있다. 예를 들어, CB에 대해 단일 플래그가 시그널링된다. 일례에서, CB에 대해 복수의 플래그가 시그널링된다. 예를 들어, CB는 복수의 구역 각각이 메모리 업데이트 영역 단위의 크기를 갖는 복수의 구역을 포함하고, 각각의 구역에 대해 개별 플래그가 시그널링된다.

[0222] 도 19는 본 개시의 일 실시예에 따른 디코딩 프로세스(1900)를 개략적으로 설명하는 흐름도를 도시한다. 디코딩 프로세스(1900)는 비디오 인코더에 사용될 수 있다. 다양한 실시예에서, 디코딩 프로세스(1900)는 단말 기기(310, 320, 330, 340)의 처리 회로, 비디오 인코더(403)의 기능을 수행하는 처리 회로, 비디오 인코더(403)의 기능을 수행하는 처리 회로, 비디오 디코더(510)의 기능을 수행하는 처리 회로, 비디오 인코더(603)의 기능을 수행하는 처리 회로 등과 같은, 처리 회로에 의해 실행된다. 일부 실시예에서, 디코딩 프로세스(1900)는 소프트웨어 명령어로 구현되며, 따라서 처리 회로가 소프트웨어 명령어를 실행할 때, 처리 회로는 디코딩 프로세스(1900)를 수행한다. 프로세스는 (S1901)에서 시작하여 (S1910)으로 진행한다.

[0223] (S1910)에서, 코딩된 비디오 비트스트림으로부터 현재 픽처의 블록 내의 영역을 재구축할 수 있다. 블록은 현재 픽처에서 하나 이상의 코딩 블록(CBs)을 포함할 수 있다. 블록은 현재 픽처의 슈퍼블록 또는 CTB일 수 있다. 일례에서, 슈퍼블록 또는 CTB와 같은 블록은 현재 픽처에서 가장 큰 CB이다. 일례에서, 영역은 슈퍼블록 또는 CTB의 CB이다. 일례에서, 영역은 슈퍼블록 또는 CTB 내의 CB의 일부이다. 일례에서 블록의 크기는 128×128이다. 일례에서, 영역의 크기는 64×64와 같이 메모리 업데이트 영역 단위의 크기이다.

[0224] (S1920) 단계에서, 재구축 후, 도 15에 설명된 바와 같이, 본 개시에 설명된 (i) 재구축된 영역의 콘텐츠 및/또

는 (ii) 재구축된 영역의 예측 정보에 기초하여 인트라 블록 복사(IBC) 모드에 대한 이전 로컬 참조 범위를 업데이트할지를 판정할 수 있다.

[0225] 일 실시예에서, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지는 재구축된 영역의 콘텐츠에 기초하여 결정된다. 예를 들어, 재구축된 영역의 콘텐츠는 재구축된 영역의 콘텐츠가 재구축된 영역의 텍스처 콘텐츠의 백분율 및/또는 스크린 콘텐츠의 백분율을 지시하는 재구축된 영역의 재구축된 샘플에 기초하여 결정된다. 따라서, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지는 재구축된 영역에서의 텍스처 콘텐츠의 백분율 및/또는 스크린 콘텐츠의 백분율에 기초하여 판정된다. 일례에서, 재구축된 영역의 콘텐츠는 스크린 콘텐츠를 포함하지 않으며, IBC 모드에 대한 이전 로컬 참조 범위는 업데이트되지 않는 것으로 결정된다.

[0226] 일 실시예에서, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지는 재구축된 영역의 예측 정보에 기초하여 결정된다. 예측 정보가 디코딩된다.

[0227] 일례에서, 영역의 예측 정보는 영역의 하나 이상의 예측 모드를 지시한다. IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지는 하나 이상의 예측 모드가 미리 정의된 예측 모드 세트에 있는지에 기초하여 판정된다. 일례에서, 미리 정의된 예측 모드 세트는 IBC 모드, 팔레트 모드, 변환 스윕 모드, BDPCM 모드 등과 같은 스크린 콘텐츠 코딩 모드(들)를 포함한다.

[0228] 일례에서, 영역의 예측 정보는 이전 로컬 참조 범위를 재구축된 영역의 재구축된 샘플로 업데이트할지 여부를 지시한다. IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지 여부는 예측 정보 내의, 재구축된 영역의 재구축된 샘플로 이전 로컬 참조 범위를 업데이트할지에 대한 지시에 기초하여 판정될 수 있다. 예를 들어, 재구축된 영역의 재구축된 샘플로 이전 로컬 참조 범위를 업데이트할지 여부는 그 영역의 예측 정보에 포함된 플래그로 지시된다. IBC 모드에 대한 이전 로컬 참조 범위를 업데이트할지 여부는 플래그에 기초하여 결정될 수 있다. 일례에서, 플래그는 코딩된 비디오 비트스트림에서 시그널링된다.

[0229] 일례에서, 64×64 와 같은, 메모리 업데이트 영역 단위의 크기를 갖는 각각의 영역에 대해 개별 플래그가 시그널링된다. 일례에서, 각각의 CB에 대해 개별 플래그가 신호된다.

[0230] 각각의 슈퍼블록 또는 각각의 CTB에 대해 개별 플래그가 시그널링될 수 있다. 일례에서, 플래그는 블록(예: 슈퍼블록 또는 CTB)의 재구축된 샘플(들)이 IBC 모드에서 로컬 참조 범위를 업데이트하는 데 사용되는지를 지시한다. 따라서, 단일 플래그(예: 플래그)가 블록(예: 슈퍼블록 또는 CTB)의 다수의 영역에서 재구축된 샘플이 IBC 모드에서 로컬 참조 범위를 업데이트하는 데 사용되는지를 지시하는 데 사용될 수 있다. 일례에서, 다수의 영역 각각은 CB 또는 CB의 일부이다. 일례에서, 복수의 영역 각각의 크기는 64×64 와 같은, 메모리 업데이트 영역 단위의 크기이다.

[0231] IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 결정한 것에 응답으로, 디코딩 프로세스(1900)는 단계(S1930)로 진행한다. 그렇지 않으면, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하기로 결정한 것에 응답으로, 경우 프로세스(1900)는 (S1940) 및 (S1950)을 포함하는 분기로 진행한다.

[0232] (S1930)에서, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하지 않기로 결정한다. 따라서, IBC 모드에 대한 이전 로컬 참조 범위는 업데이트되지 않는다. 현재 픽처의 코딩 블록(CB)은 재구축된 영역을 포함하지 않는 이전 로컬 참조 범위에 기초하는 IBC 모드를 사용하여 디코딩될 수 있다. 일례에서, CB는 슈퍼블록 또는 CTB에 있다.

[0233] (S1930) 후, 디코딩 프로세스(1900)는 (S1999)로 진행하여 끝난다.

[0234] (S1940) 및 (S1950)을 포함하는 분기에서, IBC 모드에 대한 이전 로컬 참조 범위를 업데이트하는 것으로 결정된다.

[0235] (S1940)에서, 이전 로컬 참조 범위 및 재구축된 영역의 재구축된 샘플에 기초하여 현재 로컬 참조 범위를 생성할 수 있다.

[0236] 이전 로컬 참조 범위는 참조 샘플 메모리(RSM)에 저장된 다수의 영역을 포함할 수 있다. 현재 로컬 참조 범위는 이전 로컬 참조 범위 내의 다수의 영역 중 하나의 참조 샘플을 재구축된 영역의 재구축된 샘플로 대체하여 생성할 수 있다.

[0237] 일부 실시예에서, 이전 로컬 참조 범위 내의 다수의 영역 중 대체될 영역은 다수의 영역의 순위에 기초하여 결정된다. 따라서, 다수의 영역의 순위에 기초하여 재구축된 영역의 재구축된 샘플로 대체될 다수의 영역 중의 영

역이 선택될 수 있다. 현재 로컬 참조 범위는 이전 로컬 참조 범위 내의 선택된 영역을 재구축된 영역의 재구축된 샘플로 대체하여 생성할 수 있다.

