(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2024년07월08일
(11) 등록번호 10-2683361
(24) 등록일자 2024년07월04일

(51) 국제특허분류(Int. Cl.)
H04N 19/70 (2014.01) *H04N 19/11* (2014.01)
H04N 19/176 (2014.01) *H04N 19/184* (2014.01)
H04N 19/593 (2014.01)
(52) CPC특허분류
H04N 19/70 (2015.01)
H04N 19/11 (2015.01)
(21) 출원번호 10-2021-7027532
(22) 출원일자(국제) 2020년06월10일
심사청구일자 2021년08월27일
(85) 번역문제출일자 2021년08월27일
(65) 공개번호 10-2021-0118178
(43) 공개일자 2021년09월29일
(86) 국제출원번호 PCT/US2020/036993
(87) 국제공개번호 WO 2021/006986
국제공개일자 2021년01월14일
(30) 우선권주장
62/873,044 2019년07월11일 미국(US)
(뒷면에 계속)

(56) 선행기술조사문헌
Yao-Jen Chang, et al., Non-CE8: On IBC merge mode, JVET-O0455r2, 2019.07.05.*
Xiaozhong Xu, et al., CE8: CPR reference memory reuse without increasing memory requirement (CE8.1.2a and CE8.1.2d), JVET-M0407-v3, 2019.01.17.
Xiaozhong Xu, BoG report on CE-8 related contributions, JVET-O1106, 2019.07.06.*
Benjamin Bross, et al., Versatile Video Coding (Draft 4), JVET-M1001-v1, 2019.02.01.
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
텐센트 아메리카 엘엘씨
미국 94306 캘리포니아주 팔로 알토 파크 블러바드 2747
(72) 발명자
수 샤오종
미국 94306 캘리포니아주 팔로 알토 2747 파크 불러바드 텐센트 아메리카 엘엘씨 내
리 샹
미국 94306 캘리포니아주 팔로 알토 2747 파크 불러바드 텐센트 아메리카 엘엘씨 내
리우 샨
미국 94306 캘리포니아주 팔로 알토 2747 파크 불러바드 텐센트 아메리카 엘엘씨 내
(74) 대리인
유미특허법인

전체 청구항 수 : 총 3 항 심사관 : 이남숙

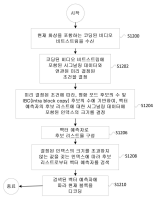
(54) 발명의 명칭 **인트라 화상 블록 보상에 대한 예측자 후보 리스트 크기 시그널링을 위한 방법 및 장치**

(57) 요 약

비디오 디코딩 방법은 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계를 포함한다. 상기 방법은 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계를 포함한다. 상기 방법은 미리 결정된 조건에 기반하여 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보 수에

(뒷면에 계속)

대 표 도



기반한 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계를 포함한다. 상기 방법은 벡터 예측자로 후보 리스트를 구성하는 단계를 포함한다. 상기 방법은 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 후보 리스트에서 벡터 예측자를 검색하는 단계를 포함한다. 상기 방법은 검색된 벡터 예측자에 따라 현재 블록을 디코딩하는 단계를 더 포함한다.

(52) CPC특허분류
H04N 19/176 (2015.01)
H04N 19/184 (2015.01)
H04N 19/52 (2015.01)
H04N 19/593 (2015.01)

(30) 우선권주장
62/904,307 2019년09월23일 미국(US)
16/863,661 2020년04월30일 미국(US)

명 세 서

청구범위

청구항 1

비디오 디코딩 방법으로서,

현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계;

상기 현재 화상에 포함된 현재 블록에 대한 상기 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계 - 상기 미리 결정된 조건은 최대 병합 모드 후보 수(maximum number of merge mode candidates)가 최대 IBC 후보 수(maximum number of IBC candidates)와 상이한지 여부를 명시(specify)함 - ;

상기 미리 결정된 조건에 기반하여, (i) 최대 병합 모드 후보 수 및 (ii) 최대 IBC 후보 수 중 하나로 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계;

벡터 예측자의 상기 후보 리스트를 구성하는 단계;

상기 인덱스의 크기를 초과하지 않는 값을 갖는 인덱스에 따라 벡터 예측자의 상기 후보 리스트에서 벡터 예측자를 검색하는 단계; 및

상기 검색된 벡터 예측자에 따라 상기 현재 블록을 디코딩하는 단계

를 포함하고,

상기 최대 병합 모드 후보 수가 상기 최대 IBC 후보 수와 상이한 경우, 상기 인덱스의 크기를 상기 최대 병합 모드 후보 수로 결정하고, 상기 최대 병합 모드 후보 수가 상기 최대 IBC 후보 수보다 작은 경우, 상기 인덱스의 크기를 상기 최대 IBC 후보 수로 결정하는, 비디오 디코딩 방법.

청구항 2

삭제

청구항 3

비디오 디코딩을 위한 비디오 디코딩 장치로서,

현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하고,

상기 현재 화상에 포함된 현재 블록에 대한 상기 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정 - 상기 미리 결정된 조건은 최대 병합 모드 후보 수(maximum number of merge mode candidates)가 최대 IBC 후보 수(maximum number of IBC candidates)와 상이한지 여부를 명시함 - 하며,

상기 미리 결정된 조건에 기반하여, (i) 최대 병합 모드 후보 수 및 (ii) 최대 IBC 후보 수 중 하나로 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하고,

벡터 예측자의 상기 후보 리스트를 구성하고,

상기 인덱스의 크기를 초과하지 않는 값을 갖는 인덱스에 따라 벡터 예측자의 상기 후보 리스트에서 벡터 예측자를 검색하고, 그리고

상기 검색된 벡터 예측자에 따라 상기 현재 블록을 디코딩하도록

구성된 처리 회로를 포함하고,

상기 최대 병합 모드 후보 수가 상기 최대 IBC 후보 수와 상이한 경우, 상기 처리 회로는 추가로 상기 인덱스의 크기를 상기 최대 병합 모드 후보 수로 결정하도록 구성되고, 상기 최대 병합 모드 후보 수가 상기 최대 IBC 후보 수보다 작은 경우, 상기 처리 회로는 추가로 상기 인덱스의 크기를 상기 최대 IBC 후보 수로 결정하도록 구성되는,

비디오 디코딩 장치.

청구항 4

삭제

청구항 5

명령이 저장된 컴퓨터가 판독가능한 비일시적 매체로서,

상기 명령은 비디오 디코더의 프로세서에 의해 실행될 때, 상기 프로세서가,

현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계;

상기 현재 화상에 포함된 현재 블록에 대한 상기 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계 - 상기 미리 결정된 조건은 최대 병합 모드 후보 수(maximum number of merge mode candidates)가 최대 IBC 후보 수(maximum number of IBC candidates)와 상이한지 여부를 명시함 -;

상기 미리 결정된 조건에 기반하여, (i) 최대 병합 모드 후보 수 및 (ii) 최대 IBC 후보 수 중 하나로 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계;

벡터 예측자의 상기 후보 리스트를 구성하는 단계;

상기 인덱스의 크기를 초과하지 않는 값을 갖는 인덱스에 따라 벡터 예측자의 상기 후보 리스트에서 벡터 예측자를 검색하는 단계; 및

상기 검색된 벡터 예측자에 따라 상기 현재 블록을 디코딩하는 단계

를 포함하고,

상기 최대 병합 모드 후보 수가 상기 최대 IBC 후보 수와 상이한 경우, 상기 인덱스의 크기를 상기 최대 병합 모드 후보 수로 결정하고, 상기 최대 병합 모드 후보 수가 상기 최대 IBC 후보 수보다 작은 경우, 상기 인덱스의 크기를 상기 최대 IBC 후보 수로 결정하는

방법을 수행하게 하는, 컴퓨터가 판독가능한 비일시적 매체.

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

발명의 설명

기 술 분 야

[0001] 본 출원은, 2019년 9월 23일에 출원되고 발명의 명칭이 "인트라 화상 블록 보상을 위한 예측 방법"인 미국 가출원 번호 제62/904,307호 및 2019년 7월 11일에 출원되고 발명의 명칭이 "인트라 화상 블록 보상을 위한 예측자 후보 리스트 크기 시그널링"인 미국 가출원 번호 제62/873,044호에 대한 우선권을 주장하는, 2020년 4월 30일에 출원되고 발명의 명칭이 "인트라 화상 블록 보상에 대한 예측자 후보 리스트 크기 시그널링을 위한 방법 및 장치"인 미국 출원 번호 제16/863,661호에 대한 우선권을 주장하는 바이다. 이전 출원의 전체 개시는 그 전체로서 원용에 의해 본 명세서에 포함된다.

[0002] 본 개시는 일반적으로 비디오 코딩과 관련된 실시예를 설명한다.

배 경 기 술

[0003] 여기에서 제공된 배경 설명은 본 개시의 맥락을 일반적으로 제시하기 위한 것이다. 현재 명명된 발명가의 작업은 이 배경 섹션에 설명된 한도 내에서 출원 당시 선행 기술로 자격이 되지 않을 수 있는 설명의 측면은 명시적 또는 묵시적으로 본 개시에 대한 선행 기술로 인정되지 않는다.

[0004] 비디오 코딩 및 디코딩은 모션 보상(motion compensation)이 있는 인터 화상 예측을 사용하여 수행될 수 있다. 압축되지 않은 디지털 비디오는 일련의 화상을 포함할 수 있으며, 각 화상은 예를 들어 1920×1080 휘도 샘플 및 관련 색차 샘플의 공간 차원을 갖는다. 일련의 화상은 예를 들어 초당 60개의 화상 또는 60Hz의 고정 또는 가변 화상 레이트(rate)(비공식적으로는 프레임 레이트라고도 함)를 가질 수 있다. 압축되지 않은 비디오는 상당한 비트레이트 요건을 가진다. 예를 들어, 샘플당 8비트의 1080p60 4:2:0 비디오(60Hz 프레임 레이트에서 1920×1080 휘도 샘플 해상도)에는 1.5 Gbit/s에 가까운 대역폭이 필요하다. 이러한 비디오 1시간에는 600 GBytes 이상의 저장 공간이 필요하다.

[0005] 비디오 코딩 및 디코딩의 한 가지 목적은 압축을 통해 입력 비디오 신호에서 중복성을 줄이는 것이다. 압축은 앞서 언급한 대역폭 또는 저장 공간 요건을, 경우에 따라 2배 이상으로 줄이는 데 도움이 될 수 있다. 무손실

(lossless) 압축과 손실(lossy) 압축, 그리고 이들의 조합이 사용될 수 있다. 무손실 압축은 압축된 원본(original) 신호에서 원본 신호의 정확한 카피(copy)를 재구성할 수 있는 기술을 나타낸다. 손실 압축을 사용할 때, 재구성된 신호는 원본 신호와 동일하지 않을 수 있지만 원본 신호와 재구성된 신호 사이의 왜곡은 재구성된 신호가 의도한 애플리케이션에 유용할 만큼 충분히 작다. 비디오의 경우, 손실 압축이 널리 사용된다. 허용되는 왜곡의 양은 애플리케이션에 따라 다르며; 예를 들어, 특정 소비자 스트리밍 애플리케이션의 사용자는 텔레비전 배포 애플리케이션의 사용자보다 더 높은 왜곡을 허용할 수 있다. 달성 가능한 압축 비율은 허가/허용 가능한 왜곡이 높을수록 압축 비율이 높아질 수 있음을 반영할 수 있다.

[0006] 모션 보상은 손실 압축 기술일 수 있으며 이전에 재구성된 화상 또는 그 일부(참조 화상)로부터의 샘플 데이터 블록이, 모션 벡터(motion vector)(이후 MV)에 의해 지시되는 방향으로 공간적으로 시프트된 후, 새로 재구성된 화상 또는 화상 부분의 예측에 사용된다. 일부 경우에, 참조 화상이 현재 재구성 중인 화상과 동일할 수 있다. MV는 X와 Y의 2차원 또는 3차원을 가질 수 있으며, 세 번째는 사용 중인 참조 화상의 지시(indication)이다(후자는 간접적으로 시간 차원이 될 수 있음).

[0007] 일부 비디오 압축 기술에서, 샘플 데이터의 특정 에어리어(area)에 적용 가능한 MV는 다른 MV로부터, 예를 들어 재구성 중인 에어리어에 공간적으로 인접하고 디코딩 순서에서 그 MV보다 앞선 샘플 데이터의 다른 에어리어와 관련된 것들로부터 예측될 수 있다. 그렇게 하면 MV를 코딩하는 데 필요한 데이터의 양을 상당히 줄일 수 있으므로 중복성을 제거하고 압축을 높일 수 있다. 예를 들어, MV 예측은 효과적으로 작동할 수 있으며, 이는 카메라에서 유도된 입력 비디오 신호(자연(natural) 비디오라고 함)를 코딩할 때 단일 MV가 적용 가능한 에어리어보다 큰 에어리어가 유사한 방향으로 이동할 통계적 가능성이 있기 때문이며, 따라서 일부 경우에, 이웃 에어리어의 MV로부터 유도된 유사한 모션 벡터를 사용하여 예측할 수 있다. 그 결과, 주어진 에어리어에 대해 발견된 MV가 주변 MV로부터 예측된 MV와 유사하거나 동일하게 되며, 엔트로피 코딩 후 MV를 직접 코딩하면 사용되는 것보다 적은 수의 비트로 표현될 수 있다. 일부 경우에, MV 예측은 원본 신호(즉, 샘플 스트림)으로부터 유도된 신호(즉, MV)의 무손실 압축의 예일 수 있다. 다른 경우에, 예를 들어 여러 주변의 MV로부터 예측자를 계산할 때 반올림 에러가 발생하기 때문에, MV 예측 자체는 손실이 될 수 있다.

[0008] 다양한 MV 예측 메커니즘이 H.265/HEVC(ITU-T Rec. H.265, "고효율 비디오 코딩", 2016년 12월)에 기술되어 있다. H.265가 제공하는 많은 MV 예측 메커니즘 중에서 여기에서 설명하는 기술은 이후 "공간 병합(spatial merge)"이라고 한다.

[0009] 도 1을 참조하면, 현재 블록(101)은 공간적으로 시프트된 동일한 크기의 이전 블록으로부터 예측 가능하도록 모션 탐색(search) 프로세스 동안 인코더에 의해 발견된 샘플을 포함한다. 그 MV를 직접 코딩하는 대신에, MV는 A0, A1 및 B0, B1, B2(각각 102에서 106)으로 표시된 5개의 주변 샘플 중 하나와 연관된 MV를 사용하여, 하나 이상의 참조 화상, 예를 들어 (디코딩 순서에서) 가장 최근의 참조 화상과 연관된 메타데이터로부터 유도될 수 있다. H.265에서, MV 예측은 이웃 블록이 사용하고 있는 동일한 참조 화상으로부터의 예측자를 사용할 수 있다. 후보 리스트를 형성하는 순서는 A0 → B0 → B1 → A1 → B2일 수 있다.

발명의 내용

[0010] 일 실시예에 따르면, 비디오 디코딩 방법은 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계를 포함한다. 상기 방법은 상기 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계를 포함한다. 상기 방법은 상기 미리 결정된 조건에 기반하여 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보 수에 기반하여, 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계를 포함한다. 상기 방법은 벡터 예측자로 상기 후보 리스트를 구성하는 단계를 포함한다. 상기 방법은 상기 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 상기 후보 리스트에서 벡터 예측자를 검색하는 단계를 포함한다. 상기 방법은 검색된 벡터 예측자에 따라 상기 현재 블록을 디코딩하는 단계를 더 포함한다.

[0011] 일 실시예에 따르면, 비디오 디코딩 방법은 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계를 포함한다. 상기 방법은 현재 블록에 대한 상기 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색하는 단계를 더 포함한다. 상기 방법은 상기 현재 블록에 대해 검색된 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지 판정하는 단계를 더 포함한다. 상기 방법은 상기 현재 블록에 대한 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지의 판정에 기반하여 최대 IBC(intra block copy) 후보 수를 설정하는 단계를 더 포함한다.

[0012] 일 실시예에 따르면, 비디오 디코딩을 위한 비디오 디코딩 장치는 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하도록 구성된 처리 회로를 포함한다. 상기 처리 회로는 추가로, 상기 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하도록 구성된다. 상기 처리 회로는 추가로, 상기 미리 결정된 조건에 기반하여 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보의 수에 기반하여, 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하도록 구성된다. 상기 처리 회로는 추가로, 벡터 예측자로 상기 후보 리스트를 구성하고, 상기 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 상기 후보 리스트에서 벡터 예측자를 검색하도록 구성된다. 상기 처리 회로는 추가로, 상기 검색된 벡터 예측자에 따라 상기 현재 블록을 디코딩하도록 구성된다.

[0013] 일 실시예에 따르면, 비디오 디코딩을 위한 비디오 디코더 장치는 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하도록 구성된 처리 회로를 포함한다. 상기 처리 회로는 추가로, 현재 블록에 대한 상기 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색하도록 구성된다. 상기 처리 회로는 추가로, 상기 현재 블록에 대해 검색된 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지 판정하도록 구성된다. 상기 처리 회로는 추가로, 상기 현재 블록에 대한 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지의 판정에 기반하여, 최대 IBC(intra block copy) 후보 수를 설정하도록 구성된다.

[0014] 컴퓨터가 판독가능한 비일시적 매체는 명령을 저장하며, 상기 명령은 비디오 디코더의 프로세서에 의해 실행될 때, 상기 프로세서가 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계를 포함하는 방법을 실행하게 한다. 상기 방법은 상기 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계를 포함한다. 상기 방법은 상기 미리 결정된 조건에 기반하여 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보의 수에 기반하여 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계를 포함한다. 상기 방법은 벡터 예측자를 사용하여 후보 리스트를 구성하는 단계를 포함한다. 상기 방법은 상기 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 상기 후보 리스트에서 벡터 예측자를 검색하는 단계를 포함한다. 상기 방법은 상기 검색된 벡터 예측자에 따라 상기 현재 블록을 디코딩하는 단계를 더 포함한다.

[0015] 컴퓨터가 판독가능한 비일시적 매체는 명령을 저장하며, 상기 명령이 비디오 디코더의 프로세서에 의해 실행될 때 상기 프로세서가 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계를 포함하는 방법을 실행하게 한다. 상기 방법은 현재 블록에 대한 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색하는 단계를 더 포함한다. 상기 방법은 상기 현재 블록에 대해 검색된 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지 판정하는 단계를 더 포함한다. 상기 방법은 상기 현재 블록에 대한 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지의 판정에 기반하여 최대 IBC(intra block copy) 후보 수를 설정하는 단계를 더 포함한다.

도면의 간단한 설명

[0016] 개시된 주제의 추가 특징, 특성 및 다양한 이점은 다음의 상세한 설명 및 첨부 도면으로부터 더욱 명백해질 것이다.

도 1은 일 예에서 현재 블록 및 그 주변 공간 병합 후보의 개략도이다.

도 2는 일 실시예에 따른 통신 시스템(200)의 간략화된 블록도의 개략도이다.

도 3은 일 실시예에 따른 통신 시스템(300)의 간략화된 블록도의 개략도이다.

도 4는 일 실시예에 따른 디코더의 단순화된 블록도의 개략도이다.