- [0238] 일부 예에서, 다수의 영역은 다수의 영역의 콘텐츠에 기초하여 순위가 매겨진다. 각각의 콘텐츠는 다수의 영역 각각에 대응할 수 있다. 일례에서, 다수의 영역 각각에 대응하는 콘텐츠 각각은 대응하는 영역에서의 스크린 콘텐츠의 백분율 또는 텍스처 콘텐츠의 백분율로 지시되고, 다수의 영역은 다수의 영역의 스크린 콘텐츠의 백분율을 각각 또는 텍스처 콘텐츠의 백분율 각각에 기초하여 순위가 매겨진다.
- [0239] 일부 예에서, 다수의 영역의 각각의 영역은 미리 정의된 예측 모드 세트 중의 하나 이상의 예측 모드로 코딩되는 각자의 영역에서 특별히 코딩된 다수의 참조 샘플을 포함한다. 다수의 영역은 다수의 영역에서 특별히 코딩된 샘플 각각의 수에 기초하여 순위가 매겨질 수 있다.
- [0240] 일부 예에서, 다수의 영역의 각각의 영역과 연관된 참조 빈도는 현재 꽂처 또는 현재 꽂처의 일부에서 CB를 예측하는 데 사용되는 블록 벡터에 의해 각각의 영역이 참조되는 횟수를 지시한다. 다수의 영역은 다수의 영역의 각각의 참조 빈도에 기초하여 순위가 매겨질 수 있다.
- [0241] (S1950)에서, 현재 꽂처의 CB는 재구축된 영역을 포함하는 현재 로컬 참조 범위를 기초로 하는 IBC 모드를 사용하여 디코딩될 수 있다.
- [0242] (S1950) 후, 프로세스(1900)는 (S1999)로 진행하여 끝난다.
- [0243] 디코딩 프로세스(1900)는 다양한 시나리오에 적합하게 적응될 수 있고 디코딩 프로세스(1900)의 단계는 그에 따라 조정될 수 있다. 디코딩 프로세스(1900)의 단계 중 하나 이상이 적응, 생략, 반복 및/또는 결합될 수 있다. 디코딩 프로세스(1900)를 구현하기 위해 임의의 적절한 순서가 사용될 수 있다. 추가 단계(들)가 추가될 수 있다.
- [0244] 일례에서, 이전 로컬 참조 범위는 다수의 영역을 포함한다. 다수의 영역 중 하나에서의 서브영역의 재구축된 샘플을 제외한 다수의 영역의 재구축된 샘플은 RSM에 저장된다. 다수의 영역 중 하나에서의 서브영역의 재구축된 샘플은 RSM에 다수의 영역 중 하나에서의 서브영역에 대응하는 패딩된 샘플을 저장하는 패딩 프로세스에 의해 대체된다.
- [0245] 본 개시의 실시예는 개별적으로 또는 임의의 순서로 조합되어 사용될 수 있다. 또한, 각각의 방법(또는 실시예), 인코더 및 디코더는 처리 회로(예: 하나 이상의 프로세서 또는 하나 이상의 집적 회로)에 의해 구현될 수 있다. 일례에서, 하나 이상의 프로세서는 컴퓨터로 판독 가능한 비일적 매체에 저장된 프로그램을 실행한다.
- [0246] 전술한 기술은 컴퓨터로 판독 가능한 명령어를 사용하고 하나 이상의 컴퓨터로 판독 가능한 매체에 물리적으로 저장될 수 있는 컴퓨터 소프트웨어로서 구현될 수 있다. 예를 들어, 도 20은 개시된 주제의 특정 실시예를 구현하기에 적합한 컴퓨터 시스템(2000)을 도시한다.
- [0247] 컴퓨터 소프트웨어는 컴퓨터 중앙 처리 유닛(central processing unit, CPU), 그래픽 처리 유닛(graphics processing unit, GPU) 등에 의해, 어셈블리(assembly), 컴파일(compilation), 링킹(linking), 또는 이와 유사한 메커니즘을 거쳐 직접, 또는 해석(interpretation), 마이크로 코드 실행(micro-code execution) 등을 통해 실행될 수 있는 명령어를 포함하는 코드를 생성할 수 있는 임의의 적절한 기계어 코드 또는 컴퓨터 언어를 사용하여 코딩될 수 있다.
- [0248] 명령어는, 예를 들어 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트폰, 게이밍 기기, 사물인터넷 기기 등을 포함한, 다양한 유형의 컴퓨터 또는 그 구성요소에서 실행될 수 있다.
- [0249] 도 20에 도시된 컴퓨터 시스템(2000)의 구성요소는 본질적으로 예시적인 것이며, 본 개시의 실시예를 구현하는 컴퓨터 소프트웨어의 사용 또는 기능의 범위에 대한 어떠한 한정도 시사하려는 것은 아니다. 구성요소의 구성은 컴퓨터 시스템(2000)의 예시적인 실시예에 나타낸 구성요소 중 어느 하나 또는 조합과 관련된 임의의 종속성 또는 요건을 가지는 것으로 해석되어서는 안 된다.
- [0250] 컴퓨터 시스템(2000)은 특정 휴먼 인터페이스 입력 기기(human interface input device)를 포함할 수 있다. 이러한 휴먼 인터페이스 입력 기기는 한 명 이상의 인간 사용자에 의한 입력, 예를 들어 촉각 입력(예: 키 누름(keystroke), 스와이프(swip), 데이터 장갑(data glove) 움직임), 오디오 입력(예: 음성, 박수), 시각적 입력(예: 제스처), 후각 입력(도시되지 않음)에 응답할 수 있다. 휴먼 인터페이스 기기는 또한 오디오(예: 음성, 음악, 주변 소리), 이미지(예: 스캔된 이미지, 정지 이미지 카메라로부터 획득한 사진 이미지), 비디오(예: 2차원

비디오, 입체 비디오(stereoscopic video)를 포함한 3차원 비디오)와 같은, 사람에 의한 의식적 입력과 반드시 직접 관련이 있는 것은 아닌 특정 미디어를 캡처하는 데 사용될 수도 있다.

[0251] 입력 휴먼 인터페이스 기기는 키보드(2001), 마우스(2002), 트랙 패드(2003), 터치 스크린(2010), 데이터 장갑(도시되지 않음), 조이스틱(2005), 마이크로폰(2006), 스캐너(2007), 카메라(2008) 중 하나 이상(각각 하나만 표시됨)을 포함할 수 있다.

[0252] 컴퓨터 시스템(2000)은 특정 휴먼 인터페이스 출력 기기를 포함할 수 있다. 이러한 휴먼 인터페이스 출력 기기는 예를 들어 촉각 출력, 소리, 빛 및 냄새/맛을 통해, 한 명 이상의 인간 사용자의 감각을 자극할 수 있다. 이러한 휴먼 인터페이스 출력 기기는 촉각 출력 기기(예: 터치 스크린(2010), 데이터 장갑(도시되지 않음), 또는 조이스틱(2005)에 의한 촉각 피드백이지만, 입력 기기의 역할을 하지 않는 촉각 피드백 기기도 있을 수 있음), 오디오 출력 기기(예: 스피커(2009), 헤드폰(도시되지 않음)), 시각적 출력 기기(예: 각각 터치 스크린 입력 기능이 있거나 없는, 각각 촉각 피드백 기능이 있거나 없는 CRT 스크린, LCD 스크린, 플라즈마 스크린, OLED 스크린을 포함한, 스크린(2010) - 그 일부는 스테레오그래픽 출력(stereographic), 가상 현실 안경(virtual-reality glasses)(도시되지 않음), 홀로그래픽 디스플레이(holographic display) 및 연기 탱크(smoke tank)(도시되지 않음)와 같은 수단을 통해 2차원 시각적 출력 또는 3차원 이상의 출력을 할 수 있음 -), 및 프린터(도시되지 않음)를 포함할 수 있다.

[0253] 컴퓨터 시스템(2000)은 또한 CD/DVD 등의 매체(2021)를 갖는 CD/DVD ROM RW(2020)을 포함한 광학 매체, 썬 드라이브(thumb-drive)(2022), 착탈 가능한 하드 드라이브 또는 솔리드 스테이트 드라이브(2023), 테이프 및 플로피 디스크(도시되지 않음)와 같은 레거시 자기 매체, 보안 동글(security dongle)(도시되지 않음)과 같은 특수한 ROM/ASIC/PLD 기반 기기 등의 인간이 액세스 가능할 수 있는 저장 기기 및 그 연관 매체를 포함할 수도 있다.

[0254] 당업자는 또한 현재 개시된 주제와 관련하여 사용된 바와 같이 컴퓨터로 판독 가능한 매체"라는 용어가 송신 매체, 반송파(carrier wave) 또는 기타 일시적인 신호를 포함하지 않는다는 것을 이해해야 한다.

[0255] 컴퓨터 시스템(2000)은 또한 하나 이상의 통신 네트워크(2055)에 대한 인터페이스(2054)를 포함할 수 있다. 네트워크는 예를 들어 무선, 유선, 광 등의 네트워크일 수 있다. 네트워크는 또한 로컬, 광역, 대도시, 차량 및 산업, 실시간, 지연 허용 등의 네트워크일 수 있다. 네트워크의 예로는 이더넷, 무선 LAN, GSM, 3G, 4G, 5G, LTE, 등을 포함하는 셀룰러 네트워크; 케이블 TV, 위성 TV, 및 지상파 방송 TV를 포함하는 TV 유선 또는 무선 광역 디지털 네트워크; CANBus를 포함하는 차량 및 산업용, 등을 포함한다. 특정 네트워크는 일반적으로 특정 범용 데이터 포트 또는 주변 버스(2049)(예: 컴퓨터 시스템(2000)의 USB 포트)에 부착된 외부 네트워크 인터페이스 어댑터를 필요로 하며; 다른 것은 일반적으로 이하에 설명하는 바와 같이 시스템 버스에 부착함으로써 컴퓨터 시스템(2000)의 코어에 통합된다(예: PC 컴퓨터 시스템에의 이더넷 인터페이스 또는 스마트폰 컴퓨터 시스템에의 셀룰러 네트워크 인터페이스). 이러한 네트워크 중 임의의 것을 사용하여, 컴퓨터 시스템(2000)은 다른 네트워크와 통신할 수 있다. 이러한 통신은 단방향, 수신 전용(예: TV 방송), 단방향 전송 전용(예: CANbus에서 특정 CANbus 기기로) 또는 양방향(예: 로컬 또는 광역 디지털 네트워크를 사용하여 다른 컴퓨터 시스템으로)일 수 있다. 특정 프로토콜 및 프로토콜 스택이 전술한 바와 같은 네트워크 및 네트워크 인터페이스 각각에 사용될 수 있다.

[0256] 전술한 휴먼 인터페이스 기기, 인간이 액세스 가능한 저장 기기 및 네트워크 인터페이스는 컴퓨터 시스템(2000)의 코어(2040)에 부착될 수 있다.

[0257] 코어(2040)는 하나 이상의 중앙 처리 유닛(CPU)(2041), 그래픽 처리 유닛(GPU)(2042), 필드 프로그래머블 게이트 어레이(Field Programmable Gate Area, FPGA)(2043) 형태의 특화된 프로그램 가능한 처리 유닛, 특정 태스크를 위한 하드웨어 가속기(2044), 그래픽 어댑터(2050) 등을 포함할 수 있다. 판독 전용 메모리(Read-only memory, ROM)(2045), 랜덤 액세스 메모리(2046), 사용자가 액세스할 수 없는 내부 하드 드라이브, SSD 등의 내부 대용량 저장장치(2047)와 함께, 이러한 기기는 시스템 버스(2048)를 통해 연결될 수 있다. 일부 컴퓨터 시스템에서, 시스템 버스(2048)는 추가적인 CPU, GPU 등에 의한 확장을 가능하게 하는 하나 이상의 물리 플리그의 형태로 액세스 가능할 수 있다. 주변 기기는 코어의 시스템 버스(2048)에 직접 연결되거나 주변 버스(2049)를 통해 연결될 수 있다. 일례에서, 스크린(2010)이 그래픽 어댑터(2050)에 연결될 수 있다. 주변 버스를 위한 아키텍처로는 PCI, USB 등을 포함한다.

[0258] CPU(2041), GPU(2042), FPGA(2043) 및 가속기(2044)는, 조합하여, 전술한 컴퓨터 코드를 구성할 수 있는 특정

명령어를 실행할 수 있다. 그 컴퓨터 코드는 ROM(2045) 또는 RAM(2046)에 저장될 수 있다. 이행 데이터(transitional data)는 RAM(2046)에 저장될 수 있는 반면, 영구 데이터는 예를 들어, 내부 대용량 저장장치(2047)에 저장될 수 있다. 메모리 소자 중 어느 것에 대한 빠른 저장 및 검색은, 하나 이상의 CPU(2041), GPU(2042), 대용량 저장장치(2047), ROM(2045), RAM(2046) 등과 밀접하게 연관될 수 있는 캐시 메모리의 사용을 통해 가능해질 수 있다.

[0259] 컴퓨터로 판독 가능한 매체는 다양한 컴퓨터로 구현되는 동작(computer-implemented operation)을 수행하기 위한 컴퓨터 코드를 가질 수 있다. 매체 및 컴퓨터 코드는 본 개시의 목적을 위해 특별히 설계되고 구축된 것일 수 있거나, 컴퓨터 소프트웨어 분야의 당업자에게 잘 알려져 있고 이용 가능한 종류일 수 있다.

[0260] 한정이 아닌 예로서, 아키텍처(2000), 구체적으로 코어(2040)를 갖는 컴퓨터 시스템은 하나 이상의 유형의 컴퓨터로 판독 가능한 매체에 구현된 소프트웨어 실행하는 프로세서(들)(CPU, GPU, FPGA, 가속기 등을 포함)의 결과로서 기능을 제공할 수 있다. 이러한 컴퓨터로 판독 가능한 매체는 위에서 소개한 바와 같이 사용자가 액세스할 수 있는 대용량 저장장치와 연관된 매체일 수 있을 뿐만 아니라 코어 내부 대용량 저장장치(2047) 또는 ROM(2045)과 같은, 비일시적인 성질의 코어(2040)의 특정 저장장치일 수 있다. 본 개시의 다양한 실시예를 구현하는 소프트웨어는 이러한 기기에 저장되고 코어(2040)에 의해 실행될 수 있다. 컴퓨터로 판독 가능한 매체는 구체적인 필요에 따라, 하나 이상의 메모리 소자 또는 칩을 포함할 수 있다. 소프트웨어는 코어(2040) 및 특히 내부의 프로세서(CPU, GPU, FPGA 등 포함)로 하여금 RAM(2046)에 저장된 데이터 구조를 정의하고 소프트웨어에 의해 정의된 프로세스에 따라 이러한 데이터 구조를 수정하는 것을 포함하여, 여기에 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하도록 할 수 있다. 추가로 또는 대안으로서, 컴퓨터 시스템은 여기에 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하기 위해 소프트웨어 대신 또는 소프트웨어와 함께 작동할 수 있는, 논리 배선(logic hardwired)의 결과로서 그렇지 않으면 회로(예: 가속기(2044))에 다른 방식으로 구현되는 기능을 제공할 수 있다. 소프트웨어에 대한 언급은 논리를 포함할 수 있으며, 적절한 경우 그 반대도 마찬가지이다. 컴퓨터로 판독 가능한 매체에 대한 언급은 실행을 위한 소프트웨어를 저장하는 회로(예: 집적회로(IC)), 실행을 위한 논리를 구현하는 회로, 또는 적절한 경우 둘 다를 포함할 수 있다. 본 개시는 하드웨어와 소프트웨어의 임의의 적절한 조합을 포함한다.

[0261] [부록 A: 약어]