도 5는 일 실시예에 따른 인코더의 단순화된 블록도의 개략도이다.

도 6은 다른 실시예에 따른 인코더의 블록도를 도시한다.

도 7은 다른 실시예에 따른 디코더의 블록도를 도시한다.

도 8은 일 실시예에 따른 인트라 화상 블록 보상의 개략도이다.

도 9a 내지 도 9d는 일 실시예에 따른 하나의 코딩 트리 유닛(CTU) 크기 탐색 범위를 갖는 인트라 화상 블록 보상의 개략도이다.

도 10a 내지 도 10d는 실시예들에 따른 버퍼가 어떻게 업데이트되는지의 개략도이다.

도 11a는 이력 기반 MV 예측(HMVP) 버퍼에 대한 디코딩 흐름도의 예시이다.

도 11b는 HMVP 버퍼를 업데이트하는 개략도이다.

도 12는 일 실시예에 따른 예시적인 디코딩 프로세스의 예시이다.

도 13은 일 실시예에 따른 예시적인 디코딩 프로세스의 예시이다.

도 14는 본 개시의 실시예에 따른 컴퓨터 시스템의 개략도이다.

발명을 실시하기 위한 구체적인 내용

[0017] 도 2는 본 개시의 실시예에 따른 통신 시스템(200)의 간략화된 블록도를 예시한다. 통신 시스템(200)은 예를 들어 네트워크(250)를 통해 서로 통신할 수 있는 복수의 단말 디바이스를 포함한다. 예를 들어, 통신 시스템(200)은 네트워크(250)를 통해 상호 연결된 제1 쌍의 단말 디바이스(210, 220)를 포함한다. 도 2의 예에서, 제1 쌍의 단말 디바이스(210, 220)는 데이터의 단방향 전송을 수행한다. 예를 들어, 단말 디바이스(210)는 네트워크(250)를 통해 다른 단말 디바이스(220)로 전송하기 위해 비디오 데이터(예: 단말 디바이스(210)에 의해 캡처된 비디오 화상의 스트림)를 코딩할 수 있다. 인코딩된 비디오 데이터는 하나 이상의 코딩된 비디오 비트스트림의 형태로 전송될 수 있다. 단말 디바이스(220)는 네트워크(250)로부터 코딩된 비디오 데이터를 수신하고, 코딩된 비디오 데이터를 디코딩하여 비디오 화상을 복구하고, 복구된 비디오 데이터에 따라 비디오 화상을 디스플레이할 수 있다. 단방향 데이터 전송은 미디어 서빙 애플리케이션 등에서 일반적일 수 있다.

[0018] 다른 예에서, 통신 시스템(200)은 예를 들어 화상 회의 동안 발생할 수 있는 코딩된 비디오 데이터의 양방향 전송을 수행하는 제2 쌍의 단말 디바이스(230, 240)를 포함한다. 데이터의 양방향 전송을 위해, 일 예에서 단말 디바이스(230 및 240)의 각 단말 디바이스는 네트워크(250)를 통해 단말 디바이스(230, 240) 중 다른 단말 디바이스로의 전송을 위해 비디오 데이터(예: 단말 디바이스에 의해 캡처된 비디오 화상의 스트림)를 코딩할 수 있다. 단말 디바이스(230, 240)의 각 단말 디바이스는 또한 단말 디바이스(230, 240)의 다른 단말 디바이스로부터 전송된 코딩된 비디오 데이터를 수신하고, 코딩된 비디오 데이터를 디코딩하여 비디오를 복구할 수 있으며, 복구된 비디오 데이터에 따라 액세스 가능한 디스플레이 디바이스에 비디오 화상을 디스플레이할 수 있다.

[0019] 도 2의 예에서, 단말 디바이스(210, 220, 230, 240)는 서버, 퍼스널 컴퓨터 및 스마트 폰으로 예시될 수 있지만, 본 개시의 원리는 이에 제한되지 않을 수 있다. 본 개시의 실시예는 랩톱 컴퓨터, 태블릿 컴퓨터, 미디어 플레이어 및/또는 전용 화상 회의 장비를 사용한 애플리케이션을 찾는다. 네트워크(250)는 예를 들어 유선(와이어드(wired)) 및/또는 무선 통신 네트워크를 포함하는 단말 디바이스(210, 220, 230, 240) 사이에서 코딩된 비디오 데이터를 전달하는 임의의 수의 네트워크를 나타낸다. 통신 네트워크(250)는 회선 교환(circuit-switched) 및/또는 패킷 교환(packet-switched) 채널에서 데이터를 교환할 수 있다. 대표적인 네트워크는 통신 네트워크, 근거리 네트워크, 광역 네트워크 및/또는 인터넷을 포함한다. 본 논의의 목적을 위해, 네트워크(250)의 아키텍처 및 토폴로지는 이하에서 설명되지 않는 한 본 개시의 작동에 중요하지 않을 수 있다.

[0020] 도 3은 개시된 주제에 대한 애플리케이션에 대한 예로서, 스트리밍 환경에서 비디오 인코더 및 비디오 디코더의 배치를 예시한다. 개시된 주제는 예를 들어 화상 회의, 디지털 TV, 그리고 CD, DVD, 메모리 스틱 등을 포함하는 디지털 미디어에 압축된 비디오를 저장하는 것 등을 포함하는 다른 비디오 가능 애플리케이션에 동일하게 적용될 수 있다.

[0021] 스트리밍 시스템은 비디오 소스(301), 예를 들어 압축되지 않은 비디오 화상의 스트림(302)을 생성하는 디지털 카메라를 포함할 수 있는 캡처 서브시스템(313)을 포함할 수 있다. 일 예에서, 비디오 화상의 스트림(302)은 디지털 카메라에 의해 촬영된 샘플을 포함한다. 인코딩된 비디오 데이터(304)(또는 코딩된 비디오 비트스트림)와 비교할 때 높은 데이터 볼륨(volume)을 강조하기 위해 굵은 선으로 묘사된 비디오 화상의 스트림(302)은, 비디오 소스(301)에 결합된 비디오 인코더(303)를 포함하는 전자 디바이스(320)에 의해 처리될 수 있다. 비디오 인코더(303)는 하드웨어, 소프트웨어, 또는 이들의 조합을 포함하여 아래에서 더 상세히 설명되는 바와 같이 개시된 주제의 측면을 가능하게 하거나 구현할 수 있다. 비디오 화상의 스트림(302)과 비교할 때 더 낮은 데이터 볼륨을 강조하기 위해 가는 선으로 묘사된 인코딩된 비디오 데이터(304)(또는 인코딩된 비디오 비트스트림(304))는 향후 사용을 위해 스트리밍 서버(305)에 저장될 수 있다. 도 3의 클라이언트 서브시스템(306 및 308)과 같은 하나 이상의 스트리밍 클라이언트 서브시스템은, 인코딩된 비디오 데이터(304)의 카피(307 및 309)를 검색하기(retrieve) 위해 스트리밍 서버(305)에 액세스할 수 있다. 클라이언트 서브시스템(306)은 예를 들어 전자 디바이스(330)에서 비디오 디코더(310)를 포함할 수 있다. 비디오 디코더(310)는 인코딩된 비디오 데이터의 인커밍

카피(307)를 디코딩하고, 디스플레이(312)(예: 디스플레이 스크린) 또는 다른 렌더링 디바이스(묘사되지 않음) 상에서 렌더링될 수 있다. 일부 스트리밍 시스템에서, 인코딩된 비디오 데이터(304, 307 및 309)(예: 비디오 비트스트림)은 특정 비디오 코딩/압축 표준에 따라 인코딩될 수 있다. 이러한 표준의 예로는 ITU-T 권장 사항 H.265가 있다. 일 예에서, 개발 중인 비디오 코딩 표준은 비공식적으로 VVC(Versatile Video Coding)로 알려져 있다. 개시된 주제는 VVC의 콘텍스트(context)에서 사용될 수 있다.

[0022] 전자 디바이스(320, 330)는 다른 컴포넌트(도시되지 않음)를 포함할 수 있음에 유의한다. 예를 들어, 전자 디바이스(320)는 비디오 디코더(도시되지 않음)를 포함할 수 있고, 전자 디바이스(330)는 비디오 인코더(도시되지 않음)도 포함할 수 있다.

[0023] 도 4는 본 개시의 일 실시예에 따른 비디오 디코더(410)의 블록도를 도시한다. 비디오 디코더(410)는 전자 디바이스(430)에 포함될 수 있다. 전자 디바이스(430)는 수신기(431)(예: 수신 회로)를 포함할 수 있다. 비디오 디코더(410)는 도 3의 예에서의 비디오 디코더(310) 대신에 사용될 수 있다.

[0024] 수신기(431)는 비디오 디코더(410)에 의해 디코딩될 하나 이상의 코딩된 비디오 시퀀스를 수신할 수 있고; 동일 또는 다른 실시예에서, 한 번에 하나의 코딩된 비디오 시퀀스로, 각각의 코딩된 비디오 시퀀스의 디코딩은 다른 코딩된 비디오 시퀀스와 독립적이다. 코딩된 비디오 시퀀스는 인코딩된 비디오 데이터를 저장하는 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는 채널(401)로부터 수신될 수 있다. 수신기(431)는 각각의 사용 엔티티(도시되지 않음)로 전달될 수 있는 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림과 함께 인코딩된 비디오 데이터를 수신할 수 있다. 수신기(431)는 코딩된 비디오 시퀀스를 다른 데이터로부터 분리할 수 있다. 네트워크 지터(jitter)를 방지하기 위해 버퍼 메모리(415)가 수신기(431)와 엔트로피 디코더/파서(parser)(420)(이하 "파서(420)") 사이에 결합될 수 있다. 특정 애플리케이션에서, 버퍼 메모리(415)는 비디오 디코더(410)의 일부이다. 다른 경우에, 비디오 디코더(410)(도시되지 않음) 외부에 있을 수 있다. 또 다른 경우에, 예를 들어 네트워크 지터를 방지하기 위해 비디오 디코더(410) 외부에 버퍼 메모리(도시되지 않음)가 있을 수 있고, 예를 들어, 플레이아웃 타이밍(playout timing)을 핸들링하기 위해 비디오 디코더(410) 내부에 또 다른 버퍼 메모리(415)가 있을 수 있다. 수신기(431)가 충분한 대역폭과 제어 가능성(controllability)을 가지는 저장/전달 디바이스 또는 동시동기식(isosynchronous) 네트워크로부터 데이터를 수신할 때, 버퍼 메모리(415)는 필요하지 않거나 작을 수 있다. 인터넷과 같은 최선형(best effort) 패킷 네트워크에서 사용하기 위해, 버퍼 메모리(415)가 필요할 수 있고, 비교적 클 수 있으며, 유리하게는 적응적 크기(adaptive size)를 가질 수 있고, 그리고 운영 체제 또는 비디오 디코더(410)의 외부의 유사한 엘리먼트(도시지 않음)에서 적어도 부분적으로 구현될 수 있다.

[0025] 비디오 디코더(410)는 코딩된 비디오 시퀀스로부터 심볼(421)을 재구성하기 위해 파서(420)를 포함할 수 있다. 이러한 심볼의 카테고리는 비디오 디코더(410)의 작동을 관리하는 데 사용되는 정보, 및 도 4에 도시된 바와 같이 전자 디바이스(430)에 결합될 수 있지만 전자 디바이스(430)의 필수 부분이 아닌 렌더(render) 디바이스(412)(예: 디스플레이 스크린)와 같은 렌더링 디바이스를 제어하기 위한 감재적인 정보를 포함한다. 렌더링 디바이스(들)에 대한 제어 정보는 SEI(Supplemental Enhancement Information) 메시지 또는 VUI(Video Usability Information) 파라미터 세트 프래그먼트(fragment)(도시되지 않음)의 형태일 수 있다. 파서(420)는 수신된 코딩된 비디오 시퀀스를 파싱/엔트로피 디코딩할 수 있다. 코딩된 비디오 시퀀스의 코딩은 비디오 코딩 기술 또는 표준을 따를 수 있고, 가변 길이 코딩, 허프만 코딩(Huffman coding), 콘텍스트 감도(context sensitivity)가 있거나 없는 산술 코딩 등을 포함하는 다양한 원리를 따를 수 있다. 파서(420)는 그룹에 대응하는 적어도 하나의 파라미터에 기반하여, 비디오 디코더에서 픽셀의 서브그룹 중 적어도 하나에 대한 서브그룹 파라미터의 세트를 코딩된 비디오 시퀀스로부터 추출할 수도 있다. 서브그룹은 화상 그룹(Groups of Picture, GOP), 화상, 타일(tile), 슬라이스(slice), 매크로블록(macroblock), 코딩 유닛(Coding Unit, CU), 블록, 변환 유닛(blocks, Transform Unit, TU), 예측 유닛(Prediction Unit, PU) 등을 포함할 수 있다. 파서(420)는 또한 변환 계수, 양자화기 파라미터 값, 모션 벡터 등과 같은 정보를 코딩된 비디오 시퀀스로부터 추출할 수 있다.

[0026] 파서(420)는 버퍼 메모리(415)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩/파싱 작동을 수행하여 심볼(421)을 생성할 수 있다.

[0027] 심볼(421)의 재구성은 코딩된 비디오 화상의 유형 또는 그 일부(예: 인터 및 인트라 화상, 인터 및 인트라 블록), 및 기타 팩터(factor)에 따라 다수의 상이한 유닛을 포함할 수 있다. 어떤 유닛이 관련되는지, 그리고 어떻게 제어되는지는, 파서(420)에 의해 코딩된 비디오 시퀀스로부터 파싱된 서브그룹 제어 정보에 의해 제어될 수 있다. 파서(420)와 다수의 유닛 사이의 이러한 서브그룹 제어 정보의 흐름은 아래에 명확성을 위해 도시되지

않는다.

[0028] 이미 언급된 기능 블록 외에도, 비디오 디코더(410)는 개념적으로 아래에서 설명되는 바와 같이 다수의 기능 유닛으로 세분될 수 있다. 상업적 제약(constraint) 하에서 작동하는 실제 구현에서, 이러한 유닛 중 많은 부분이 서로 밀접하게 상호 작용하고 적어도 부분적으로 서로 통합될 수 있다. 그러나, 개시된 주제를 설명하기 위해 아래의 기능 유닛으로의 개념적 세분화가 적절하다.

[0029] 제1 유닛은 스케일러/역 변환(scaler/inverse transform) 유닛(451)이다. 스케일러/역 변환 유닛(451)은 사용할 변환, 블록 크기, 양자화 팩터, 양자화 스케일링 행렬 등을 포함하는 제어 정보뿐만 아니라 양자화된 변환 계수를 파서(420)로부터 심볼(들)(421)로서 수신한다. 스케일러/역 변환 유닛(451)은 애그리게이터(aggregator)(455)에 입력될 수 있는 샘플 값을 포함하는 블록을 출력할 수 있다.

[0030] 일부 경우에, 스케일러/역 변환(451)의 출력 샘플은 인트라 코딩된 블록에 속할 수 있으며; 즉, 이전에 재구성된 화상으로부터의 예측 정보를 사용하지 않지만 현재 화상의 이전에 재구성된 부분으로부터의 예측 정보를 사용할 수 있는 블록이다. 이러한 예측 정보는 인트라 화상 예측 유닛(452)에 의해 제공될 수 있다. 일부 경우에, 인트라 화상 예측 유닛(452)은 현재 화상 버퍼(458)로부터 페치된(fetched) 주변의 이미 재구성된 정보를 사용하여, 재구성 중인 블록과 동일한 크기 및 형태의 블록을 생성한다. 현재 화상 버퍼(458)는 예를 들어 부분적으로 재구성된 현재 화상 및/또는 완전히 재구성된 현재 화상을 버퍼링한다. 애그리게이터(455)는 일부 경우에, 샘플 단위로 인트라 예측 유닛(452)이 생성한 예측 정보를 스케일러/역 변환 유닛(451)에 의해 제공되는 출력 샘플 정보에 추가한다.

[0031] 다른 경우에, 스케일러/역 변환 유닛(451)의 출력 샘플은 인터 코딩되고 잠재적으로 모션 보상된 블록에 속할 수 있다. 그러한 경우에, 모션 보상 예측 유닛(453)은 예측에 사용되는 샘플들을 페치하기 위해 참조 화상 메모리(457)에 액세스할 수 있다. 블록에 속하는 심볼(421)에 따라 페치된 샘플을 모션 보상한 후, 이러한 샘플은 출력 샘플 정보를 생성하기 위해 애그리게이터(455)에 의해 스케일러/역 변환 유닛(451)의 출력에 추가될 수 있다(이 경우 잔차 샘플 또는 잔차 신호라고 함). 모션 보상 예측 유닛(453)이 예측 샘플을 페치하는 참조 화상 메모리(457) 내의 어드레스는 모션 벡터에 의해 제어될 수 있으며, 모션 벡터는 예를 들어 X, Y 및 참조 화상 컴포넌트를 가질 수 있는 심볼(421)의 형태로 모션 보상 예측 유닛(453)에 대해 이용 가능하다. 모션 보상은 또한 서브-샘플 정확한(exact) 모션 벡터가 사용 중일 때, 참조 화상 메모리(457)로부터 페치된 샘플 값의 보간, 모션 벡터 예측 메커니즘 등을 포함할 수 있다.

[0032] 애그리게이터(455)의 출력 샘플은 루프 필터 유닛(456)에서 다양한 루프 필터링 기술의 대상이 될 수 있다. 비디오 압축 기술은 코딩된 비디오 시퀀스(코딩된 비디오 비트스트림이라고도 함)에 포함된 파라미터에 의해 제어되고, 파서(420)로부터의 심볼(421)로서 루프 필터 유닛(456)에 이용 가능하게 되는 인루프 필터(in-loop filter) 기술을 포함할 수 있지만, 또한 코딩된 화상 또는 코딩된 비디오 시퀀스의 이전(디코딩 순서에서) 부분의 디코딩 동안 획득된 메타 정보에 응답할 수 있을 뿐만 아니라, 또한 이전에 재구성 및 루프 필터링된 샘플 값에 응답할 수 있다.

[0033] 루프 필터 유닛(456)의 출력은 렌더 디바이스(412)에 출력될 수 있을 뿐만 아니라 미래의 화상간 예측에 사용하기 위해 참조 화상 메모리(457)에 저장될 수 있는 샘플 스트림일 수 있다.

[0034] 완전히 재구성된 특정 코딩된 화상은 미래 예측을 위한 참조 화상으로 사용될 수 있다. 예를 들어, 현재 화상에 대응하는 코딩된 화상이 완전히 재구성되고 코딩된 화상이 (예: 파서(420)에 의해) 참조 화상으로 식별되면, 현재 화상 버퍼(458)는 참조 화상 메모리(457)의 메모리의 일부가 될 수 있으며, 새로운(fresh) 현재 화상 버퍼가 다음의 코딩된 화상의 재구성을 시작하기 전에 재할당될 수 있다.