[0262] JEM: joint exploration model

[0263] VVC: versatile video coding

[0264] BMS: benchmark set

[0265] MV: Motion Vector

[0266] HEVC: High Efficiency Video Coding

[0267] SEI: Supplementary Enhancement Information

[0268] VUI: Video Usability Information

[0269] GOPs: Groups of Pictures

[0270] TUs: Transform Units,

[0271] PUs: Prediction Units

[0272] CTUs: Coding Tree Units

[0273] CTBs: Coding Tree Blocks

[0274] PBs: Prediction Blocks

[0275] HRD: Hypothetical Reference Decoder

[0276] SNR: Signal Noise Ratio

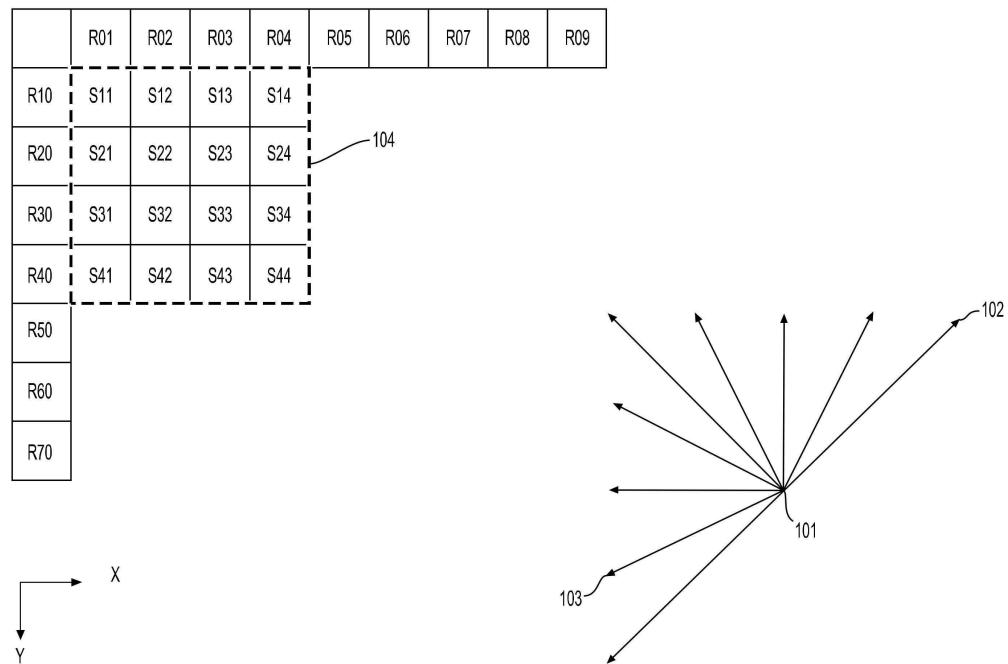
[0277] CPUs: Central Processing Units

[0278] GPUs: Graphics Processing Units

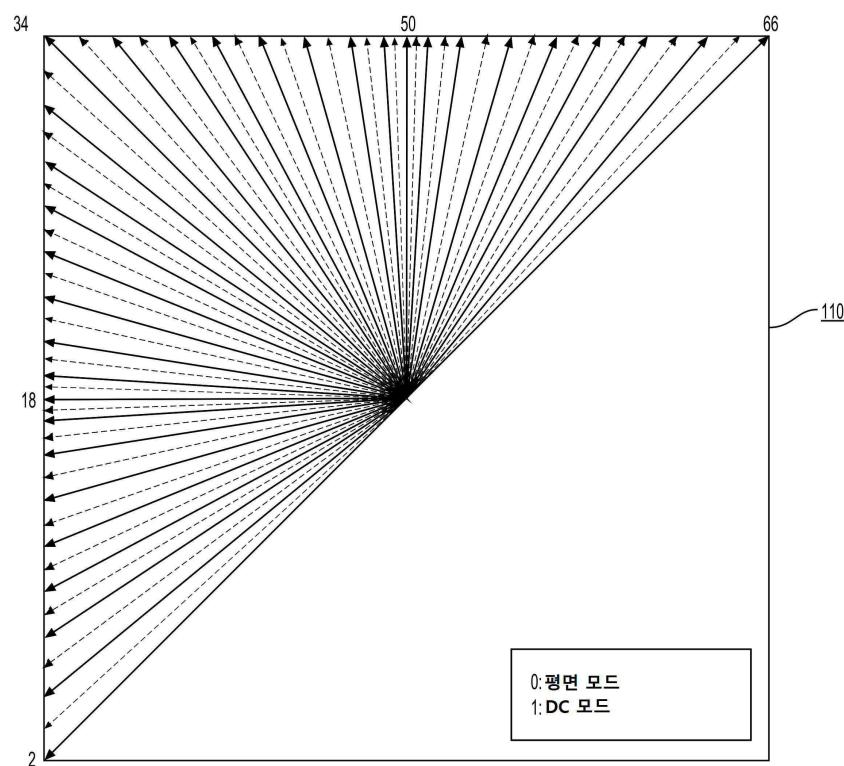
[0279]	CRT: Cathode Ray Tube
[0280]	LCD: Liquid-Crystal Display
[0281]	OLED: Organic Light-Emitting Diode
[0282]	CD: Compact Disc
[0283]	DVD: Digital Video Disc
[0284]	ROM: Read-Only Memory
[0285]	RAM: Random Access Memory
[0286]	ASIC: Application-Specific Integrated Circuit
[0287]	PLD: Programmable Logic Device
[0288]	LAN: Local Area Network
[0289]	GSM: Global System for Mobile communications
[0290]	LTE: Long-Term Evolution
[0291]	CANBus: Controller Area Network Bus
[0292]	USB: Universal Serial Bus
[0293]	PCI: Peripheral Component Interconnect
[0294]	FPGA: Field Programmable Gate Areas
[0295]	SSD: solid-state drive
[0296]	IC: Integrated Circuit
[0297]	CU: Coding Unit
[0298]	R-D: Rate-Distortion
[0299]	본 개시는 몇몇 예시적인 실시예를 설명했지만, 본 개시의 범위 내에 속하는 변경, 순열 및 다양한 대체 등가물이 있다. 따라서, 당업자는 본 명세서에 명시적으로 도시되지 않거나 설명되지 않았지만, 본 개시의 원리를 구현하고 따라서 본 개시의 사상 및 범위 내에 있는 수많은 시스템 및 방법을 고안할 수 있음을 이해할 것이다.