[0035] 비디오 디코더(410)는 ITU-T Rec. H.265와 같은 표준에서의 미리 결정된 비디오 압축 기술에 따라 디코딩 작동을 수행할 수 있다. 코딩된 비디오 시퀀스가 비디오 압축 기술 또는 표준의 신택스(syntax)와 비디오 압축 기술 또는 표준에 문서화된 프로필(profile)을 모두 준수한다는 의미에서, 코딩된 비디오 시퀀스는 사용되는 비디오 압축 기술 또는 표준에 의해 지정된 신택스를 따를 수 있다. 구체적으로, 프로필은 비디오 압축 기술 또는 표준에서 사용 가능한 모든 도구로부터 해당 프로필에서 사용할 수 있는 유일한 도구로 특정 도구를 선택할 수 있다. 또한, 규정 준수를 위해 필요한 것은 코딩된 비디오 시퀀스의 복잡성이 비디오 압축 기술 또는 표준의 레벨에 의해 정의된 범위 내에 있어야 한다는 것이다. 일부 경우에, 레벨이 최대 화상 크기, 최대 프레임 레이트, 최대 재구성 샘플 레이트(예를 들어, 초당 메가샘플(megasample)로 측정됨), 최대 참조 화상 크기 등을 제한한다. 레벨에 의해 설정된 한계는, 일부 경우에, 코딩된 비디오 시퀀스에서 시그널링되는 HRD(Hypothetical

Reference Decoder) 사양 및 HRD 버퍼 관리를 위한 메타데이터를 통해 추가로 제한될 수 있다.

[0036] 일 실시예에서, 수신기(431)는 인코딩된 비디오와 함께 추가적인(중복된(redundant)) 데이터를 수신할 수 있다. 추가 데이터는 코딩된 비디오 시퀀스(들)의 일부로서 포함될 수 있다. 추가 데이터는 비디오 디코더(410)에 의해 데이터를 적절하게 디코딩하거나 및/또는 원래의 비디오 데이터를 보다 정확하게 재구성하기 위해 사용될 수 있다. 추가 데이터는 예를 들어 시간, 공간 또는 신호 잡음비(signal noise ratio, SNR) 향상 계층(enhancement layer), 중복된 슬라이스, 중복된 화상, 순방향 에러 정정 코드 등의 형태일 수 있다.

[0037] 도 5는 본 개시의 일 실시예에 따른 비디오 인코더(503)의 블록도를 도시한다. 비디오 인코더(503)는 전자 디바이스(520)에 포함된다. 전자 디바이스(520)는 송신기(540)(예: 송신 회로)를 포함한다. 비디오 인코더(503)는 도 3의 예에서의 비디오 인코더(303) 대신에 사용될 수 있다.

[0038] 비디오 인코더(503)는 비디오 인코더(503)에 의해 코딩될 비디오 이미지(들)를 캡처할 수 있는 비디오 소스(501)(이는 도 5의 예에서 전자 디바이스(520)의 일부가 아님)로부터 비디오 샘플을 수신할 수 있다. 다른 예에서, 비디오 소스(501)는 전자 디바이스(520)의 일부이다.

[0039] 비디오 소스(501)는 임의의 적절한 비트 깊이(예를 들어, 8비트, 10비트, 12비트, ...), 임의의 색상 공간(예: BT.601 Y CrCB, RGB,...) 및 임의의 적절한 샘플링 구조(예를 들어, Y CrCb 4:2:0, Y CrCb 4:4:4)일 수 있는 디지털 비디오 샘플 스트림의 형태로 비디오 인코더(503)에 의해 코딩될 소스 비디오 시퀀스를 제공할 수 있다. 미디어 서빙(media serving) 시스템에서, 비디오 소스(501)는 미리 준비된 비디오를 저장하는 저장 디바이스(storage device)일 수 있다. 화상 회의 시스템에서, 비디오 소스(501)는 비디오 시퀀스로서 로컬 이미지 정보를 캡처하는 카메라일 수 있다. 비디오 데이터는 순서대로 보일 때 모션을 부여하는 복수의 개별 화상으로서 제공될 수 있다. 화상 자체는 픽셀의 공간 어레이로 구성될 수 있으며, 여기서 각 픽셀은 사용 중인 샘플링 구조, 색 공간 등에 따라 하나 이상의 샘플을 포함할 수 있다. 당업자는 픽셀과 샘플 간의 관계를 쉽게 이해할 수 있다. 아래 설명은 샘플에 중점을 둔다.

[0040] 일 실시예에 따르면, 비디오 인코더(503)는 실시간으로 또는 애플리케이션에 의해 요구되는 임의의 다른 시간 제약 하에, 소스 비디오 시퀀스의 화상을 코딩된 비디오 시퀀스(543)로 코딩 및 압축할 수 있다. 적절한 코딩 속도를 적용하는 것은 컨트롤러(550)의 기능 중 하나이다. 일부 실시예에서, 컨트롤러(550)는 후술되는 바와 같이 다른 기능 유닛을 제어하고, 다른 기능 유닛에 기능적으로 결합된다. 커플링(coupling)은 명확성을 위해 묘사되지 않는다. 컨트롤러(550)에 의해 설정된 파라미터는 레이트 제어 관련 파라미터(화상 스킵(skip), 양자화기(quantizer), 레이트-왜곡 최적화 기술의 람다(lambda) 값,...), 화상 크기, GOP(group of picture) 레이아웃, 최대 모션 벡터 검색 범위 및 등을 포함할 수 있다. 컨트롤러(550)는 특정 시스템 설계에 최적화된 비디오 인코더(503)에 속하는 다른 적절한 기능을 갖도록 구성될 수 있다.

[0041] 일부 실시예에서, 비디오 인코더(503)는 코딩 루프(coding loop)에서 작동하도록 구성된다. 지나치게 단순화된 설명으로서, 일 예에서, 코딩 루프는 소스 코더(530)(예: 코딩될 입력 화상에 기반하여 심볼 스트림과 같은 심볼 및 참조 화상(들)을 생성하는 것을 담당) 및 비디오 인코더(503)에 내장된 (로컬) 디코더(533)를 포함한다. 디코더(533)는 (원격) 디코더가 생성하는 것과 유사한 방식으로 샘플 데이터를 생성하기 위해 심볼을 재구성한다(개시된 주제에서 고려되는 비디오 압축 기술에서 심볼과 코딩된 비디오 비트스트림 사이의 압축이 무손실이기 때문에). 재구성된 샘플 스트림(샘플 데이터)은 참조 화상 메모리(534)에 입력된다. 심볼 스트림의 디코딩이 디코더 위치(로컬 또는 원격)에 관계없이 정확한 비트 결과를 가져오므로, 참조 화상 메모리(534)의 콘텐츠도 로컬 인코더와 원격 인코더 사이에서 비트 정확하다. 다시 말해서, 인코더의 예측 부분은 디코딩 동안 예측을 사용할 때 디코더가 "보는(see)" 것과 정확히 동일한 샘플 값을 참조 화상 샘플로 "본다". 참조 화상 동시성(synchronicity)의 기본 원리(예를 들어 채널 에러로 인해 동시성이 유지될 수 없으면 결과적으로 드리프트(drift)가 발생)는 일부 관련 기술에서도 사용된다.

[0042] "로컬" 디코더(533)의 작동은 비디오 디코더(410)와 같은 "원격" 디코더의 작동과 동일할 수 있으며, 이는 도 4와 관련하여 위에서 이미 상세하게 설명되었다. 또한 도 4를 간략히 참조하면, 그러나 심볼이 이용가능하고, 엔트로피 코더(545) 및 파서(420)에 의한 코딩된 비디오 시퀀스에 대한 심볼의 인코딩/디코딩이 무손실일 수 있기 때문에, 버퍼 메모리(415)를 포함하는, 비디오 디코더(410)의 엔트로피 디코딩 및 파서(420)는 로컬 디코더(533)에서 완전히 구현되지 않을 수 있다.

[0043] 이 시점에서 관찰할 수 있는 사항은 디코더에 존재하는 파싱/엔트로피 디코딩을 제외한 임의의 디코더 기술이 대응하는 인코더에서, 반드시 실질적으로 동일한 기능 형태로 존재할 필요가 있다는 것이다. 이러한 이유로, 개

시된 주제는 디코더 작동에 초점을 맞춘다. 인코더 기술에 대한 설명은 포괄적으로 설명된 디코더 기술의 반대이므로 축약될 수 있다. 특정 에어리어에서만 더 자세한 설명이 필요하며 아래에서 제공된다.

[0044] 작동 동안, 일부 예에서, 소스 코더(530)는 "참조 화상"로서 지정된 비디오 시퀀스로부터의 하나 이상의 이전에 코딩된 화상을 참조하여 예측적으로 입력 화상을 코딩하는, 모션 보상된 예측 코딩을 수행할 수도 있다. 이러한 방식으로, 코딩 엔진(532)은 입력 화상의 픽셀 블록과 입력 화상에 대한 예측 참조(들)로서 선택될 수 있는 참조 화상(들)의 픽셀 블록 간의 차이를 코딩한다.

[0045] 로컬 비디오 디코더(533)는 소스 코더(530)에 의해 생성된 심볼에 기반하여, 참조 화상으로 지정될 수 있는 화상의 코딩된 비디오 데이터를 디코딩할 수 있다. 코딩 엔진(532)의 작동은 유리하게는 손실 프로세스(lossy process)일 수 있다. 코딩된 비디오 데이터가 비디오 디코더(도 5에 도시되지 않음)에서 디코딩될 수 있을 때, 재구성된 비디오 시퀀스는 일반적으로 일부 에러가 있는 소스 비디오 시퀀스의 복제물(replica)일 수 있다. 로컬 비디오 디코더(533)는 참조 화상에 대해 비디오 디코더에 의해 수행될 수 있는 디코딩 프로세스를 복제하고, 재구성된 참조 화상이 참조 화상 캐시(534)에 저장되게 할 수 있다. 이러한 방식으로, 비디오 인코더(503)는 원단(far-end) 비디오 디코더에 의해 획득될(전송 에러 없음) 재구성된 참조 화상으로서 공통 콘텐츠를 갖는 재구성된 참조 화상의 카피를 국부적으로 저장할 수도 있다.

[0046] 예측자(predictor)(535)는 코딩 엔진(532)에 대한 예측 탐색을 수행할 수 있다. 즉, 코딩될 신규 화상에 대해, 예측자(535)는 신규 화상에 대한 적절한 예측 참조 역할을 할 수 있는, 샘플 데이터(후보 참조 픽셀 블록으로서) 또는 참조 화상 모션 벡터, 블록 모양 등과 같은 특정 메타데이터에 대해 참조 화상 메모리(534)를 검색할 수 있다. 예측자(535)는 적절한 예측 참조를 찾기 위해 샘플 블록 단위로 작동할 수 있다. 일부 경우에, 예측자(535)에 의해 획득된 탐색 결과에 의해 결정된 바와 같이, 입력 화상은 참조 화상 메모리(534)에 저장된 다수의 참조 화상으로부터 도출된 예측 참조를 가질 수 있다.

[0047] 컨트롤러(550)는 예를 들어 비디오 데이터를 인코딩하는 데 사용되는 파라미터 및 서브그룹 파라미터의 설정을 포함하는, 소스 코더(530)의 코딩 작동을 관리할 수 있다.

[0048] 전술한 모든 기능 유닛의 출력은 엔트로피 코더(545)에서 엔트로피 코딩될 수 있다. 엔트로피 코더(545)는 허프만 코딩, 가변 길이 코딩, 산술 코딩 등과 같은 기술에 따라 심볼을 무손실 압축함으로써 다양한 기능 유닛에 의해 생성된 심볼을 코딩된 비디오 시퀀스로 변환한다.

[0049] 송신기(540)는 엔트로피 코더(545)에 의해 생성된 코딩된 비디오 시퀀스(들)를 버퍼링하여 통신 채널(560)을 통한 전송을 준비할 수 있으며, 이는 인코딩된 비디오 데이터를 저장하는 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있다. 송신기(540)는 비디오 코더(503)로부터의 코딩된 비디오 데이터를 전송될 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림(소스는 도시되지 않음)과 병합할 수 있다.

[0050] 컨트롤러(550)는 비디오 인코더(503)의 작동을 관리할 수 있다. 코딩 동안, 컨트롤러(550)는 각각의 화상에 적용될 수 있는 코딩 기술에 영향을 미칠 수 있는 특정 코딩된 화상 유형을 각각의 코딩된 화상에 할당할 수 있다. 예를 들어, 화상은 종종 다음 화상 유형 중 하나로 할당될 수 있다:

[0051] 인트라 화상(Intra Picture)(I 화상)은 시퀀스에서의 다른 화상을 사용하지 않고, 예측 소스로서 코딩 및 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱은 예를 들어 독립 디코더 리프레시(Independent Decoder Refresh)("IDR") 화상을 포함하는 다양한 유형의 인트라 화상을 허용한다. 당업자는 I 화상의 이러한 변형 및 각각의 애플리케이션 및 특징을 알고 있다.

[0052] 예측 화상(predictive picture)(P 화상)은 각 블록의 샘플 값을 예측하기 위해 최대 하나의 모션 벡터 및 참조 인덱스를 사용하는 인트라 예측 또는 인터 예측을 사용하여 인코딩 및 디코딩될 수 있는 것일 수 있다.

[0053] 양방향 예측 화상(bi-directionally predictive picture)(B Picture)은 각 블록의 샘플 값을 예측하기 위해 최대 2개의 모션 벡터 및 참조 인덱스를 사용하는 인트라 예측 또는 인터 예측을 사용하여 인코딩 및 디코딩될 수 있는 것일 수 있다. 유사하게, 다수의 예측 화상은 단일 블록의 재구성을 위해 2개 이상의 참조 화상 및 관련 메타데이터를 사용할 수 있다.

[0054] 소스 화상은 일반적으로 복수의 샘플 블록(예를 들어, 각각 4×4, 8×8, 4×8 또는 16×16 샘플의 블록)으로 공간적으로 세분화되고 블록 단위로 코딩될 수 있다. 블록은 블록의 각각의 화상에 적용된 코딩 할당에 의해 결정된 바와 같이 다른(이미 코딩된) 블록을 참조하여 예측적으로 코딩될 수 있다. 예를 들어, I 화상의 블록은 비예측적으로 코딩될 수 있거나 동일한 화상의 이미 코딩된 블록을 참조하여 예측적으로 코딩될 수 있다(공간 예

측 또는 인트라 예측). P 화상의 픽셀 블록은 하나의 이전에 코딩된 참조 화상을 참조하여 공간적 예측을 통해 또는 시간적 예측을 통해 예측적으로 코딩될 수 있다. B 화상의 블록은 하나 또는 2개의 이전에 코딩된 참조 화상을 참조하여 공간적 예측을 통해 또는 시간적 예측을 통해 예측적으로 코딩될 수 있다.

[0055] 비디오 인코더(503)는 ITU-T Rec. H.265와 같은 미리 결정된 비디오 코딩 기술 또는 표준에 따라 코딩 작동을 수행할 수 있다. 그 작동에서, 비디오 인코더(503)는 입력 비디오 시퀀스에서 시간적 및 공간적 중복성을 이용하는 예측 코딩 작동을 포함하는 다양한 압축 작동을 수행할 수 있다. 따라서, 코딩된 비디오 데이터는 사용되는 비디오 코딩 기술 또는 표준에 의해 지정된 신택스를 따를 수도 있다.

[0056] 일 실시예에서, 송신기(540)는 인코딩된 비디오와 함께 추가 데이터를 전송할 수 있다. 소스 코더(530)는 코딩된 비디오 시퀀스의 일부로서 이러한 데이터를 포함할 수 있다. 추가 데이터는 시간/공간/SNR 향상 계층, 중복된 화상 및 슬라이스와 같은 다른 형태의 중복 데이터, SEI 메시지, VUI 파라미터 세트 프래그먼트 등을 포함할 수 있다.

[0057] 비디오는 시간적 시퀀스에서 복수의 소스 화상(비디오 화상)으로서 캡처될 수 있다. 인트라 화상 예측(종종 인트라 예측으로 축약됨)은 주어진 화상에서 공간적 상관을 사용하고, 인터 화상 예측은 화상 간의(시간적 또는 기타) 상관을 사용한다. 일 예에서, 현재 화상이라고 하는 인코딩/디코딩 중인 특정 화상이 블록으로 파티셔닝된다(partitioned). 현재 화상의 블록이 비디오의 이전에 코딩되고 여전히 버퍼링된 참조 화상의 참조 블록과 유사할 때, 현재 화상의 블록은 모션 벡터라고 하는 벡터에 의해 코딩될 수 있다. 모션 벡터는 참조 화상의 참조 블록을 가리키며, 다수의 참조 화상이 사용 중인 경우 참조 화상을 식별하는 3차원을 가질 수 있다.

[0058] 일부 실시예에서, 이중 예측(bi-prediction) 기술이 인터 화상 예측에서 사용될 수 있다. 이중 예측 기술에 따르면, 비디오의 현재 화상에 대해 디코딩 순서가 둘 다 우선인(그러나 각각 과거 및 미래일 수 있음) 제1 참조 화상 및 제2 참조 화상과 같은 2개의 참조 화상(디스플레이 순서에서 각각 과거 및 미래에 있을 수 있음)을 사용한다. 현재 화상의 블록은 제1 참조 화상의 제1 참조 블록을 가리키는 제1 모션 벡터 및 제2 참조 화상의 제2 참조 블록을 가리키는 제2 모션 벡터에 의해 코딩될 수 있다. 블록은 제1 참조 블록과 제2 참조 블록의 조합에 의해 예측될 수 있다.

[0059] 또한, 인코딩 효율을 향상시키기 위해 인터 화상 예측에 병합 모드 기술이 사용될 수 있다.

[0060] 본 개시의 일부 실시예에 따르면, 인터 화상 예측 및 인트라 화상 예측과 같은 예측이 블록 단위로 수행된다. 예를 들어, HEVC 표준에 따르면, 비디오 화상 시퀀스의 화상은 압축을 위해 CTU(Coding Tree Units)로 파티셔닝되며, 화상의 CTU는 64×64 픽셀, 32×32 픽셀 또는 16×16 픽셀과 같은 동일한 크기를 갖는다. 일반적으로, CTU는 1개의 루마(luma) CTB와 2개의 크로마(chroma) CTB인 3개의 코딩 트리 블록(coding tree block, CTB)을 포함한다. 각 CTU는 하나 또는 여러 개의 코딩 유닛(coding unit, CU)으로 분할된(split) 재귀적으로 쿼드트리(quadtree)일 수 있다. 예를 들어, 64×64 픽셀의 CTU는 64×64 픽셀의 하나의 CU, 또는 32×32 픽셀의 4개의 CU 또는 16×16 픽셀의 16개의 CU로 분할될 수 있다. 일 예에서, 각각의 CU는 인터 예측 유형 또는 인트라 예측 유형과 같은 CU에 대한 예측 유형을 결정하기 위해 분석된다. CU는 시간 및/또는 공간 예측 가능성에 따라 하나 이상의 예측 유닛(prediction unit, PU)으로 분할된다. 일반적으로, 각 PU는 루마 예측 블록(prediction block, PB)과 2개의 크로마 PB를 포함한다. 일 실시예에서, 코딩(인코딩/디코딩)에서의 예측 작동은 예측 블록 단위로 수행된다. 예측 블록의 예로서 루마 예측 블록을 사용하여, 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.

[0061] 도 6은 본 개시의 다른 실시예에 따른 비디오 인코더(603)의 다이어그램을 도시한다. 비디오 인코더(603)는 비디오 화상의 시퀀스에서 현재 비디오 화상 내의 샘플 값들의 처리 블록(예: 예측 블록)을 수신하고, 처리 블록을 코딩된 비디오 시퀀스의 일부인 코딩된 화상으로 인코딩하도록 구성된다. 일 예에서, 비디오 인코더(603)는 도 3의 예에서의 비디오 인코더(303) 대신에 사용된다.

[0062] HEVC 예에서, 비디오 인코더(603)는 8×8 샘플들의 예측 블록 등과 같은 처리 블록에 대한 샘플 값들의 행렬을 수신한다. 비디오 인코더(603)는 처리 블록이 예를 들어 레이트 왜곡 최적화를 사용하여 인트라 모드, 인터 모드, 또는 이중 예측 모드를 사용하여 가장 잘 코딩되었는지를 판정한다. 처리 블록이 인트라 모드로 코딩될 때, 비디오 인코더(603)는 인트라 예측 기술을 사용하여 처리 블록을 코딩된 화상으로 인코딩할 수도 있고; 처리 블록이 인터 모드 또는 이중 예측 모드로 코딩되어야 할 때, 비디오 인코더(603)는 인터 예측 또는 이중 예측 기술을 각각 사용하여 처리 블록을 코딩된 화상으로 인코딩할 수도 있다. 특정 비디오 코딩 기술에서, 병합 모드는 모션 벡터가 예측자 외부의 코딩된 모션 벡터 컴포넌트의 이점 없이 하나 이상의 모션 벡터 예측자로부터 유

도되는 인터 화상 예측 서브모드일 수 있다. 특정 다른 비디오 코딩 기술에서, 대상 블록(subject block)에 적용 가능한 모션 벡터 컴포넌트가 존재할 수 있다. 일 예에서, 비디오 인코더(603)는 처리 블록의 모드를 결정하기 위한 모드 결정 모듈(도시되지 않음)과 같은 다른 컴포넌트를 포함한다.

[0063] 도 6 예에서, 비디오 인코더(603)는 도 6에 도시된 바와 같이 함께 결합된, 인터 인코더(630), 인트라 인코더(622), 잔차 계산기(623), 스위치(626), 잔차 인코더(624), 일반 컨트롤러(general controller)(621), 및 엔트로피 인코더(625)를 포함한다.

[0064] 인터 인코더(630)는 현재 블록(예: 처리 블록)의 샘플을 수신하고, 블록을 참조 화상의 하나 이상의 참조 블록(예: 이전 화상 및 이후 화상의 블록)과 비교하며, 인터 예측 정보(예: 인터 인코딩 기술에 따른 중복 정보의 기술, 모션 벡터, 병합 모드 정보), 및 임의의 적절한 기술을 사용하는 인터 예측 정보에 기반한 인터 예측 결과(예: 예측 블록)를 계산한다. 일부 예에서, 참조 화상들은 인코딩된 비디오 정보에 기반하여 디코딩되는, 디코딩된 참조 화상들이다.

[0065] 인트라 인코더(622)는 현재 블록(예: 처리 블록)의 샘플을 수신하고, 일부 경우에 블록을 동일한 화상에서 이미 코딩된 블록과 비교하며, 변환 후에 양자화된 계수 그리고, 일부 경우에 또한 인트라 예측 정보(예: 하나 이상의 인트라 인코딩 기술에 따른 인트라 예측 방향 정보)를 생성하도록 구성된다. 일 예에서, 인트라 인코더(622)는 또한 인트라 예측 정보 및 동일한 화상에서의 참조 블록에 기반하여 인트라 예측 결과(예: 예측된 블록)를 계산한다.

[0066] 일반 컨트롤러(621)는 일반 제어 데이터를 결정하고, 일반 제어 데이터에 기반하여 비디오 인코더(603)의 다른 컴포넌트를 제어하도록 구성된다. 일 예에서, 일반 컨트롤러(621)는 블록의 모드를 결정하고, 그 모드에 기반하여 제어 신호를 스위치(626)에 제공한다. 예를 들어, 모드가 인트라 모드일 때, 일반 컨트롤러(621)는 스위치(626)를 제어하여 잔차 계산기(623)에서 사용할 인트라 모드 결과를 선택하고, 엔트로피 인코더(625)를 제어하여 인트라 모드 정보를 선택하며, 인트라 예측 정보를 비트스트림에 포함시키고; 모드가 인터 모드일 때, 일반 컨트롤러(621)는 스위치(626)를 제어하여 잔차 계산기(623)에서 사용할 인터 예측 결과를 선택하고, 엔트로피 인코더(625)를 제어하여 인터 예측 정보를 선택하며, 인터 예측 정보를 비트스트림에 포함시킨다.

[0067] 잔차 계산기(623)는 수신된 블록과 인트라 인코더(622) 또는 인터 인코더(630)로부터 선택된 예측 결과 사이의 차이(잔차 데이터)를 계산하도록 구성된다. 잔차 인코더(624)는 잔차 데이터에 기반하여 작동하여 잔차 데이터를 인코딩하여 변환 계수를 생성하도록 구성된다. 일 예에서, 잔차 인코더(624)는 잔차 데이터를 공간 도메인에서 주파수 도메인으로 변환하고, 변환 계수를 생성하도록 구성된다. 변환 계수는 그다음 양자화 처리를 거쳐 양자화된 변환 계수가 획득된다. 다양한 실시예에서, 비디오 인코더(603)는 또한 잔차 디코더(628)를 포함한다. 잔차 디코더(628)는 역 변환을 수행하고 디코딩된 잔차 데이터를 생성하도록 구성된다. 디코딩된 잔차 데이터는 인트라 인코더(622) 및 인터 인코더(630)에 의해 적절하게 사용될 수 있다. 예를 들어, 인터 인코더(630)는 디코딩된 잔차 데이터 및 인터 예측 정보에 기반하여 디코딩된 블록을 생성할 수 있고, 인트라 인코더(622)는 디코딩된 잔차 데이터 및 인트라 예측 정보에 기반하여 디코딩된 블록을 생성할 수 있다. 디코딩된 블록은 디코딩된 화상을 생성하도록 적절하게 처리되고, 디코딩된 화상은 메모리 회로(도시되지 않음)에 버퍼링되고 일부 예에서 참조 화상으로서 사용될 수 있다.

[0068] 엔트로피 인코더(625)는 인코딩된 블록을 포함하도록 비트스트림을 포맷하도록 구성된다. 엔트로피 인코더(625)는 HEVC 표준과 같은 적절한 표준에 따라 다양한 정보를 포함하도록 구성된다. 일 예에서, 엔트로피 인코더(625)는 일반 제어 데이터, 선택된 예측 정보(예: 인트라 예측 정보 또는 인터 예측 정보), 잔차 정보, 및 기타 적절한 정보를 비트스트림에 포함시키도록 구성된다. 개시된 주제에 따르면, 인터 모드 또는 이중 예측 모드의 병합 서브모드에서 블록을 코딩할 때, 잔차 정보가 없음을 유의한다.

[0069] 도 7은 본 개시의 다른 실시예에 따른 비디오 디코더(710)의 다이어그램을 도시한다. 비디오 디코더(710)는 코딩된 비디오 시퀀스의 일부인 코딩된 화상을 수신하고, 코딩된 화상을 디코딩하여 재구성된 화상을 생성하도록 구성된다. 일 예에서, 비디오 디코더(710)는 도 3의 예에서의 비디오 디코더(310) 대신에 사용된다.

[0070] 도 7 예에서, 비디오 디코더(710)는 도 7에 도시된 바와 같이 함께 결합된, 엔트로피 디코더(771), 인터 디코더(780), 잔차 디코더(773), 재구성 모듈(774) 및 인트라 디코더(772)를 포함한다.

[0071] 엔트로피 디코더(771)는 코딩된 화상을 구성하는 신택스 엘리먼트를 나타내는 특정 심볼을 코딩된 화상으로부터 재구성하도록 구성될 수 있다. 그러한 심볼은 예를 들어, 블록이 코딩되는 모드(예: 인트라 모드, 인터 모드, 이중 예측 모드, 후자의 2개의 병합 서브모드 또는 다른 서브모드), 인트라 디코더(772) 또는 인터 디코더(780)

각각에 의해 예측에 사용되는 특정 샘플 또는 메타데이터를 식별할 수 있는 예측 정보(예를 들어, 인트라 예측 정보 또는 인터 예측 정보), 예를 들어 양자화된 변환 계수의 형태의 잔차 정보 등을 포함할 수 있다. 일 예에서, 예측 모드가 인터 예측 모드 또는 이중 예측 모드일 때, 인터 예측 정보는 인터 디코더(780)에 제공되고; 예측 유형이 인트라 예측 유형일 때우, 인트라 예측 정보는 인트라 디코더(772)에 제공된다. 잔차 정보는 역 양자화될 수 있고 잔차 디코더(773)에 제공된다.

[0072] 인터 디코더(780)는 인터 예측 정보를 수신하고, 인터 예측 정보에 기반하여 인터 예측 결과를 생성하도록 구성된다.

[0073] 인트라 디코더(772)는 인트라 예측 정보를 수신하고, 인트라 예측 정보에 기반하여 예측 결과를 생성하도록 구성된다.

[0074] 잔차 디코더(773)는 역 양자화(inverse quantization)를 수행하여 역 양자화된 변환 계수를 추출하고, 역양자화된 변환 계수(de-quantized transform coefficient)를 처리하여 주파수 도메인에서 공간 도메인으로 잔차를 변환하도록 구성된다. 잔차 디코더(773)는 또한 특정 제어 정보(양자화 파라미터(Quantizer Parameter, QP)를 포함하기 위해)를 요구할 수 있으며, 해당 정보는 엔트로피 디코더(771)에 의해 제공될 수 있다(데이터 경로는 낮은 볼륨 제어 정보일 수 있으므로 도시되지 않음).

[0075] 재구성 모듈(774)은 공간 도메인에서 잔차 디코더(773)에 의해 출력된 잔차와 예측 결과(경우에 따라 인터 또는 인트라 예측 모듈에 의해 출력됨)를 조합하여, 재구성된 화상의 일부일 수 있는 재구성된 블록을 형성하며, 이는 재구성된 비디오의 일부일 수 있다. 디블록킹(deblocking) 작동과 같은 다른 적절한 작동이 시각적 품질을 개선하기 위해 수행될 수 있음에 유의한다.

[0076] 비디오 인코더(303, 503, 603), 및 비디오 디코더(310, 410, 710)는 임의의 적절한 기술을 사용하여 구현될 수 있음에 유의한다. 일 실시예에서, 비디오 인코더(303, 503, 603), 및 비디오 디코더(310, 410, 710)는 하나 이상의 집적 회로를 사용하여 구현될 수 있다. 다른 실시예에서, 비디오 인코더(303, 503, 503), 및 비디오 디코더(310, 410, 710)는 소프트웨어 명령을 실행하는 하나 이상의 프로세서를 사용하여 구현될 수 있다.

[0077] 상이한 화상으로부터의 블록 기반 보상은 모션 보상으로 지칭될 수 있다. 블록 보상은 또한 인트라 화상 블록 보상, 인트라 블록 카피(intra block copy, IBC) 또는 현재 화상 참조(current picture referencing, CPR)로 지칭될 수 있는 동일한 화상 내의 이전에 재구성된 에어리어로부터 수행될 수 있다. 예를 들어, 현재 블록과 참조 블록 사이의 오프셋을 지시하는 변위 벡터(displacement vector)를 블록 벡터라고 한다. 일부 실시예에 따르면, 블록 벡터는 이미 재구성되어 참조용으로 이용 가능한 참조 블록을 가리킨다. 또한, 병렬 처리를 고려하여, 타일/슬라이스 경계 또는 웨이브프론트 사다리꼴 경계(wavefront ladder-shaped boundary)를 넘어선 참조 에어리어도 블록 벡터에 의한 참조에서 제외될 수 있다. 이러한 제약으로 인해, 블록 벡터는 모션 보상에서의 모션 벡터와 상이할 수 있으며, 여기서 모션 벡터는 임의의 값(x 또는 y 방향에서 양수 또는 음수)일 수 있다.

[0078] 블록 벡터의 코딩은 명시적이거나 암시적일 수 있다. 인터 코딩에서 때때로 AMVP(Advanced Motion Vector Prediction) 모드로 지칭되는 명시적 모드에서, 블록 벡터와 그의 예측자 사이의 차이가 시그널링된다. 암시적 모드에서, 블록 벡터는 병합 모드의 모션 벡터와 유사한 방식으로 블록 벡터의 예측자로부터 복구된다. 일부 실시예에서, 블록 벡터의 해상도는 정수 위치(integer position)로 제한된다. 다른 실시예에서, 블록 벡터의 해상도는 분수 위치(fractional position)를 가리키도록 허용될 수 있다.

[0079] 블록 레벨에서 인트라 블록 카피의 사용은 IBC 플래그라고 하는 블록 레벨 플래그를 사용하여 시그널링될 수 있다. 일 실시예에서, IBC 플래그는 현재 블록이 병합 모드에서 코딩되지 않을 때 시그널링된다. IBC 플래그는 또한 현재 디코딩된 화상을 참조 화상으로 처리함으로써 수행되는 참조 인덱스 접근법에 의해 시그널링될 수 있다. HEVC SCC(Screen Content Coding)에서, 이러한 참조 화상이 리스트의 마지막 위치에 놓인다. 이 특별한 참조 화상은 또한 DPB에서 다른 시간적 참조 화상과 함께 관리될 수 있다. IBC는 또한 플립된(flipped) IBC(예: 참조 블록이 현재 블록을 예측하는 데 사용되기 전에 수평 또는 수직으로 플립됨) 또는 라인 기반(line based)(IBC)(예: M×N 코딩 블록 내부의 각 보상 유닛은 M×1 또는 1×N 라인임)과 같은 변형을 포함할 수 있다.

[0080] 도 8은 인트라 화상 블록 보상(예: 인트라 블록 카피 모드)의 실시예를 도시한다. 도 8에서, 현재 화상(800)은 이미 코딩/디코딩된 블록 영역(region) 세트(즉, 회색 사각형) 및 아직 코딩/디코딩되지 않은 블록 영역 세트(즉, 흰색 사각형)를 포함한다. 아직 코딩/디코딩되지 않은 블록 영역 중 하나의 블록(802)은 이전에 코딩/디코딩된 다른 블록(806)을 가리키는 블록 벡터(804)와 연관될 수 있다. 따라서, 블록(806)과 연관된 임의의 모션

정보는 블록(802)의 코딩/디코딩을 위해 사용될 수 있다.

[0081] 일부 실시예에서, CPR 모드의 탐색 범위는 현재 CTU 내에 있도록 제한된다. CPR 모드에 대한 참조 샘플을 저장하기 위한 효과적인 메모리 요건은 샘플의 1CTU 크기이다. 현재 64×64 영역에 재구성된 샘플을 저장하기 위해 기존 참조 샘플 메모리를 고려하면, 3개의 64×64 크기 참조 샘플 메모리가 필요하다. 본 개시의 실시예는 CPR 모드의 유효 탐색 범위를 왼쪽 CTU의 일부로 확장하면서 참조 픽셀을 저장하기 위한 총 메모리 요건은 변경되지 않고 유지된다(1 CTU 크기, 총 4 64×64 참조 샘플 메모리).

[0082] 도 9a에서, CTU(900)의 좌측 상단 영역(upper left region)은 현재 디코딩되고 있는 영역이다. CTU(900)의 좌측 상단 영역이 디코딩될 때, 참조 샘플 메모리의 엔트리[1]는 도 10a에 도시된 바와 같이 이 영역으로부터의 샘플로 덮어 쓰인다(overwritten)(예: 덮어 쓰기된 메모리 위치(들)은 대각선 크로스 해칭(diagonal cross-hatching)을 가짐). 도 9b에서, CTU(900)의 우측 상단 영역은 디코딩되고 있는 다음 현재 영역이다. CTU(900)의 우측 상단 영역이 디코딩될 때, 참조 샘플 메모리의 엔트리[2]는 도 10b에 도시된 바와 같이 이 영역으로부터의 샘플로 덮어 쓰인다. 도 9c에서, CTU(900)의 하부 좌측 영역은 디코딩되는 다음 현재 영역이다. CTU(900)의 좌측 하단 영역이 디코딩될 때, 참조 샘플 메모리의 엔트리[3]는 도 10c에 도시된 바와 같이 이 영역으로부터의 샘플로 덮어 쓰인다. 도 9d에서, CTU(900)의 우측 하단 영역은 디코딩되는 다음 현재 영역이다. CTU(900)의 우측 하단 영역이 디코딩될 때, 참조 샘플 메모리의 엔트리[3]는 도 10d에 도시된 바와 같이 이 영역으로부터의 샘플로 덮어 쓰인다.

[0083] 일부 실시예에서, 비트스트림 적합성 조건(bitstream conformance condition)은 유효한 블록 벡터(1/16-pel 해상도에서 mvL)가 아래에 명시된 조건을 따라야 한다. 일부 실시예에서, 루마 모션 벡터 mvL은 다음의 제약 A1, A2, B1, C1, 및 C2를 따른다.

[0084] 제1 제약(A1)에서, 블록 가용성(block availability)에 대한 유도 프로세스(derivation process)(예: 이웃 블록 가용성 검사 프로세스)가 (xCb, yCb)와 같게 설정된 현재 루마 위치(xCurr, yCurr) 및 이웃 루마 위치(xCb +(mvL[0] >> 4) + cbWidth - 1, yCb +(mvL[1] >> 4) + cbHeight - 1)를 입력으로 사용하여 호출될 때, 출력은 TRUE와 같아야 한다.

[0085] 제2 제약(A2)에서, 블록 가용성에 대한 유도 프로세스(예: 이웃 블록 가용성 검사 프로세스)가 (xCb, yCb)와 같게 설정된 현재 루마 위치(xCurr, yCurr) 및 이웃 루마 위치(xCb +(mvL[0] >> 4) + cbWidth - 1, yCb +(mvL[1] >> 4) + cbHeight - 1)을 입력으로 사용하여 호출될 때, 출력은 TRUE와 같아야 한다.

[0086] 제3 제약 조건(B1)에서, 다음 조건 중 하나 또는 모두가 참(true)이다:

[0087] (i) (mvL[0] >> 4) + cbWidth의 값은 0보다 작거나 같다.

[0088] (ii) (mvL[1] >> 4) + cbHeight의 값은 0보다 작거나 같다.