도면

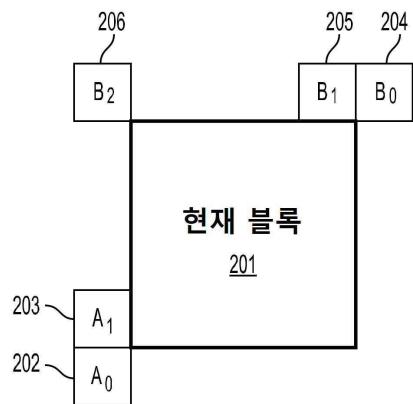
도면 1a



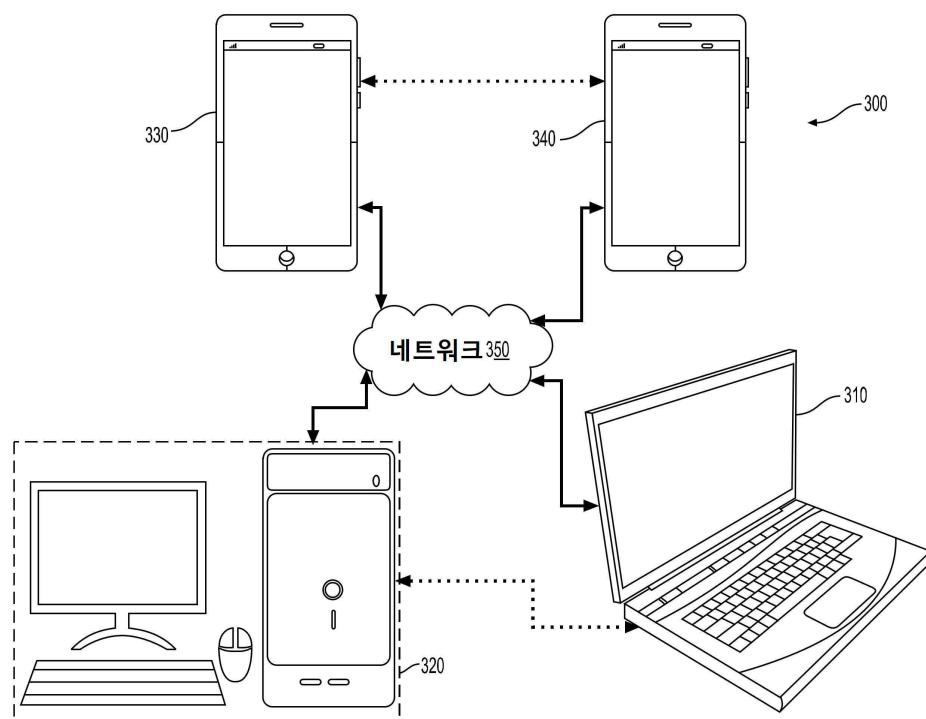
도면 1b



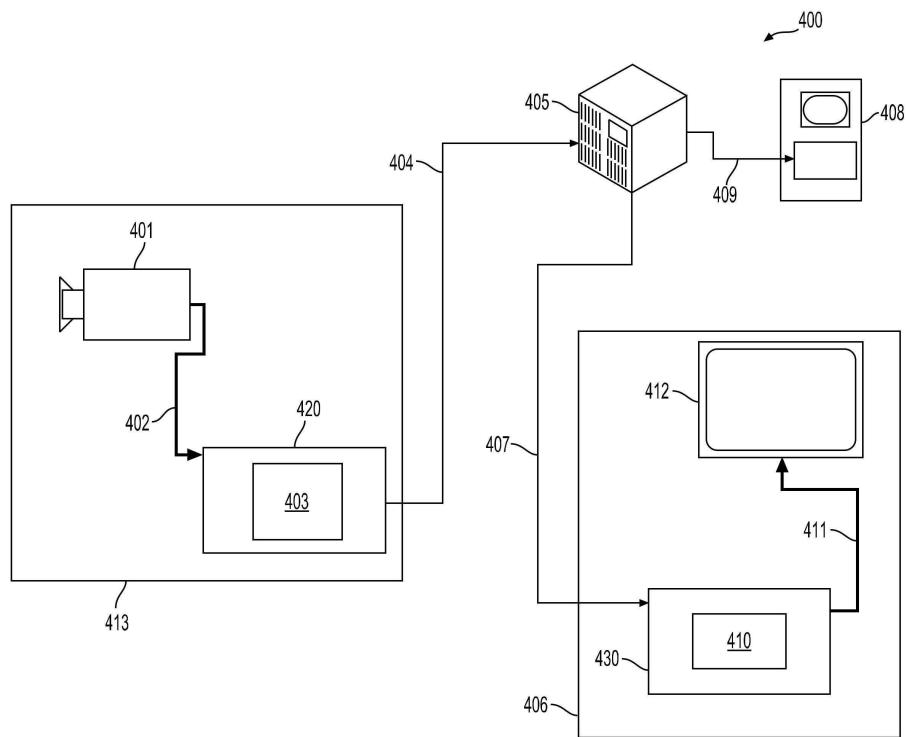
도면2



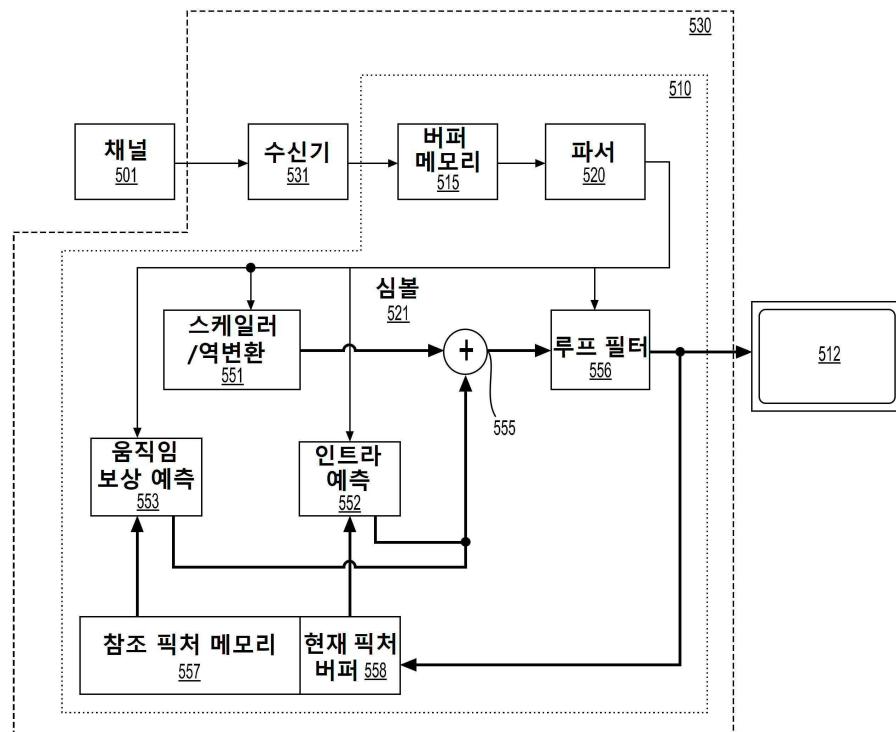
도면3



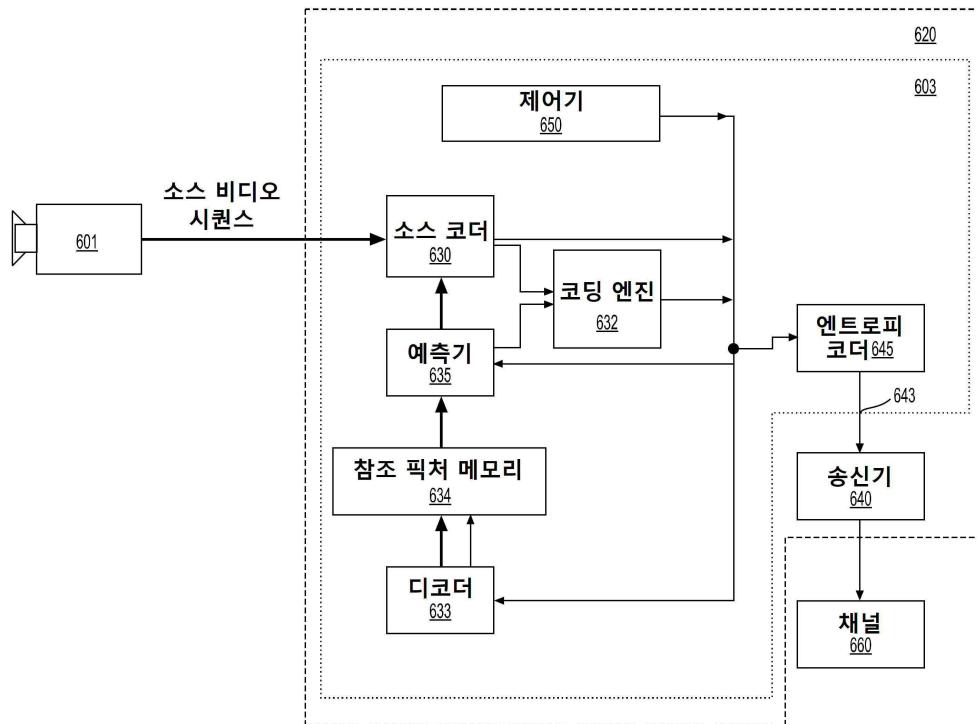
도면4



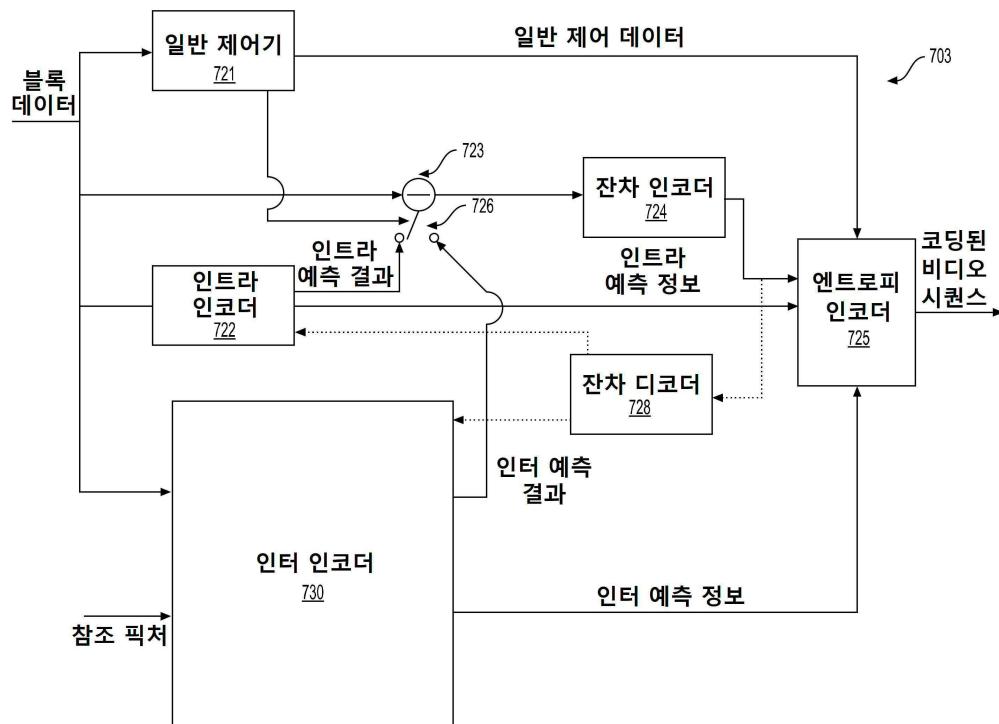
도면5



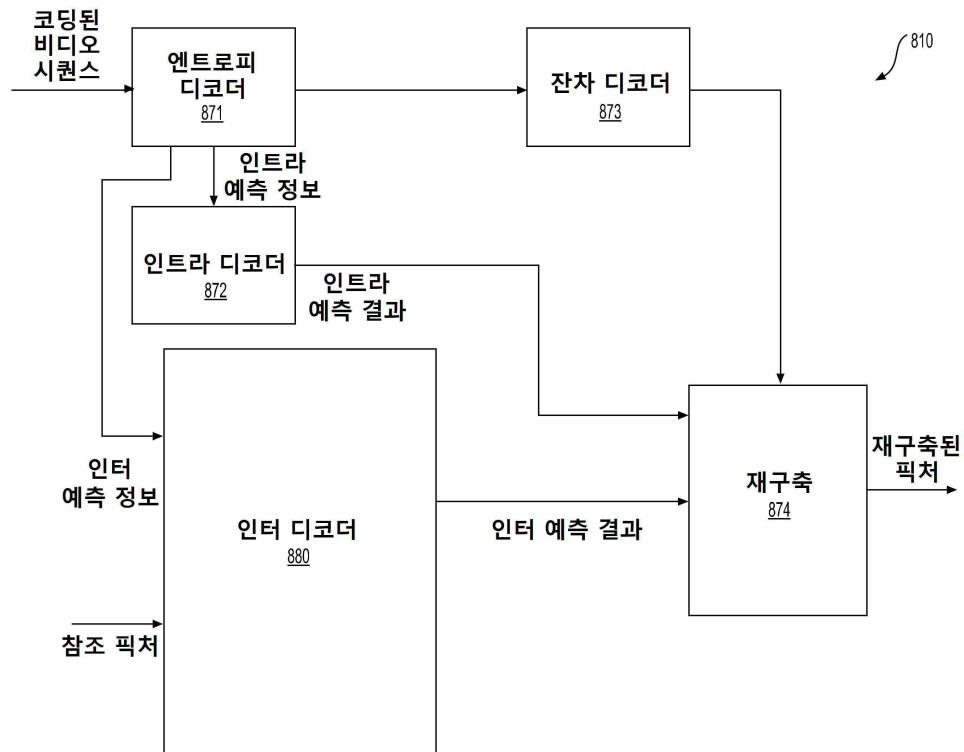
도면6



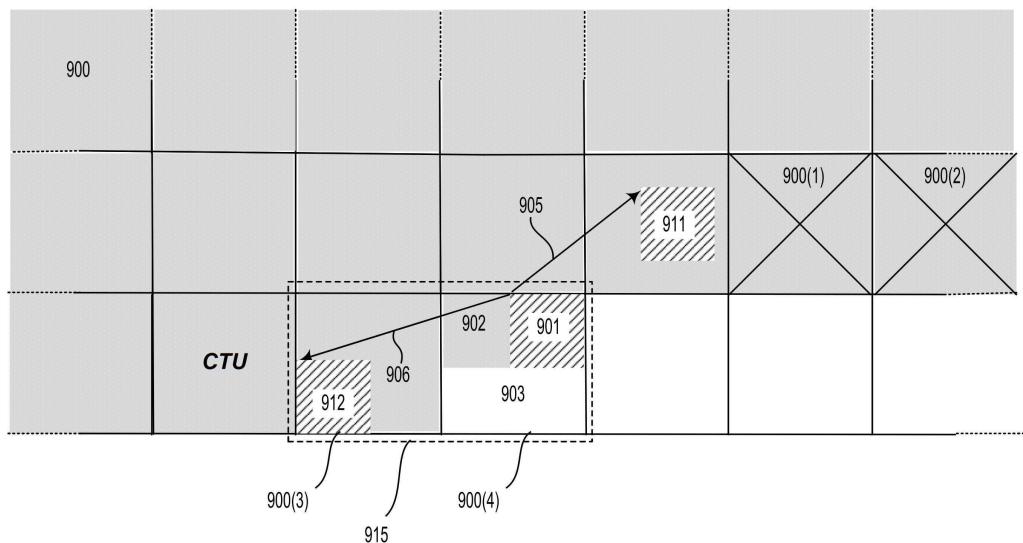
도면7



도면8



도면9



도면 10a

1011	1012	1021	1022
1013	1014	1023	1024

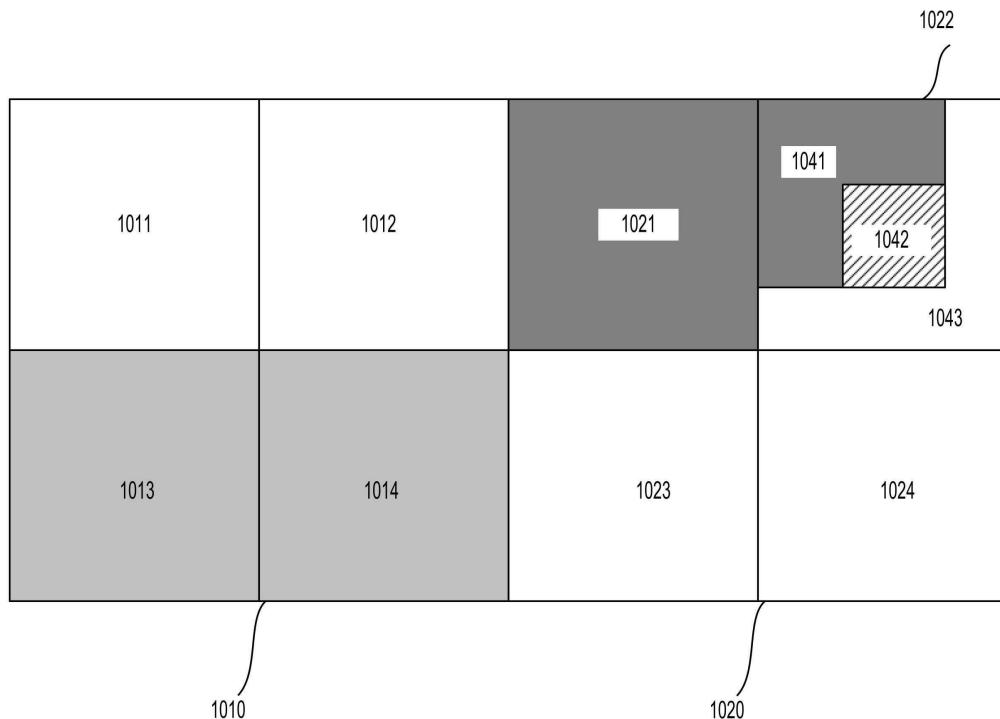
1010 1020

도면 10b

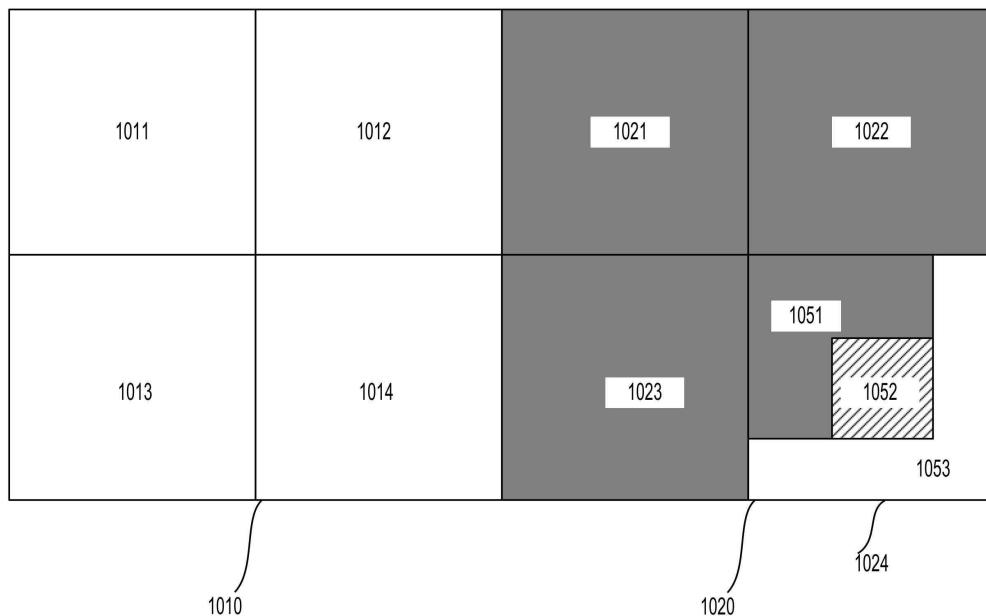
1011	1012	1021 1031 1032 1033	1022
1013	1014	1023	1024