[0089] 제4 제약 조건(C1)에서, 다음 조건이 참이다:

(i) $(yCb + (mvL[1] >> 4)) >> CtbLog2SizeY = yCb >> CtbLog2SizeY$

(ii) $(yCb + (mvL[1] >> 4) + cbHeight - 1) >> CtbLog2SizeY = yCb >> CtbLog2SizeY$

(iii) $(xCb + (mvL[0] >> 4)) >> CtbLog2SizeY >= (xCb >> CtbLog2SizeY) - 1$

(iv) $(xCb + (mvL[0] >> 4) + cbWidth - 1) >> CtbLog2SizeY <= (xCb >> CtbLog2SizeY)$

[0090]

[0091] 제5 제약 조건(C2)에서, (xCb +(mvL[0] >> 4)) >> CtbLog2SizeY가 (xCb >> CtbLog2SizeY) - 1과 같을 때, 블록 가용성에 대한 유도 프로세스(예: 이웃 블록 가용성 검사 프로세스)는 xCb, yCb)와 같게 설정된 현재 루마 위치(xCurr, yCurr) 및 이웃 루마 위치(((xCb + (mvL[0] >> 4) + CtbSizeY) >> (CtbLog2SizeY - 1)) << (CtbLog2SizeY - 1), ((yCb + (mvL[1] >> 4)) >> (CtbLog2SizeY - 1)) << (CtbLog2SizeY - 1))를 입력으로 사용하여 호출될 때, 출력은 FALSE와 같아야 한다.

[0092] 위의 수식에서, xCb와 yCb는 각각 현재 블록의 x 좌표와 y 좌표이다. 변수 cbHeight 및 cbWidth는 각각 현재 블록의 높이와 너비이다. 변수 CtbLog2sizeY는 log2 도메인의 CTU 크기를 나타낸다. 예를 들어, CtbLog2sizeY =

7은 CTU 크기가 128×128임을 의미한다. 변수 mvL0[0] 및 mvL0[1]은 각각 블록 벡터 mvL0의 x 컴포넌트 및 y 컴포넌트를 나타낸다. 출력이 FALSE이면, 참조 블록에 대한 샘플이 사용 가능한 것으로 결정된다(예: 이웃 블록이 인트라 블록 카피 사용에 이용 가능함). 출력이 TRUE이면, 참조 블록에 대한 샘플은 사용할 수 없는 것으로 결정된다.

[0093] 일부 실시예에 따르면, HMVP(history-based MVP) 방법은 이전에 코딩된 블록의 모션 정보로 정의되는 HMVP 후보를 포함한다. 다수의 HMVP 후보가 있는 표는 인코딩/디코딩 프로세스 동안 유지된다. 신규 슬라이스가 발생할 때 표가 비워진다. 인터 코딩된 비-아핀 블록(non-affine block)이 있을 때마다 연관된 모션 정보가 신규 HMVP 후보로서 표의 마지막 엔트리에 추가된다. HMVP 방법의 코딩 플로(flow)는 도 11a에 도시되어 있다.

[0094] 표 크기 S는 6으로 설정되며, 이는 최대 6개의 HMVP 후보가 표에 추가될 수 있음을 지시한다. 신규 모션 후보를 표에 삽입할 때, 동일한 HMVP가 표에 있는지를 판정하기 위해 중복 검사가 먼저 적용되도록 제한된 FIFO 규칙이 활용된다. 발견되면, 동일한 HMVP가 표에서 제거되고 이후에 모든 HMVP 후보가 앞으로 이동한다(즉, 인덱스가 1 감소). 도 11b는 신규 모션 후보를 HMVP 표에 삽입하는 예를 나타낸다.

[0095] HMVP 후보는 병합 후보 리스트 구성 프로세스에서 사용될 수 있다. 표에서 가장 최근에 나온 여러 HMVP 후보를 순서대로 확인하여 TMVP 후보 다음에 후보 리스트에 삽입한다. 서브 블록 모션 후보(즉, ATMVP)를 제외한 공간적 또는 시간적 병합 후보에 대해 HMVP 후보에 대해 프루닝(pruning)이 적용될 수 있다.

[0096] 일부 실시예에서, 프루닝 작동의 수를 줄이기 위해, 검사할 HMPV 후보의 수(L로 표시됨)는 L = (N <=4) ? M: (8 - N)로 설정되며, 여기서 N은 사용 가능한 비-서브 블록 병합 후보의 수를 지시하고, M은 표에서 사용 가능한 HMVP 후보의 수를 지시한다. 또한, 사용 가능한 병합 후보의 총 수가 시그널링된 최대 허용 병합 후보에서 1을 뺀 값에 도달하면, HMVP 리스트에서 병합 후보 리스트 구성 프로세스가 종료된다. 또한, 조합된 이중 예측 병합 후보 유도를 위한 쌍의 수가 12에서 6으로 감소된다.

[0097] HMVP 후보는 AMVP 후보 리스트 구성 프로세스에서도 사용될 수 있다. 표의 마지막 K개의 HMVP 후보의 모션 벡터는 TMVP 후보 뒤에 삽입된다. AMVP 타깃 참조 화상과 동일한 참조 화상을 가진 HMVP 후보만 AMVP 후보 리스트를 구성하는 데 사용된다. 프루닝은 HMVP 후보에 적용된다. 일부 애플리케이션에서, K는 4로 설정되고 AMVP 리스트 크기는 변경되지 않고 유지된다(즉, 2와 같음).

[0098] 일부 실시예에 따르면, 인트라 블록 카피가 인터 모드와 별개의 모드로 작동할 때, 이전에 코딩된 인트라 블록 카피 블록 벡터를 저장하기 위해 HBVP라고 하는 별도의 히스토리 버퍼가 사용될 수 있다. 인터 예측과 별개의 모드로서, 인트라 블록 카피 모드에 대한 단순화된 블록 벡터 유도 프로세스를 갖는 것이 바람직하다. AMVP 모드에서 IBC BV 예측을 위한 후보 리스트는 IBC 병합 모드에서 사용되는 것(병합 후보 리스트)을 2개의 공간적 후보 + 5개의 HBVP 후보와 공유할 수 있다.

[0099] IBC 모드의 병합 후보 리스트 크기는 MaxNumMergeCand로 할당될 수 있다. MaxNumMergeCand는 일부 예에서 Six_minus_max_num_merge_cand로 지정되는, 인터 모드 병합 후보 리스트 크기 MaxNumMergeCand에 의해 결정될 수 있다. 변수 Six_minus_max_num_merge_cand는 6에서 뺀 슬라이스에서 지원되는, 최대 병합 모션 벡터 예측(motion vector prediction, MVP) 후보 수를 지정할 수 있다.

[0100] 일부 예에서, 최대 병합 MVP 후보 수 MaxNumMergeCand는 다음과 같이 유도될 수 있다.

[0101] 수식(1): MaxNumMergeCand = 6 - six_minus_max_num_merge_cand

[0102] MaxNumMergeCand의 값은 1에서 6까지의 범위일 수 있다. 일부 비디오 코딩 시스템에서, IBC 모드에 대한 병합 리스트 크기는 모든 I/P/B 슬라이스에 대해 병합 모드의 병합 리스트 크기와 별도로 시그널링된다. 이 크기의 범위는 인터 병합 모드와 동일할 수 있다(예: 1에서 6까지 포함). 일부 예에서, 최대 IBC 후보 수 MaxNumIbcMergeCand는 다음과 같이 유도될 수 있다.

[0103] 수식(2): MaxNumIBCMergeCand = 6 - six_minus_max_num_ibc_merge_cand

[0104] 수식(2)에서, 변수 six_minus_max_num_ibc_merge_cand는 6에서 뺀 슬라이스에서 지원되는, 최대 IBC 병합 모션 벡터 예측(motion vector prediction, MVP) 후보 수를 지정한다. MaxNumIBCMergeCand의 값은 1에서 6까지의 범위일 수 있다. 일부 비디오 코딩 시스템에서, IBC 병합 모드에 대한 병합 인덱스 시그널링은 여전히 인터 병합 모드에 사용되는 병합 인덱스 시그널링을 공유할 수도 있다. 이와 관련하여, IBC 병합 모드와 인터 병합 모드는 병합 인덱스에 대해 동일한 신택스 엘리먼트를 공유할 수 있다. 병합 인덱스는 TR(truncated Rice) 코드를 사용하여 이진화되기 때문에 병합 인덱스의 최대 길이는 MaxNumMergeCand - 1이다. 그러나, MaxNumIbcMergeCand가

MaxNumMergeCand와 같지 않을 때, 병합 인덱스 시그널링을 위한 솔루션이 필요하다.

[0105] 본 개시의 실시예는 별도로 사용될 수도 있고, 임의의 순서로 조합되어 사용될 수도 있다. 또한, 본 개시의 실시예에 따른 방법, 인코더 및 디코더 각각은 처리 회로(예: 하나 이상의 프로세서 또는 하나 이상의 집적 회로)에 의해 구현될 수 있다. 일 예에서, 하나 이상의 프로세서는 컴퓨터가 판독가능한 비일시적 매체에 저장된 프로그램을 실행한다. 일부 실시예에 따르면, 블록이라는 용어는 예측 블록, 코딩 블록 또는 코딩 유닛(즉, CU)으로 해석될 수 있다.

[0106] 일부 실시예에 따르면, 병합 인덱스 이진화를 위한 최대 병합 크기 수는 MaxNumMergeCand 수와 MaxNumIbcMergeCand의 수 사이에서 전환 가능하도록 설정된다. 예를 들어, 현재 블록이 IBC 모드로 코딩될 때, 병합 인덱스에 대한 최대 병합 크기는 MaxNumIbcMergeCand이다. 그러나, 현재 블록이 IBC 모드로 코딩되지 않을 때, 병합 인덱스에 대한 최대 병합 크기는 MaxNumMergeCand이다. 표 1은 신택스 엘리먼트 및 관련 이진화의 예를 예시한다.

표 1

[0107]

merge_data( )	regular_merge_flag[ ][ ]	FL	cMax = 1
	mmvd_merge_flag[ ][ ]	FL	cMax = 1
	mmvd_cand_flag[ ][ ]	FL	cMax = 1
	mmvd_distance_idx[ ][ ]	TR	cMax = 7, cRiceParam = 0
	mmvd_direction_idx[ ][ ]	FL	cMax = 3
	ciip_flag[ ][ ]	FL	cMax = 1
	merge_subblock_flag[ ][ ]	FL	cMax = 1
	merge_subblock_idx[ ][ ]	TR	cMax = MaxNumSubblockMergeCand - 1, cRiceParam = 0
	merge_triangle_split_dir[ ][ ]	FL	cMax = 1
	merge_triangle_idx0[ ][ ]	TR	cMax = MaxNumTriangleMergeCand - 1, cRiceParam = 0
	merge_triangle_idx1[ ][ ]	TR	cMax = MaxNumTriangleMergeCand - 2, cRiceParam = 0
	merge_idx[ ][ ]	TR	cMax = (CuPredMode[0][x0][y0] != MODE_IBC ? MaxNumMergeCand : MaxNumIbcMergeCand) - 1, cRiceParam = 0

[0108] 표 1에 예시된 바와 같이, 병합 인덱스(즉, merge_idx[][])의 이진화는 현재 블록이 IBC 모드로 코딩되었는지를 기반으로 한다. 또한, FL은 고정 길이를 나타내며; cMax는 가변 길이 코드의 가능한 최대 값을 나타내고; eRiceParam은 가변 길이 코드의 라이스(rice) 파라미터이다. 라이스 파라미터는 각 입력 값의 이진 코드를 결정하는 데 사용될 수 있다. 잘린 이진 코드(truncated binary code)에서, 라이스 파라미터는 0이다.

[0109] 일부 실시예에 따르면, 병합 인덱스 이진화를 위한 최대 병합 크기 수는 MaxNumMergeCand와 MaxNumIbcMergeCand 사이의 최대 수로 설정된다. I 슬라이스/타일 그룹에서, MaxNumMergeCand의 값은 시그널링되지 않기 때문에, MaxNumMergeCand는 1의 추론된 값(즉, MaxNumIbcMergeCand에 대한 최소 가능한 값)을 가질 수 있다. 따라서, MaxNumMergeCand가 시그널링되지 않을 때, MaxNumMergeCand의 값이 1로 유추되므로 Six_minus_max_num_merge_cand의 값은 5로 유추된다. 따라서, 수식(1)에서, Six_minus_max_num_merge_cand가 5이므로 MaxNumMergeCand는 1과 같다. 수식(2)에서, MaxNumIbcMergeCand의 값은 1에서 6까지의 범위에 있다.

[0110] 표 2는 신택스 및 관련 이진화의 예를 예시한다.

표 2

[0111]

merge_data( )	regular_merge_flag[ ][ ]	FL	cMax = 1
	mmvd_merge_flag[ ][ ]	FL	cMax = 1
	mmvd_cand_flag[ ][ ]	FL	cMax = 1
	mmvd_distance_idx[ ][ ]	TR	cMax = 7, cRiceParam = 0
	mmvd_direction_idx[ ][ ]	FL	cMax = 3
	ciip_flag[ ][ ]	FL	cMax = 1
	merge_subblock_flag[ ][ ]	FL	cMax = 1
	merge_subblock_idx[ ][ ]	TR	cMax = MaxNumSubblockMergeCand - 1, cRiceParam = 0
	merge_triangle_split_dir[ ][ ]	FL	cMax = 1
	merge_triangle_idx0[ ][ ]	TR	cMax = MaxNumTriangleMergeCand - 1, cRiceParam = 0
	merge_triangle_idx1[ ][ ]	TR	cMax = MaxNumTriangleMergeCand - 2, cRiceParam = 0
	merge_idx[ ][ ]	TR	cMax = Max(MaxNumMergeCand, MaxNumIbcMergeCand) - 1, cRiceParam = 0

[0112] 표 2에 예시된 바와 같이, 병합 인덱스(즉, merge_idx[][])의 이진화는 최대 병합 모드 후보 수(즉, MaxNumMergeCand)가 최대 IBC 후보 수(즉, MaxNumIbcMergeCand)보다 큰지 여부에 기반한다.

[0113] 일부 실시예에 따르면, MaxNumIbcMergeCand의 범위는 MaxNumMergeCand보다 작거나 같을 것이다. I 슬라이스/타일 그룹에서, MaxNumMergeCand의 값이 시그널링되지 않으므로 MaxNumMergeCand의 값은 6으로 유추된다. 따라서, 수식(1)에서, MaxNumMergeCand가 6으로 유추되므로 six_minus_max_num_merge_cand의 값은 0으로 유추된다. 일부 실시예에서, 시그널링된 MaxNumIbcMergeCand 값이 MaxNumMergeCand보다 클 때, MaxNumIbcMergeCand는 MaxNumMergeCand로 클리핑된다(clipped). 따라서, 수식(2)에서, 슬라이스 유형이 I이면, MaxNumIbcMergeCand의 값은 1에서 6까지의 범위에 있어야 한다. 그러나, 슬라이스 유형이 P 또는 B이면, MaxNumIbcMergeCand의 값은 1에서 MaxNumMergeCand까지의 범위에 있다. 따라서, 슬라이스 유형이 P 또는 B일 때(즉, MaxNumMergeCand가 시그널링되지 않음), MaxNumIbcMergeCand의 값은 MaxNumIbcMergeCand = min(MaxNumIbcMergeCand, MaxNumMergeCand)와 같이 결정될 수 있다.

[0114] 도 12는 비디오 디코더(710)와 같은 비디오 디코더에 의해 수행되는 비디오 디코딩 프로세스의 실시예를 예시한다. 프로세스는 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계(S1200)에서 시작할 수 있다. 프로세스는 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건이 결정되는 단계(S1202)로 진행한다.

[0115] 프로세스는 미리 결정된 조건에 기반하여, 벡터 예측자 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 최대 병합 모드 후보 수 및 IBC 후보 수 중 하나로 설정하는 단계(S1204)로 진행한다. 예로서, 인덱스는 코딩된 비디오 비트스트림에 포함된 병합 인덱스일 수도 있다. 예를 들어, 미리 결정된 조건은 현재 블록이 IBC 모드로 코딩되는지를 판정하는 것을 포함한다. 현재 블록이 IBC 모드로 인코딩되면, 인덱스의 크기는 MaxNumIbcMergeCand로 설정된다. 그러나 현재 블록이 IBC 모드로 인코딩되지 않으면, 인덱스의 크기는 MaxNumMergeCand로 설정된다.

[0116] 다른 예로, 미리 결정된 조건은 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 큰지를 판정하는 것을 포함한다. 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 크면, 인덱스의 크기가 MaxNumMergeCand로 설정된다. 그러나, 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 적으면, 인덱스의 크기는 MaxNumIbcMergeCand로 설정된다.

[0117] 프로세스는 단계(S1204)에서 후보 리스트가 벡터 예측자로 구성되는 단계(S1206)로 진행된다. 예를 들어, 현재 블록이 병합 모드로 코딩되면, 후보 리스트는 병합 리스트이고 벡터 예측자는 모션 벡터이다. 다른 예에서, 현재 블록이 IBC 모드로 코딩되면, 후보 리스트는 블록 벡터 예측자의 리스트이다. 프로세스는 결정된 인덱스 크기를 초과하지 않는 값을 갖는 인덱스에 따라 후보 리스트로부터 벡터 예측자를 검색하는 단계(S1208)부터 진행한다. 예를 들어, 후보 리스트에서 벡터 예측자를 검색하기 위해 사용되는 인덱스의 값은 단계(S1204)에서 결정된 인덱스의 크기를 초과할 수 없다. 프로세스는 검색된 벡터 예측자에 따라 현재 블록을 디코딩하는 단계(S1210)로 진행한다.

[0118] 도 13은 비디오 디코더(710)와 같은 비디오 디코더에 의해 수행되는 비디오 디코딩 프로세스의 실시예를 도시한다. 프로세스는 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계(S1300)에서 시작할 수 있다. 프로세스는 코딩된 비디오 비트스트림으로부터의 시그널링 데이터가 현재 블록에 대해 검색되는 단계(S1302)로 진행한다. 프로세스는 최대 병합 후보 수가 현재 블록에 대해 검색된 시그널링 데이터에 포함되는지를 판정하기 위해 단계(S1304)로 진행한다. 예를 들어, MaxNumMergeCand가 시그널링되는지를 판정한다. 위에서 논의된 바와 같이, 일부 예에서, MaxNumMergeCand는 I 슬라이스/타일 그룹 유형에 대해 시그널링되지 않고, P 또는 B 슬라이스/타일 그룹 유형에 대해 시그널링된다.

[0119] 최대 병합 후보 수가 현재 블록에 대한 시그널링 데이터에 포함되는지의 판정에 기반하여 최대 IBC(Intra Block Copy) 후보 수를 설정하는 단계(S1306)로 진행한다. 예를 들어, MaxNumMergeCand가 시그널링되지 않으면, MaxNumIbcMergeCand의 값은 1에서 6까지(포함)이다. 그러나 MaxNumMergeCand가 시그널링되면, MaxNumIbcMergeCand의 값의 범위는 1에서 MaxNumMergeCand까지(포함)이다.

[0120] 일부 예에서, 코딩 유닛은 루마 컴포넌트 및 크로마 컴포넌트 모두의 샘플들을 포함한다. 이러한 크로마 컴포넌트의 샘플은 루마 컴포넌트의 샘플과 비교하여 독립적이거나 별도의 분할 트리 구조(split tree structure)를 가질 수 있다. 일부 예에서, 별도의 코딩 트리 구조는 CTU 레벨부터 시작한다. 따라서, 크로마 CU(예: 2개의 크로마 컴포넌트만 포함하는 CU)가 대응하는 샘플 위치에서 크로마 CU의 루마 대응물(counterpart)보다 클 수 있다.