1010 1020

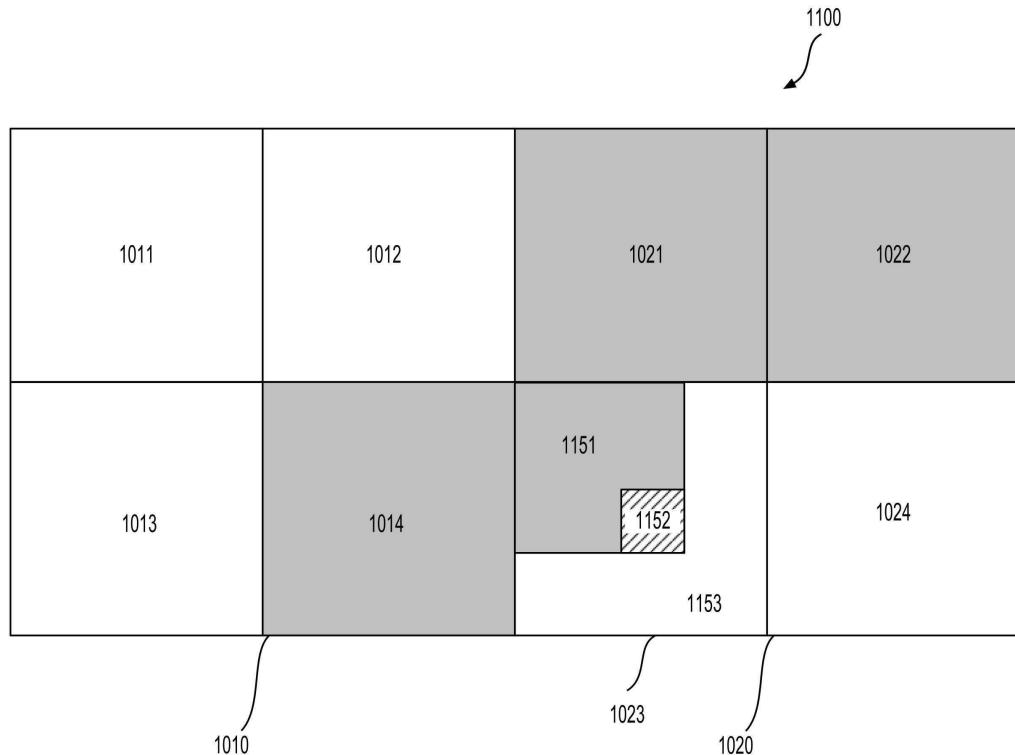
도면 10c



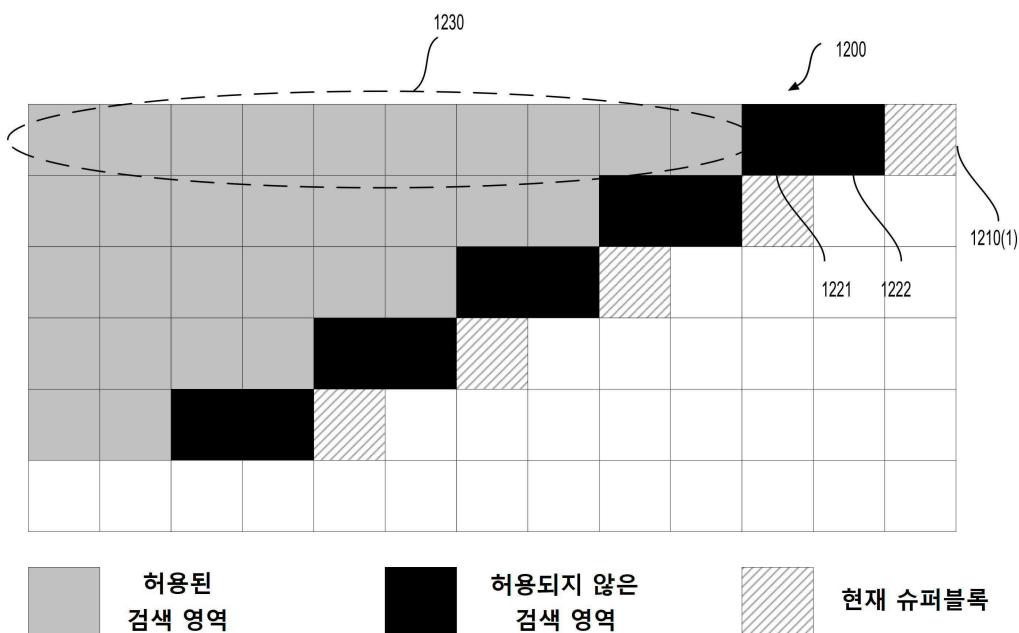
도면 10d



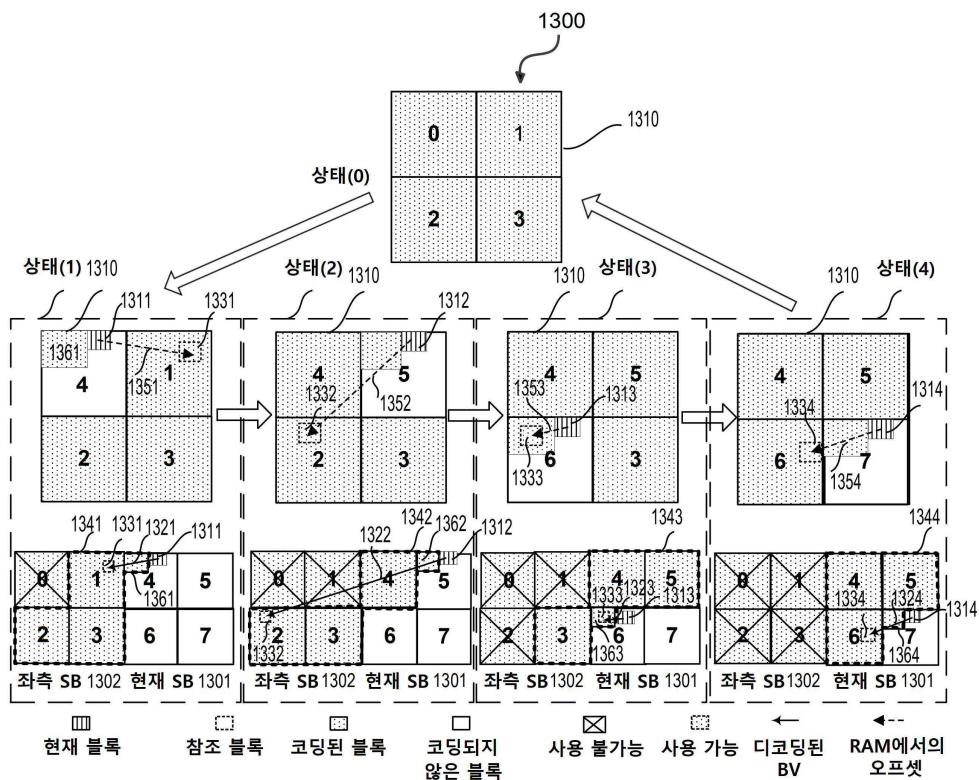
도면11



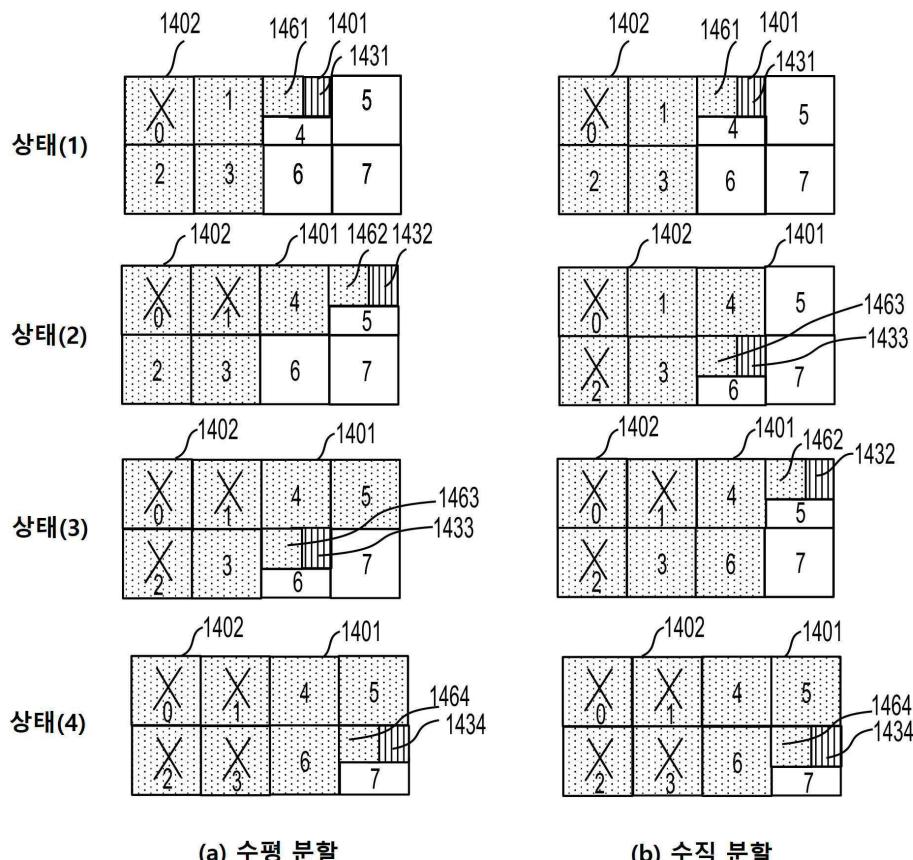
도면12



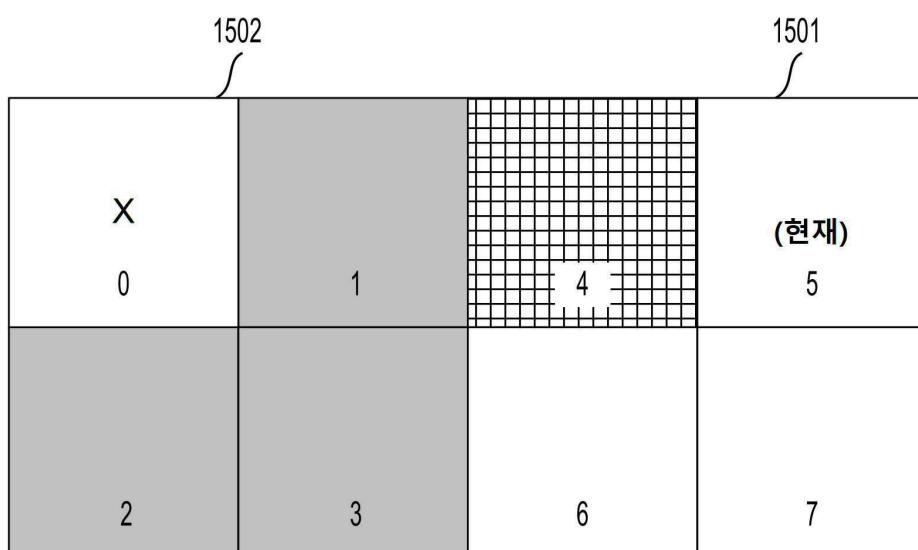