[0121] 일부 실시예에 따르면, 제1 방법에서, 듀얼 트리 구조(dual-tree structure)가 사용될 때, 크로마 블록은 적어도 다음 조건이 충족될 때 IBC 모드로 코딩될 수 있다:

[0122] 1. 크로마 CU의 각 서브블록 영역이 경우, 나란히 배치된(collated) 루마 에어리어는 IBC 모드로 코딩된다.

[0123] 2. 크로마 CU에 대해 나란히 배치된 모든 루마 에어리어 샘플은 동일한 블록 벡터를 갖는다.

[0124] 3. 전체 크로마 CU에 대해 유도된 공통 블록 벡터는 유효한(valid) BV이다. 이는 BV가 현재 화상에서 그리고 크로마 컴포넌트에 대해 주어진 제약 에어리어 내에서 이미 재구성된 에어리어를 가리킴을 의미한다.

[0125] 제1 방법에 기반하여, 디코더 측에서는 크로마 CU를 서브블록 기반 CU가 아닌 전체 CU로 취급할 수 있다. 따라서 CU를 디코딩하기 위해 나란히 배치된 루마 에어리어(예: 일반적으로 CU의 왼쪽 상단 코너(corner))에서 유도된 단일 BV를 사용하는 것으로 충분하다.

[0126] 일부 실시예에 따르면, 제2 방법에서, 듀얼 트리 구조가 사용될 때, 듀얼 트리 구조와 함께 크로마 IBC 모드의 사용을 가능하게 하기 위해 상이한 조건이 사용될 수 있다. 일 실시예에서, 크로마 블록은 (i) 모든 크로마 샘플의 대응하는 루마 샘플이 동일한 루마 코딩 블록에 속하고; (ii) 동일한 루마 코딩 블록이 IBC 모드로 코딩될 때, IBC 모드로 코딩될 수 있다. 예를 들어, 이 조건은 크로마 CU의 두 코너를 평가하는 것에 의해 검사된다. 왼쪽 상단 크로마 샘플의 루마 대응(luma correspondence)과 오른쪽 하단 크로마 샘플의 루마 대응이 동일한 루마 코딩 블록에 속한다면, 전체 크로마 CU의 대응하는 루마 에어리어는 동일한 루마 코딩 블록에 속한다. 다른 실시예에서, 크로마 블록은 대응하는 루마 코딩 시 IBC 모드에서 코딩될 수 있다.

[0127] 제2 방법에 기반하여, 디코더 측에서는 크로마 CU를 서브블록 기반 CU 대신 전체 CU로 취급할 수 있다. 따라서, CU 디코딩을 위해 나란히 배치된 루마 에어리어(일반적으로 CU의 왼쪽 상단 코너)로부터 유도된 단일 BV를 사용하는 것으로 충분하다.

[0128] 일부 실시예에 따르면, 듀얼 트리 구조에 관한 제1 방법 또는 제2 방법에 대해, 다음의 개시된 비제한적인 실시예는 제1 방법 또는 제2 방법에서 위의 조건이 충족될 때, 듀얼 트리와 함께 크로마 IBC 모드의 사용을 시그널링하는 방법을 보여준다.

[0129] 일 실시예에서, 듀얼 트리 구조를 갖는 크로마 CU에 대한 IBC 모드를 사용하기 위한 위의 제약은 적용 가능한 경우 각각의 크로마 CU에 대해 사용 플래그(usage flag)(예: ibc_flag)가 시그널링되도록 구현된다. 그러나 본 실시예에서, 제1 방법에 대한 모든 조건 또는 제2 방법에 대한 모든 조건이 충족되는 경우에만 ibc_flag가 참으로 시그널링된다. 그렇지 않으면, ibc_flag가 거짓(false)으로 시그널링된다. 일부 예에서, 제1 방법 또는 제2 방법에 대해 모든 조건이 충족될 때, 인코더가 구현되는 방식에 따라 ibc_flag가 또한 거짓으로 시그널링될 수 있다.

[0130] 다른 실시예에서, 듀얼 트리 구조를 갖는 크로마 CU에 대한 IBC 모드를 사용하기 위한 위의 제약은 사용 플래그(예: ibc_flag)가 전혀 시그널링되지 않도록 구현된다. 예를 들어, 듀얼 트리 구조의 크로마 CU의 경우, 제1 방

법 또는 제2 방법에 대한 모든 조건이 충족될 때, CU는 IBC 모드로 인코딩되고 ibc_flag는 참으로 유추된다. 그렇지 않으면, ibc_flag가 시그널링되지 않고 거짓으로 유추된다.

[0131] 위에서 설명된 기술은 컴퓨터가 판독가능한 명령을 사용하여 컴퓨터 소프트웨어로서 구현될 수 있고, 하나 이상의 컴퓨터가 판독가능한 매체에 물리적으로 저장될 수 있다. 예를 들어, 도 14는 개시된 주제의 특정 실시예를 구현하기에 적합한 컴퓨터 시스템(1400)을 도시한다.

[0132] 컴퓨터 소프트웨어는 어셈블리, 컴필레이션(compilation), 링크 또는 유사 메커니즘이 적용될 임의의 적절한 머신 코드 또는 컴퓨터 언어를 사용하여 코딩되어, 하나 이상의 컴퓨터 중앙 처리 유닛(central processing unit, CPU), 그래픽 처리 유닛(Graphics Processing Unit, GPU) 등에 의해 직접 실행되거나 해석, 마이크로 코드 실행 등을 통해 실행될 수 있는 명령을 포함하는 코드를 생성할 수 있다.

[0133] 명령은 예를 들어 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트폰, 게임 디바이스, 사물 인터넷 디바이스 등을 포함하는 다양한 유형의 컴퓨터 또는 그의 컴포넌트 상에서 실행될 수 있다.

[0134] 컴퓨터 시스템(1400)에 대한 도 14에 도시된 컴포넌트는 본질적으로 예시적이며, 본 개시의 실시예를 구현하는 컴퓨터 소프트웨어의 사용 또는 기능의 범위에 대한 임의의 제한을 제안하도록 의도되지 않는다. 또한, 컴포넌트의 구성은 컴퓨터 시스템(1400)의 예시적인 실시예에 예시된 컴포넌트 중 임의의 하나 또는 조합과 관련된 임의의 종속성 또는 요건을 갖는 것으로 해석되어서는 안 된다.

[0135] 컴퓨터 시스템(1400)은 특정 휴먼 인터페이스 입력 디바이스를 포함할 수 있다. 그러한 휴먼 인터페이스 입력 디바이스는 예를 들어 촉각 입력(예: 키스트로크(keystroke), 스와이프(swipe), 데이터 장갑 움직임(data glove movement)), 오디오 입력(예: 음성, 박수), 시각적 입력(예: 제스처), 후각 입력(도시되지 않음)을 통해 하나 이상의 인간 사용자에 의한 입력에 응답할 수 있다. 휴먼 인터페이스 디바이스는 오디오(예: 음성, 음악, 주변 사운드), 이미지(예: 스캔 이미지, 스틸 이미지 카메라로부터 획득된 사진 이미지), 비디오(예: 2차원 비디오, 입체 비디오를 포함하는 3차원 비디오)와 같은, 인간의 의식적 입력과 반드시 직접적으로 관련이 없는 특정 미디어를 캡처하는 데에도 사용할 수 있다.

[0136] 입력 휴먼 인터페이스 디바이스는 (각각 묘사된 것 중 하나만) 키보드(1401), 마우스(1402), 트랙패드(1403), 터치 스크린(1410), 데이터 글러브(도시되지 않음), 조이스틱(1405), 마이크(1406), 스캐너(1407), 카메라(1408) 중 하나 이상을 포함할 수 있다.

[0137] 컴퓨터 시스템(1400)은 또한 특정 휴먼 인터페이스 출력 디바이스를 포함할 수 있다. 이러한 휴먼 인터페이스 출력 디바이스는 예를 들어 촉각 출력, 사운드, 빛 및 냄새/미각을 통해 한 명 이상의 인간 사용자의 감각을 자극할 수 있다. 이러한 휴먼 인터페이스 출력 디바이스는 촉각 출력 디바이스(예를 들어, 터치 스크린(1410), 데이터 글러브(도시되지 않음) 또는 조이스틱(1405)에 의한 촉각 피드백, 그러나 입력 디바이스로서 기능하지 않는 촉각 피드백 디바이스가 있을 수 있음), 오디오 출력 디바이스(예: 스피커(1409), 헤드폰(도시되지 않음)), 시각 출력 디바이스(예: CRT 스크린, LCD 스크린, 플라즈마 스크린, OLED 스크린을 포함하는 스크린(1410) - 각각 터치 스크린 입력 기능을 갖거나 갖지 않으며, 각각 촉각 피드백 기능을 갖거나 갖지 않으며 - 일부는 입체 출력과 같은 수단을 통해 2차원 시각적 출력 또는 3차원 이상의 출력을 출력할 수 있음 -; 가상 현실 안경(도시되지 않음), 홀로그래픽 디스플레이 및 연기 탱크(smoke tank)(도시되지 않음) 및 프린터(도시되지 않음)를 포함할 수 있다.

[0138] 컴퓨터 시스템(1400)은 또한 CD/DVD 또는 이와 유사한 매체(1421)가 있는 CD/DVD ROM/RW(1420), 섬 드라이브(thumb-drive)(1422), 이동식 하드 드라이브 또는 솔리드 스테이트 드라이브(1423), 테이프 및 플로피 디스크(도시되지 않음)와 같은 레거시 자기 매체(legacy magnetic media), 보안 동글(dongle)(도시되지 않음)과 같은 특수 ROM/ASIC/PLD 기반 디바이스 등을 포함할 수 있다.

[0139] 당업자는 또한 현재 개시된 주제와 관련하여 사용되는 용어 "컴퓨터가 판독가능한 매체"가 전송 매체, 반송파, 또는 다른 일시적 신호를 포함하지 않는다는 것을 이해해야 한다.

[0140] 컴퓨터 시스템(1400)은 또한 하나 이상의 통신 네트워크에 대한 인터페이스를 포함할 수 있다. 네트워크는 예를 들어 무선, 유선, 광이 될 수 있다. 네트워크는 또한 로컬, 광역, 대도시, 차량 및 산업, 실시간, 지연 허용(delay-tolerant) 등일 수 있다. 네트워크의 예로는 이더넷, 무선 LAN과 같은 근거리 네트워크, GSM, 3G, 4G, 5G, LTE 등을 포함하는 셀룰러 네트워크, 케이블 TV, 위성 TV 및 지상파 방송을 포함하는 TV 유선 또는 무선 광역 디지털 네트워크, CANBus 등을 포함하는 차량 및 산업용을 포함한다. 특정 네트워크는 일반적으로 특정 범용 데이터 포트 또는 주변 버스(1449)(예: 컴퓨터 시스템(1400)의 USB 포트)에 부착된 외부 네트워크 인터페이스

어댑터를 필요하며; 다른 것들은 일반적으로 아래에 설명된 바와 같이 시스템 버스(예를 들어, PC 컴퓨터 시스템에 대한 이더넷 인터페이스 또는 스마트폰 컴퓨터 시스템에 대한 셀룰러 네트워크 인터페이스)에 부착함으로써 컴퓨터 시스템(1400)의 코어에 통합된다. 이러한 네트워크 중 하나를 사용하여 컴퓨터 시스템(1400)은 다른 엔티티와 통신할 수 있다. 이러한 통신은 예를 들어, 로컬 또는 광역 디지털 네트워크를 사용하는 다른 컴퓨터 시스템에 대해, 단방향, 수신 전용(예를 들어, 방송 TV), 단방향 송신 전용(예를 들어, CANbus에서 특정 CANbus 디바이스로) 또는 양방향일 수 있다. 특정 프로토콜 및 프로토콜 스택은 위에서 설명한 대로 각 네트워크 및 네트워크 인터페이스에서 사용할 수 있다.

[0141] 전술한 휴먼 인터페이스 디바이스, 사람이 접근 가능한 저장 디바이스, 네트워크 인터페이스가 컴퓨터 시스템(1400)의 코어(1440)에 부착될 수 있다.

[0142] 코어(1440)는 하나 이상의 중앙 처리 유닛(Central Processing Unit, CPU)(1441), 그래픽 처리 유닛(Graphics Processing Unit, GPU)(1442), FPGA(Field Programmable Gate Areas)(1443) 형태의 전문 프로그래머블 처리 유닛, 특정 태스크를 위한 하드웨어 가속기(accelerator)(1444) 등 중 하나 이상을 포함할 수 있다. 이러한 디바이스는 읽기 전용 메모리(Read-only memory, ROM)(1445), 랜덤 액세스 메모리(1446), 사용자가 액세스할 수 없는 내부 하드 드라이브, SSD 등과 같은 내부 대용량 저장소(mass storage)(1447)와 함께, 시스템 버스(1448)를 통해 연결될 수 있다. 일부 컴퓨터 시스템에서, 시스템 버스(1448)는 추가 CPU, GPU 등에 의한 확장을 가능하게 하는 하나 이상의 물리적 플러그의 형태로 액세스할 수 있다. 주변 디바이스는 코어의 시스템 버스(1448)에 직접 부착되거나 주변 디바이스 버스(1449)를 통해 부착될 수 있다. 주변기기 버스를 위한 아키텍처에는 PCI, USB 등이 포함된다.

[0143] CPU(1441), GPU(1442), FPGA(1443) 및 가속기(1444)는 조합하여 앞서 언급한 컴퓨터 코드를 구성할 수 있는 특정 명령을 실행할 수 있다. 해당 컴퓨터 코드는 ROM(1445) 또는 RAM(1446)에 저장될 수 있다. 과도기의 데이터(Transitional data)는 RAM(1446)에도 저장될 수 있는 반면, 영구 데이터(permanent data)는 예를 들어 내부 대용량 저장소(1447)에 저장될 수 있다. 임의의 메모리 디바이스에 대한 빠른 저장 및 검색은 하나 이상의 CPU(1441), GPU(1442), 대용량 저장소(1447), ROM(1445), RAM(1446) 등과 밀접하게 연관될 수 있는, 캐시 메모리의 사용을 통해 활성화될 수 있다.

[0144] 컴퓨터가 판독가능한 매체는 다양한 컴퓨터 구현 작동을 수행하기 위한 컴퓨터 코드를 가질 수 있다. 매체 및 컴퓨터 코드는 본 개시의 목적을 위해 특별히 설계되고 구성된 것일 수 있거나, 컴퓨터 소프트웨어 기술 분야의 숙련자에게 잘 알려져 있고 이용 가능한 종류일 수 있다.

[0145] 제한이 아닌 예로서, 아키텍처(1400), 특히 코어(1440)를 갖는 컴퓨터 시스템은, 하나 이상의 유형의 컴퓨터가 판독가능한 매체에 구현된 소프트웨어를 실행하는 프로세서(들)(CPU, GPU, FPGA, 가속기 등을 포함함)의 결과로서의 기능성(functionality)을 제공할 수 있다. 이러한 컴퓨터가 판독가능한 매체는 위에서 소개한 사용자 액세스 가능한 대용량 저장소와 관련된 매체일 수 있을 뿐만 아니라 코어 내부 대용량 저장소(1447) 또는 ROM(1445)과 같은 비일시적 특성을 갖는 코어(1440)의 특정 저장소일 수 있다. 본 개시의 다양한 실시예를 구현하는 소프트웨어는 이러한 디바이스에 저장되어 코어(1440)에 의해 실행될 수 있다. 컴퓨터가 판독가능한 매체는 특정 요구에 따라 하나 이상의 메모리 디바이스 또는 칩을 포함할 수 있다. 소프트웨어는 코어(1440) 및 특히 그 안의 프로세서(CPU, GPU, FPGA 등을 포함함)가 RAM(1446)에 저장된 데이터 구조를 정의하고, 소프트웨어에서 정의한 프로세스에 따라 이러한 데이터 구조를 수정하는 것을 포함하여, 여기에 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하게 할 수 있다. 추가로 또는 대안으로서, 컴퓨터 시스템은 하드웨어에 내장된 또는 그렇지 않으면 회로(예: 가속기(1444))에 내장된 로직의 결과로서의 기능성을 제공할 수 있으며, 이는 소프트웨어를 대신하거나 소프트웨어와 함께 작동하여, 여기에 기술된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행할 수 있다. 소프트웨어에 대한 참조는 로직을 포함할 수 있으며, 그 반대의 경우도 적절하다. 컴퓨터가 판독가능한 매체에 대한 참조는 적절한 경우 실행을 위한 소프트웨어를 저장하는 회로(예: 집적 회로(integrated circuit, IC)), 실행을 위한 로직을 구현하는 회로, 또는 둘 다를 포함할 수 있다. 본 개시는 하드웨어 및 소프트웨어의 임의의 적절한 조합을 포함한다.

[0146] 부록 A: 약어(Acronyms)

[0147] JEM: joint exploration model

[0148] VVC: versatile video coding

[0149] BMS: benchmark set

[0150] MV: Motion Vector

[0151] HEVC: High Efficiency Video Coding

[0152] SEI: Supplementary Enhancement Information

[0153] VUI: Video Usability Information

[0154] GOPs: Groups of Pictures

[0155] TUs: Transform Units,

[0156] PUs: Prediction Units

[0157] CTUs: Coding Tree Units

[0158] CTBs: Coding Tree Blocks

[0159] PBs: Prediction Blocks

[0160] HRD: Hypothetical Reference Decoder

[0161] SNR: Signal Noise Ratio

[0162] CPUs: Central Processing Units

[0163] GPUs: Graphics Processing Units

[0164] CRT: Cathode Ray Tube

[0165] LCD: Liquid-Crystal Display

[0166] OLED: Organic Light-Emitting Diode

[0167] CD: Compact Disc

[0168] DVD: Digital Video Disc

[0169] ROM: Read-Only Memory

[0170] RAM: Random Access Memory

[0171] ASIC: Application-Specific Integrated Circuit

[0172] PLD: Programmable Logic Device

[0173] LAN: Local Area Network

[0174] GSM: Global System for Mobile communications

[0175] LTE: Long-Term Evolution

[0176] CANBus: Controller Area Network Bus

[0177] USB: Universal Serial Bus

[0178] PCI: Peripheral Component Interconnect

[0179] FPGA: Field Programmable Gate Areas

[0180] SSD: solid-state drive

[0181] IC: Integrated Circuit

[0182] CU: Coding Unit

[0183] 본 개시는 몇몇 예시적인 실시예를 설명하였지만, 개시의 범위 내에 속하는 변경, 순열 및 다양한 대체 등가물이 있다. 따라서, 당업자는 본 명세서에 명시적으로 도시되거나 설명되지는 않았지만 본 개시의 원리를 구현하고 따라서 본 개시의 사상 및 범위 내에 있는 수많은 시스템 및 방법을 고안할 수 있음을 이해할 것이다.

[0184] (1) 비디오 디코딩 방법은, 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계; 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계; 미리 결정된 조건에 기반하여, 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보의 수에 기반한 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계; 벡터 예측자로 후보 리스트를 구성하는 단계; 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 후보 리스트에서 벡터 예측자를 검색하는 단계; 및 검색된 벡터 예측자에 따라 현재 블록을 디코딩하는 단계를 포함한다.