도면13



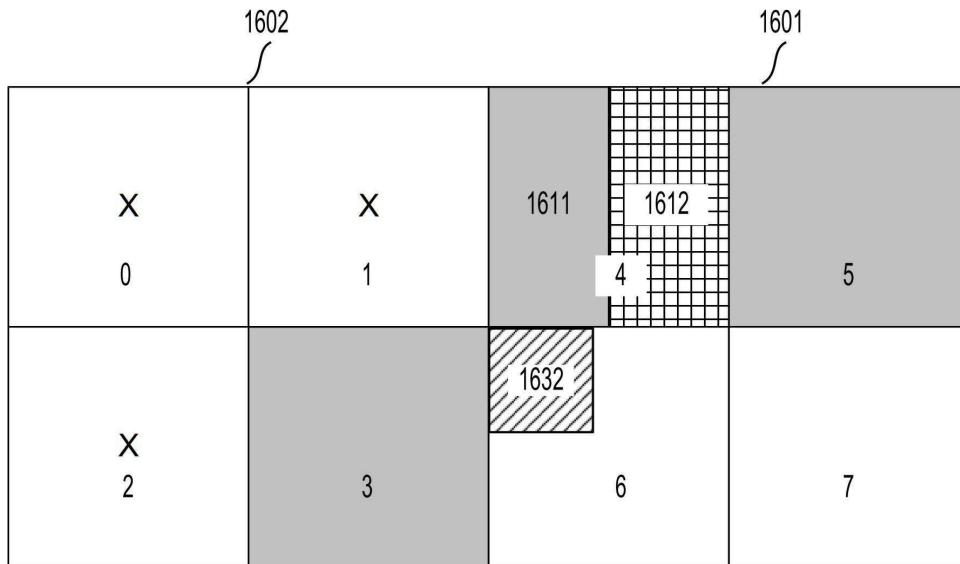
도면14



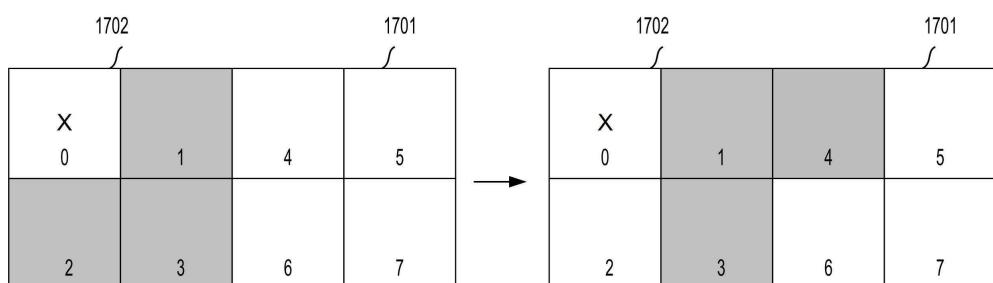
도면15



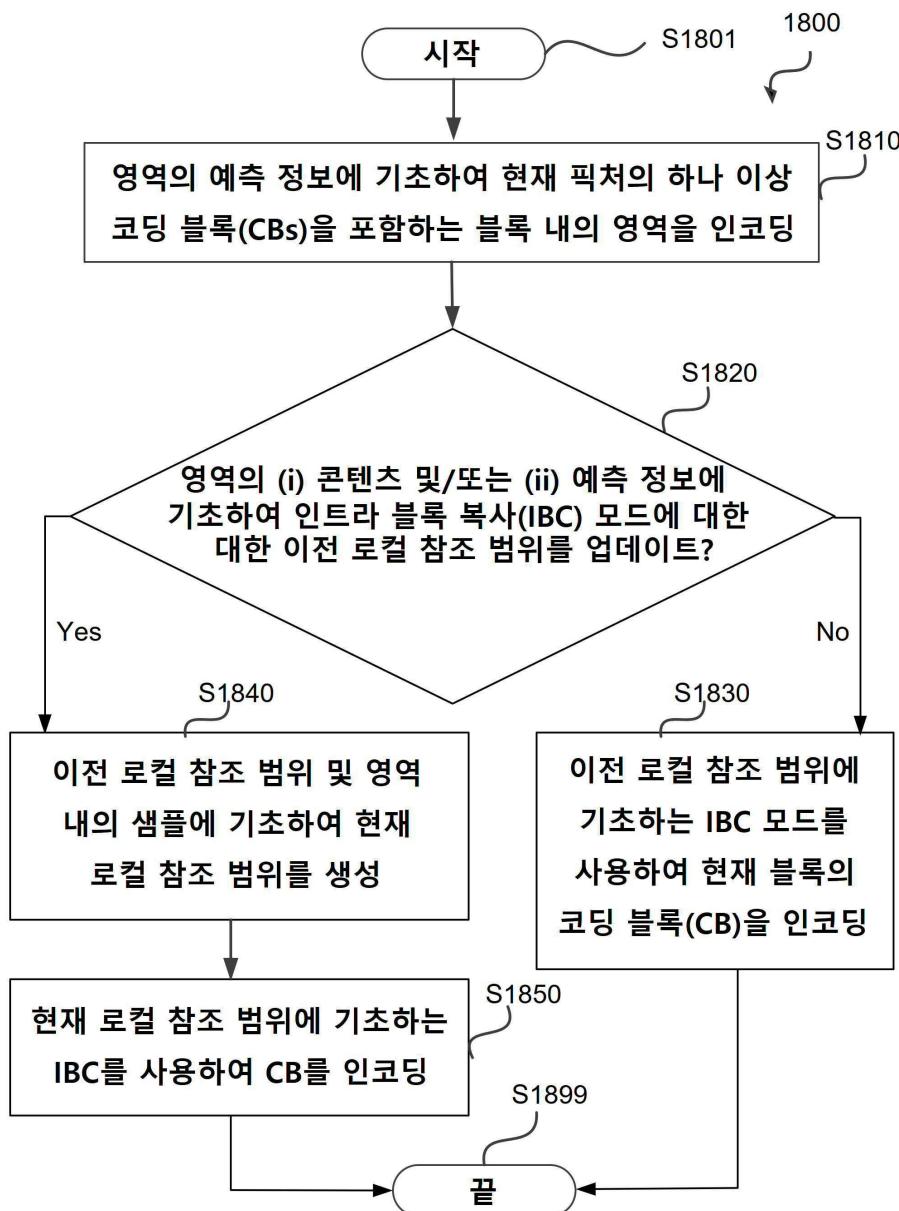
도면16



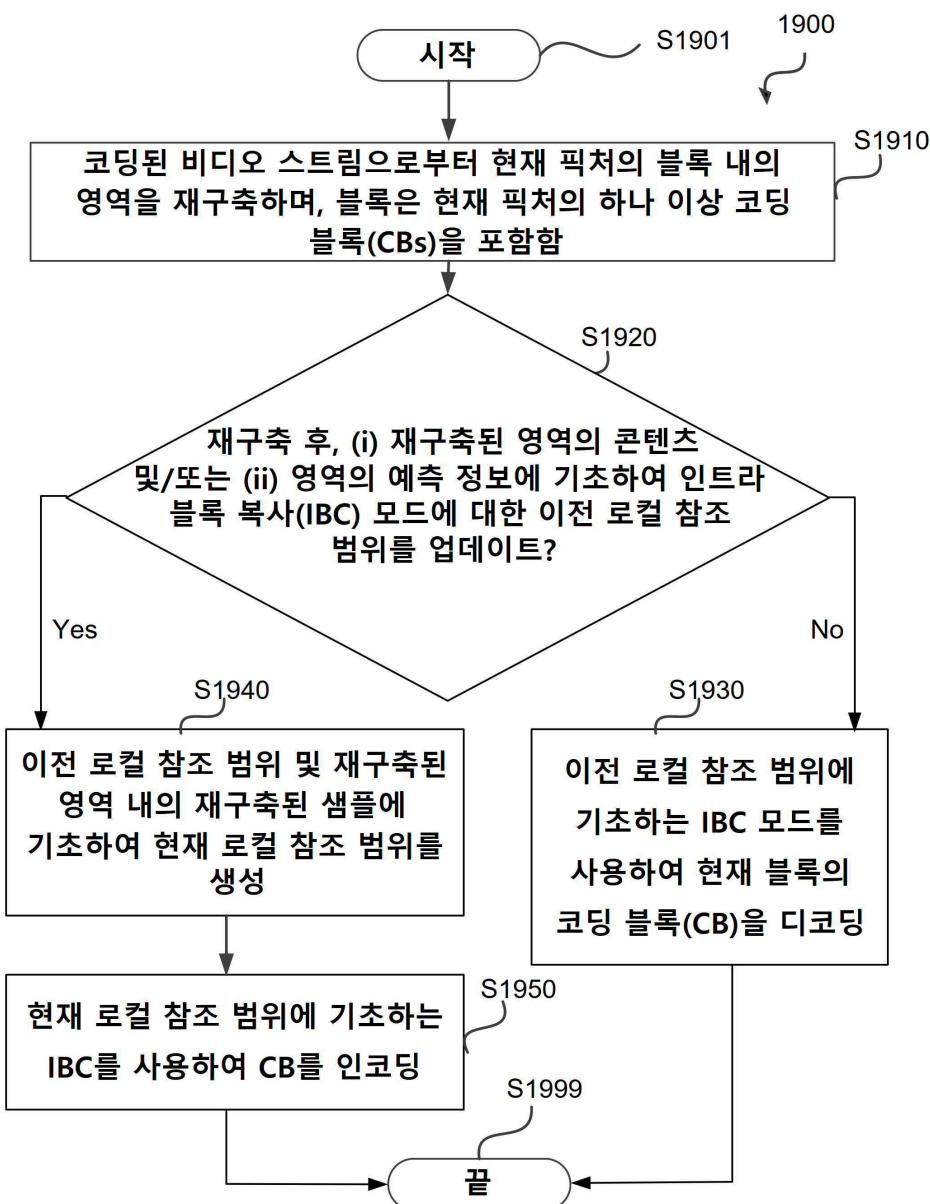
도면17



도면18



도면19



도면20