[0185] (2) 특징(1)에 따른 비디오 디코딩 방법에서, 미리 결정된 조건은 현재 블록이 IBC 모드로 인코딩되는지를 판정하는 것을 포함한다.

[0186] (3) 특징(2)에 따른 비디오 디코딩 방법에서, 현재 블록이 IBC 모드로 인코딩된다는 결정에 응답하여, 인덱스의 크기를 최대 IBC 후보 수(maximum number of IBC candidates)로 설정하고, 현재 블록이 IBC 모드로 인코딩되지 않는다는 결정에 응답하여, 인덱스의 크기를 최대 병합 모드 후보 수(maximum number of merge mode candidates)로 설정한다.

[0187] (4) 특징(1) 내지 특징(3) 중 어느 하나에 따른 비디오 디코딩 방법에서, 미리 결정된 조건은 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 큰지를 판정하는 것을 포함한다.

[0188] (5) 특징(4)에 따른 비디오 디코딩 방법에서, 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 크다는 결정에 응답하여 인덱스의 크기를 최대 병합 모드 후보 수로 설정하고, 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 작다는 결정에 응답하여, 인덱스의 크기를 최대 IBC 후보 수로 설정한다.

[0189] (6) 비디오 디코딩 방법은, 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계; 현재 블록에 대해 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색하는 단계; 현재 블록에 대해 검색된 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지 판정하는 단계; 및 현재 블록에 대한 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지의 판정에 기반하여, 최대 IBC(Intra Block Copy) 후보 수를 설정하는 단계를 포함한다.

[0190] (7) 특징(6)에 따른 비디오 디코딩 방법에서, 시그널링 데이터에 최대 병합 후보 수가 포함되어 있지 않다는 결정에 응답하여, 최대 IBC 후보 수는 1 내지 6 사이의 값으로 설정된다.

[0191] (8) 특징(7)에 따른 비디오 디코딩 방법에서, 시그널링 데이터에 최대 병합 후보 수가 포함되어 있다는 결정에 응답하여, 최대 IBC 후보 수는 1 내지 최대 병합 후보 수 사이의 값으로 설정된다.

[0192] (9) 비디오 디코딩을 위한 비디오 디코딩 장치는, 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하고, 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하며, 미리 결정된 조건에 기반하여, 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보의 수에 기반한 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하고, 벡터 예측자로 후보 리스트를 구성하며, 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 후보 리스트에서 벡터 예측자를 검색하고, 그리고 검색된 벡터 예측자에 따라 현재 블록을 디코딩하도록 구성된 처리 회로를 포함한다.

[0193] (10) 특징(9)에 따른 비디오 디코딩 장치에서, 미리 결정된 조건은 현재 블록이 IBC 모드로 인코딩되는지를 판정하는 것을 포함한다.

[0194] (11) 특징(10)에 따른 비디오 디코딩 장치에서, 현재 블록이 IBC 모드로 인코딩된다는 결정에 응답하여, 처리 회로는 인덱스의 크기를 최대 IBC 후보 수로 설정하도록 구성되고, 현재 블록이 IBC 모드로 인코딩되지 않는다는 결정에 응답하여, 처리 회로는 인덱스의 크기를 최대 병합 모드 후보 수로 설정하도록 구성된다.

[0195] (12) 특징(9) 내지 특징(12) 중 어느 하나에 따른 비디오 디코딩 장치에서, 미리 결정된 조건은 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 큰지를 판정하는 것을 포함한다.

[0196] (13) 특징(12)의 비디오 디코딩 장치에서, 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 크다는 결정에 응답하여, 처리 회로는 추가로, 인덱스의 크기를 최대 병합 모드 후보 수로 설정하도록 구성되고, 최대 병합 모드 후보 수가 최대 IBC 후보 수보다 작다는 결정에 응답하여, 처리 회로는 추가로, 인덱스의 크기를 최대 IBC 후보 수로 설정하도록 구성된다.

[0197] (14) 비디오 디코딩을 위한 비디오 디코더 장치는, 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하고, 현재 블록에 대해 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색하며, 현재 블록에 대해

검색된 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지 판정하고, 그리고 현재 블록에 대한 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지의 판정에 기반하여, 최대 IBC(Intra Block Copy) 후보 수를 설정하도록 구성된 처리 회로를 포함한다.

[0198] (15) 특징(14)의 비디오 디코더에서, 시그널링 데이터에 최대 병합 후보 수가 포함되어 있지 않다는 결정에 응답하여, 최대 IBC 후보 수는 1 내지 6 사이의 값으로 설정된다.

[0199] (16) 특징(15)에 따른 비디오 디코더에서, 시그널링 데이터에 최대 병합 후보 수가 포함되어 있다는 결정에 응답하여, 최대 IBC 후보 수는 1 내지 최대 병합 후보 수 사이의 값으로 설정된다.

[0200] (17) 컴퓨터가 판독가능한 비일시적 매체는, 명령이 저장되어 있으며, 명령은 비디오 디코더의 프로세서에 의해 실행될 때, 프로세서가, 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계; 코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정하는 단계; 미리 결정된 조건에 기반하여, 병합 모드 후보의 수 및 IBC(Intra Block Copy) 후보의 수에 기반한 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정하는 단계; 벡터 예측자로 후보 리스트를 구성하는 단계; 인덱스의 결정된 크기를 초과하지 않는 값을 갖는 인덱스에 따라 후보 리스트에서 벡터 예측자를 검색하는 단계; 및 검색된 벡터 예측자에 따라 현재 블록을 디코딩하는 단계를 포함하는 방법을 수행하게 한다.

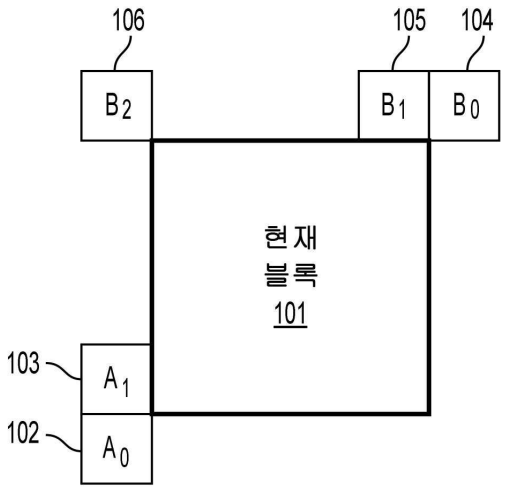
[0201] (18) 특징(17)에 따른 컴퓨터가 판독가능한 비일시적 매체에서, 미리 결정된 조건은 현재 블록이 IBC 모드로 인코딩되는지를 판정하는 것을 포함한다.

[0202] (19) 컴퓨터가 판독가능한 비일시적 매체는, 명령이 저장되어 있으며, 명령은 비디오 디코더의 프로세서에 의해 실행될 때, 프로세서가, 현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신하는 단계; 현재 블록에 대해 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색하는 단계; 현재 블록에 대해 검색된 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지 판정하는 단계; 및 현재 블록에 대한 시그널링 데이터에 최대 병합 후보 수가 포함되어 있는지의 판정에 기반하여, 최대 IBC(Intra Block Copy) 후보 수를 설정하는 단계를 포함하는 방법을 수행하게 한다.

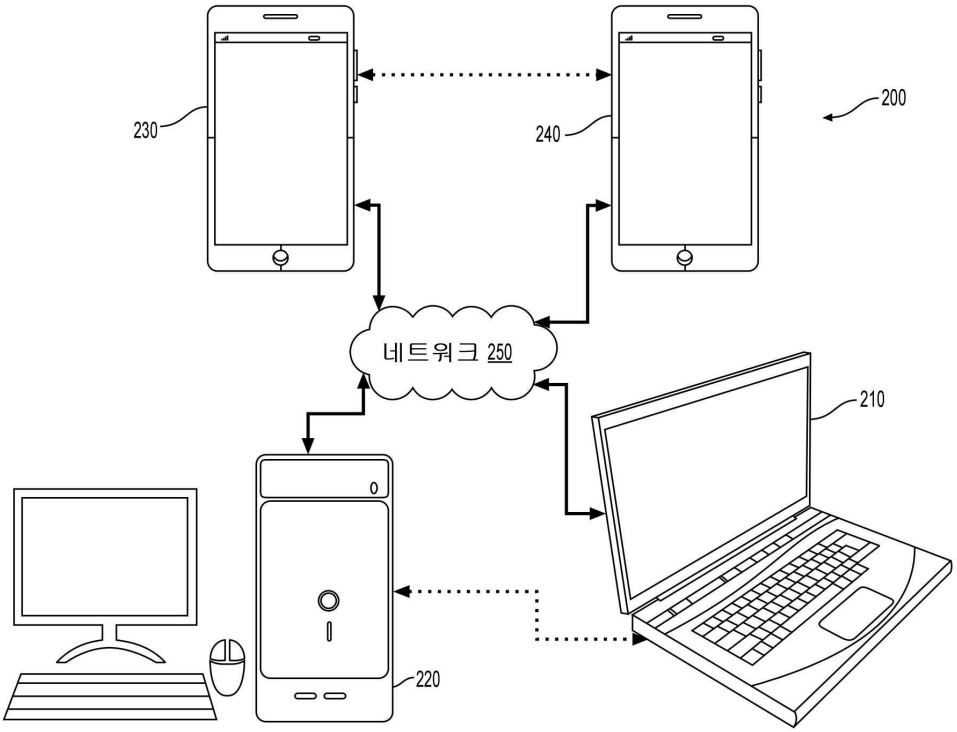
[0203] (20) 특징(19)에 따른 컴퓨터가 판독가능한 비일시적 매체에서, 시그널링 데이터에 최대 병합 후보 수가 포함되어 있지 않다는 결정에 응답하여, 최대 IBC 후보 수는 1 내지 6 사이의 값으로 설정된다.

도면

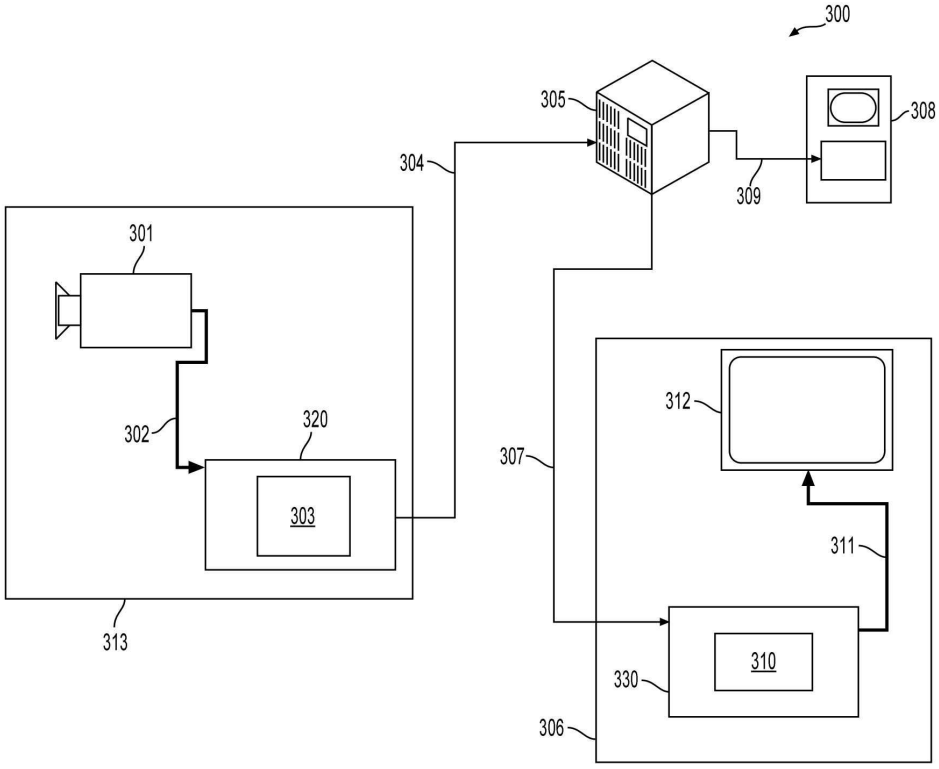
도면1



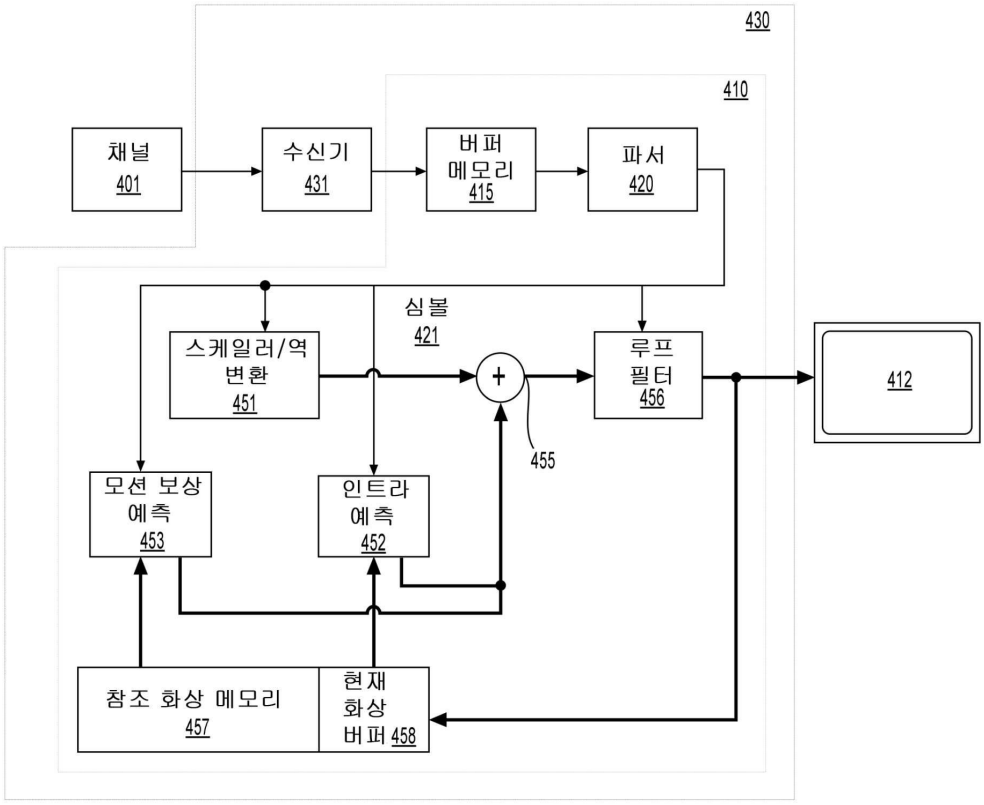
도면2


230
240
200
네트워크 250
210
220

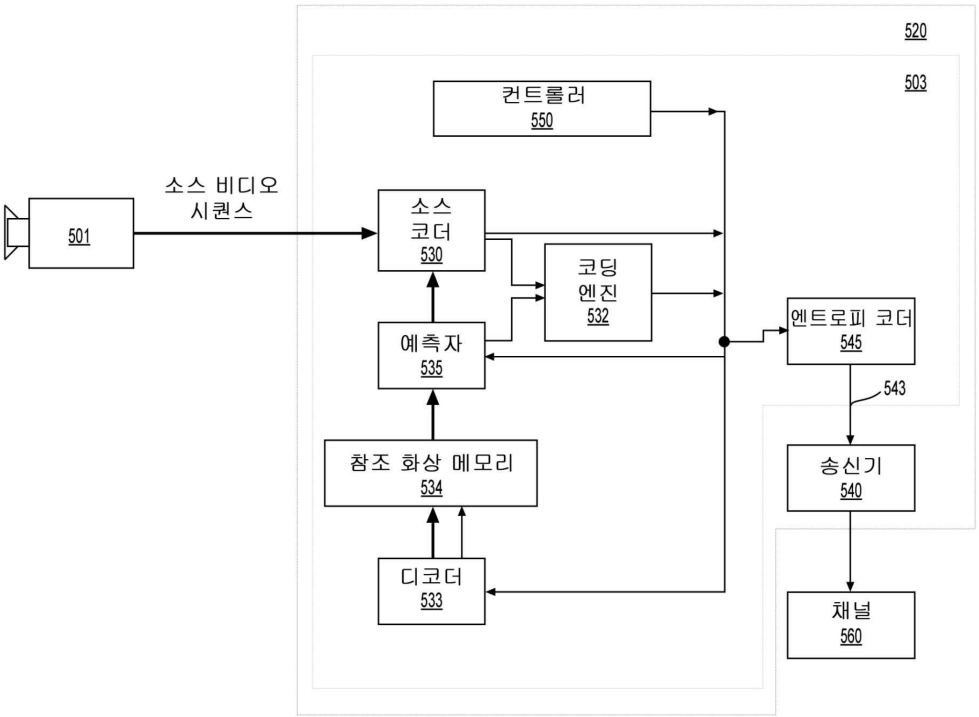
도면3


300
305
308
304
309
301
302
320
303
313
312
307
311
310
330
306

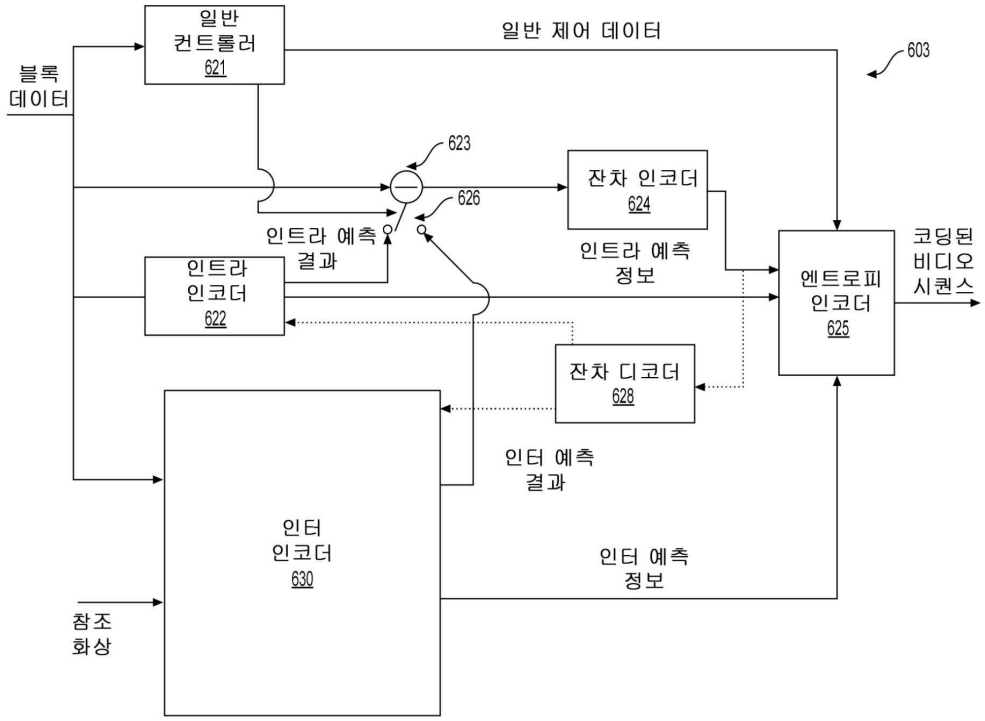
도면4


430
410
채널
401
수신기
431
버퍼
메모리
415
파서
420
심볼
421
스케일러/역
변환
451
+
455
루프
필터
456
412
모션 보상
예측
453
인트라
예측
452
참조 화상 메모리
457
현재
화상
버퍼 458

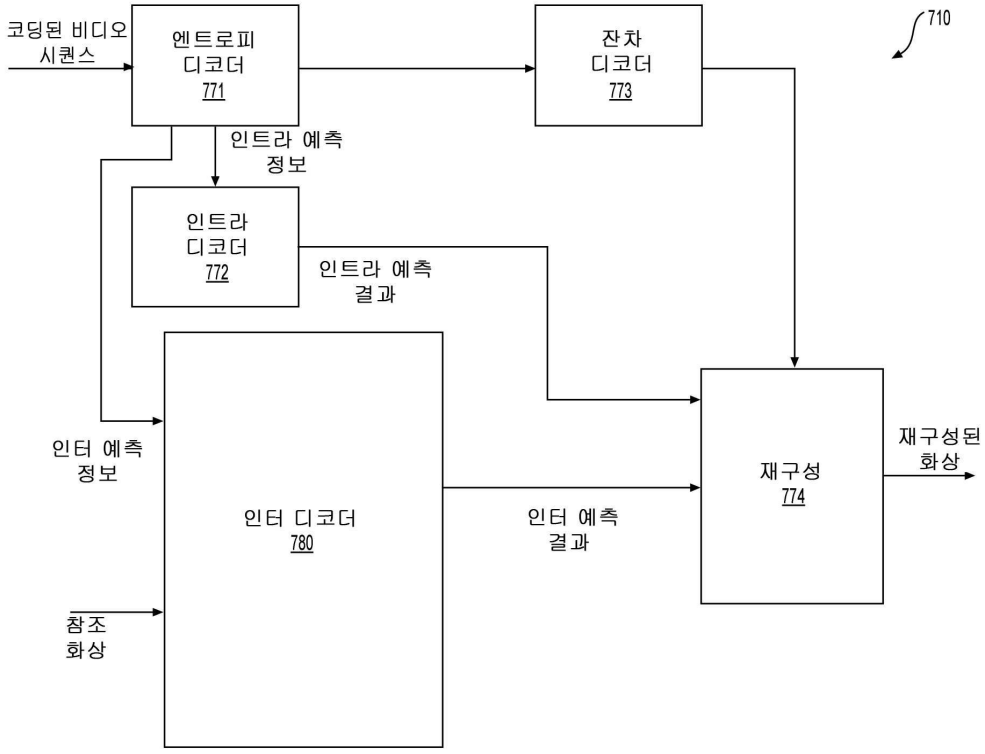
도면5


520
503
컨트롤러
550
소스 비디오
시퀀스
501
소스
코더
530
코딩
엔진
532
엔트로피 코더
545
예측자
535
543
참조 화상 메모리
534
송신기
540
디코더
533
채널
560

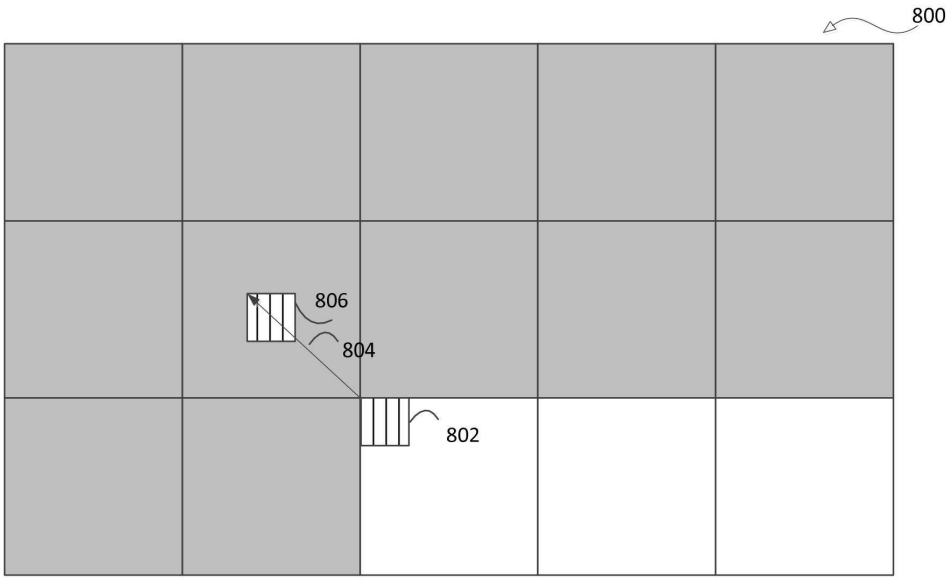
도면6


일반
컨트롤러
621
일반 제어 데이터
603
블록
데이터
623
626
잔차 인코더
624
인트라 예측
결과
인트라 예측
정보
인트라
인코더
622
엔트로피
인코더
625
코딩된
비디오
시퀀스
잔차 디코더
628
인터 예측
결과
인터
인코더
630
인터 예측
정보
참조
화상

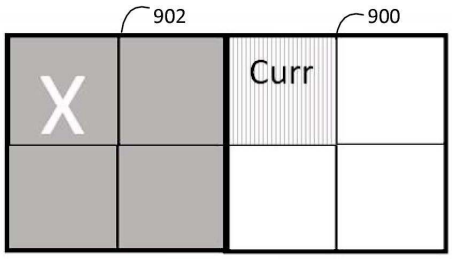
도면7


710
코딩된 비디오
시퀀스
엔트로피
디코더
771
잔차
디코더
773
인트라 예측
정보
인트라
디코더
772
인트라 예측
결과
인터 예측
정보
재구성
774
재구성된
화상
인터 디코더
780
인터 예측
결과
참조
화상

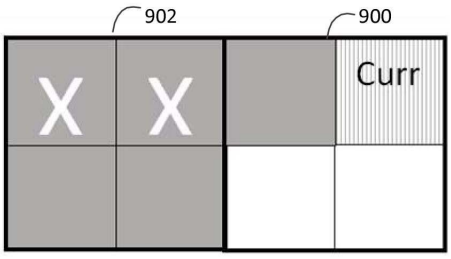
도면8



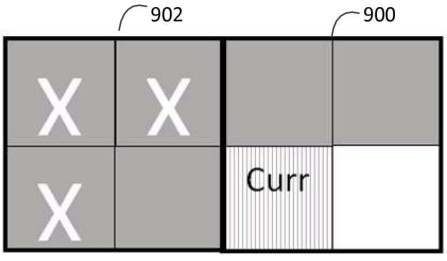
도면9a



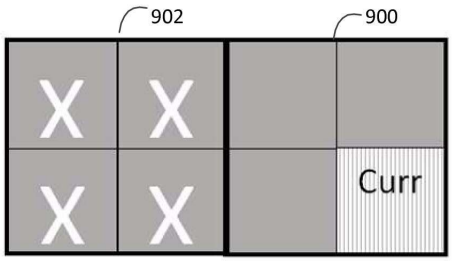
도면9b



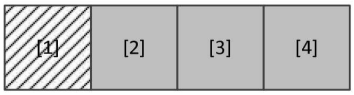
도면9c



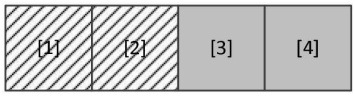
도면9d



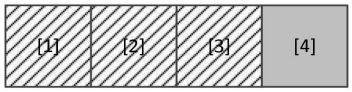
도면10a



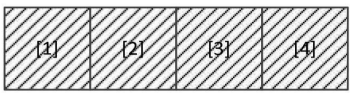
도면10b



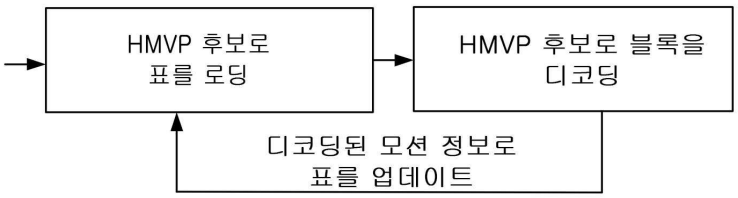
도면10c



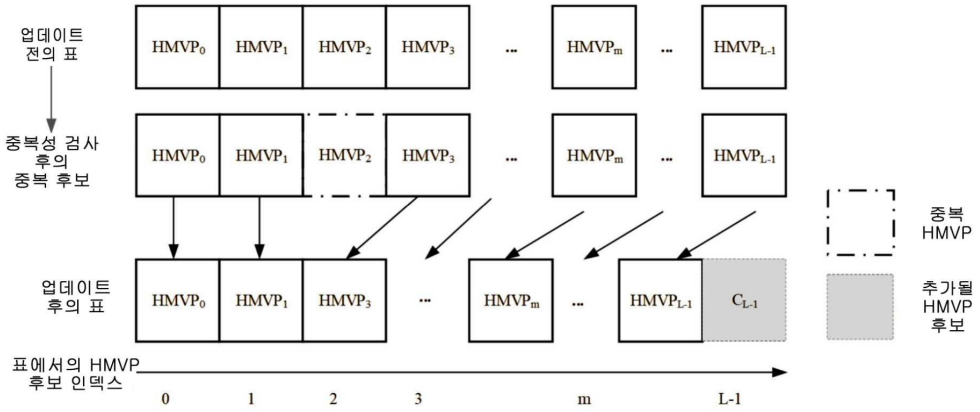
도면10d



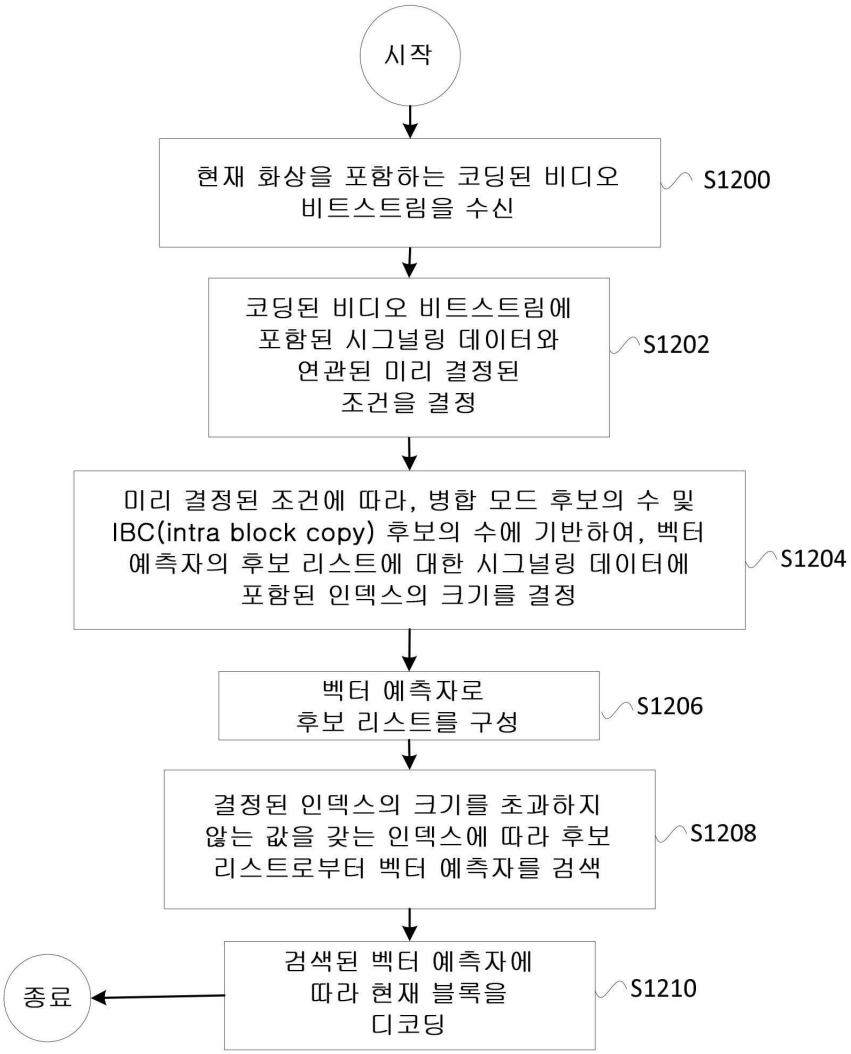
도면11a



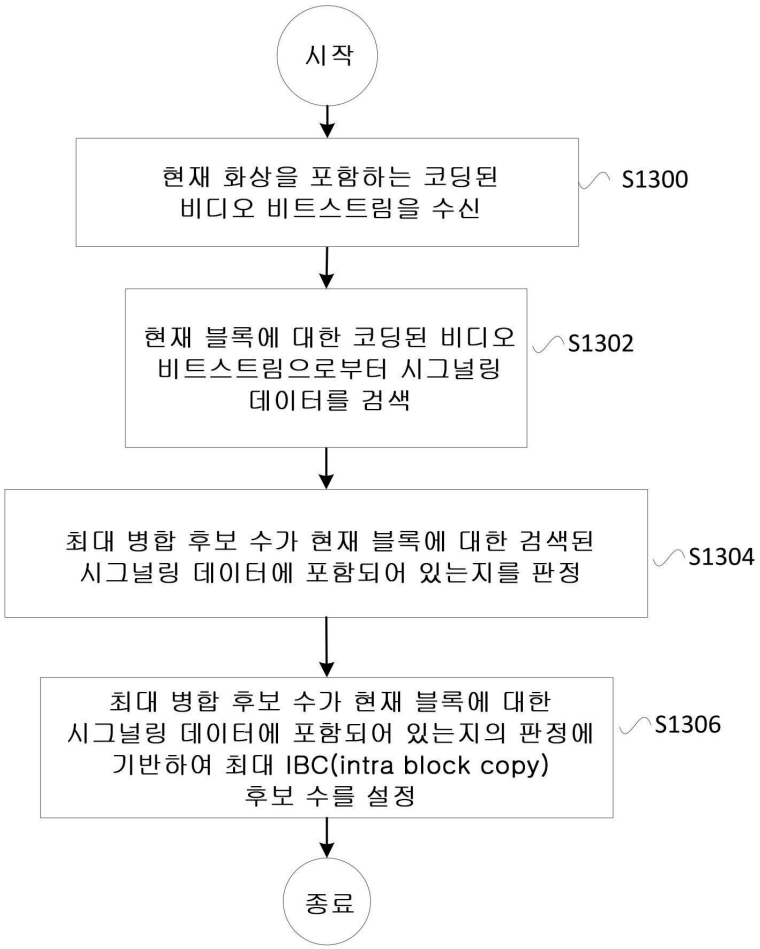
도면11b


업데이트 전의 표
중복성 검사 후의 중복 후보
업데이트 후의 표
표에서의 HMVP 후보 인덱스
HMVP0
HMVP1
HMVP2
HMVP3
HMVPm
HMVPL-1
CL-1
중복 HMVP
추가될 HMVP 후보
0
1
2
3
m
L-1

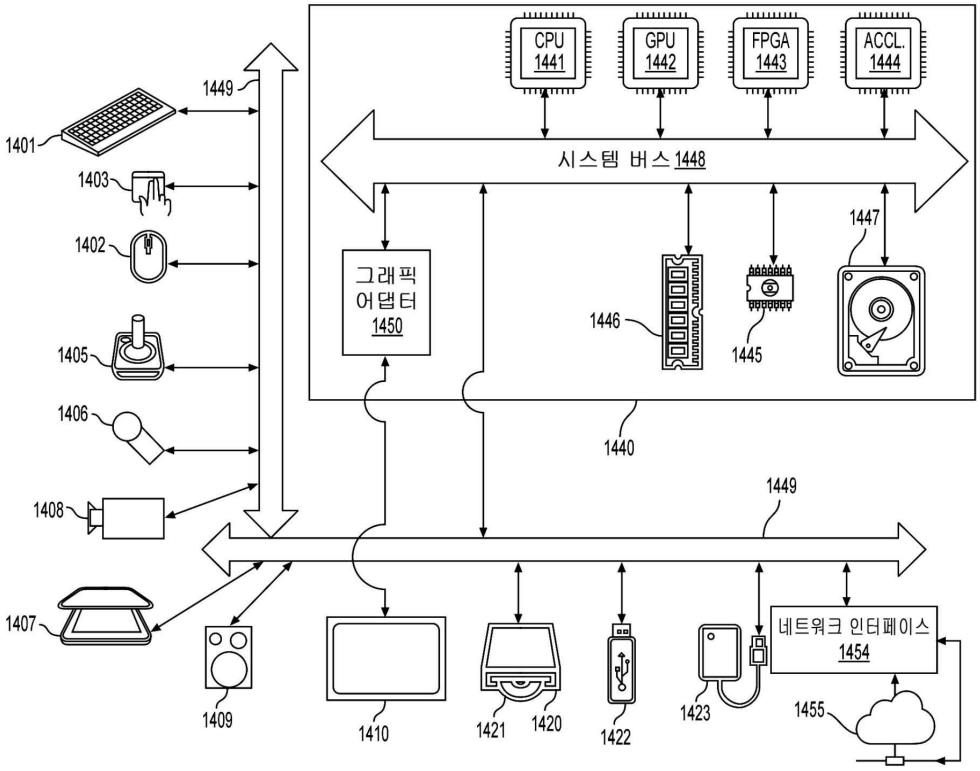
도면12


시작
현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신
S1200
코딩된 비디오 비트스트림에 포함된 시그널링 데이터와 연관된 미리 결정된 조건을 결정
S1202
미리 결정된 조건에 따라, 병합 모드 후보의 수 및 IBC(intra block copy) 후보의 수에 기반하여, 벡터 예측자의 후보 리스트에 대한 시그널링 데이터에 포함된 인덱스의 크기를 결정
S1204
벡터 예측자로 후보 리스트를 구성
S1206
결정된 인덱스의 크기를 초과하지 않는 값을 갖는 인덱스에 따라 후보 리스트로부터 벡터 예측자를 검색
S1208
검색된 벡터 예측자에 따라 현재 블록을 디코딩
S1210
종료

도면13


시작
현재 화상을 포함하는 코딩된 비디오 비트스트림을 수신
S1300
현재 블록에 대한 코딩된 비디오 비트스트림으로부터 시그널링 데이터를 검색
S1302
최대 병합 후보 수가 현재 블록에 대한 검색된 시그널링 데이터에 포함되어 있는지를 판정
S1304
최대 병합 후보 수가 현재 블록에 대한 시그널링 데이터에 포함되어 있는지의 판정에 기반하여 최대 IBC(intra block copy) 후보 수를 설정
S1306
종료

도면14


CPU
1441
GPU
1442
FPGA
1443
ACCL.
1444
시스템 버스 1448
그래픽
어댑터
1450
1446
1445
1447
1440
1449
1401
1403
1402
1405
1406
1408
1407
1409
1410
1421
1420
1422
1423
네트워크 인터페이스
1454
1455