

- (51) 국제특허분류(Int. Cl.)
H04N 19/13 (2014.01) *H04N 19/105* (2014.01)
H04N 19/117 (2014.01) *H04N 19/149* (2014.01)
H04N 19/176 (2014.01) *H04N 19/593* (2014.01)
H04N 19/70 (2014.01) *H04N 19/91* (2014.01)

(52) CPC특허분류
H04N 19/13 (2015.01)
H04N 19/105 (2015.01)

(21) 출원번호 10-2025-7011676(분할)

(22) 출원일자(국제) 2019년12월20일
심사청구일자 2025년04월09일

(62) 원출원 특허 10-2021-7017969
원출원일자(국제) 2019년12월20일
심사청구일자 2021년06월11일

(85) 번역문제출일자 2025년04월09일

(86) 국제출원번호 PCT/US2019/067992

(87) 국제공개번호 WO 2020/139774
국제공개일자 2020년07월02일

(30) 우선권주장
62/785,056 2018년12월26일 미국(US)
16/710,899 2019년12월11일 미국(US)

(71) 출원인
텐센트 아메리카 엘엘씨
미국 94306 캘리포니아주 팔로 알토 파크 블러바드 2747

(72) 발명자
쉬 명
미국 94306 캘리포니아주 팔로 알토 2747 파크 블러바드 텐센트 아메리카 엘엘씨 내

리 상
미국 94306 캘리포니아주 팔로 알토 2747 파크 블러바드 텐센트 아메리카 엘엘씨 내
(뒷면에 계속)

(74) 대리인
유미특허법인

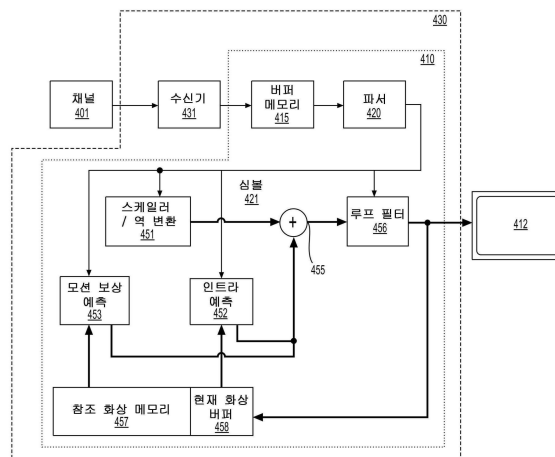
전체 청구항 수 : 총 11 항

(54) 발명의 명칭 비디오 코딩을 위한 방법 및 장치

(57) 요약

비디오 디코더에서 수행되는 비디오 디코딩 방법이 제공된다. 상기 방법에서, 선택스 엘리먼트의 빈의 코딩된 비트를 포함하는 비트스트림이 수신된다. 선택스 엘리먼트는 코딩된 화상에서 변환 스킵된 블록의 영역의 잔차에 대응한다. 컨텍스트 모델링이 수행되어 영역의 선택스 엘리먼트의 다수의 빈 각각에 대한 컨텍스트 모델을 결정한다. 컨텍스트 코딩된 선택스 엘리먼트의 빈의 수가 영역에 대해 설정된 컨텍스트 코딩된 빈의 최대 수를 초과하지 않는다. 선택스 엘리먼트의 다수의 빈의 코딩된 비트는 결정된 컨텍스트 모델을 기반으로 디코딩된다.

대표도



(52) CPC특허분류

HO4N 19/117 (2015.01)

HO4N 19/149 (2015.01)

HO4N 19/176 (2015.01)

HO4N 19/593 (2015.01)

HO4N 19/70 (2015.01)

HO4N 19/91 (2015.01)

(72) 발명자

수 소요종

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

리우 산

미국 94306 캘리포니아 팔로 알토 2747 파크 블러바드
텐센트 아메리카 엘엘씨 내

명세서

청구범위

청구항 1

비디오 인코더에서 수행되는 비디오 인코딩 방법으로서,

컨텍스트 모델링(context modeling)을 수행하여, 코딩된 화상 내 변환 스킵된 블록(transform skipped block)의 영역의 잔차(residue)에 대응하는 신택스 엘리먼트의 다수의 빈 각각에 대한 컨텍스트 모델을 결정하는 단계 - 컨텍스트 코딩되는 신택스 엘리먼트의 빈의 수는 상기 영역에 대해 설정된 컨텍스트 코딩된 빈의 최대 수를 초과하지 않음 -; 및

상기 결정된 컨텍스트 모델에 기초하여 신택스 엘리먼트의 빈의 코딩된 비트를 포함하는 비트 스트림을 생성하는 단계

를 포함하는 비디오 인코딩 방법.

청구항 2

제1항에 있어서,

상기 변환 스킵된 블록은 BDPCM(Block Differential Pulse-code Modulation)으로 코딩되고 복수의 영역으로 나뉘고,

상기 컨텍스트 코딩된 빈의 최대 수는 상기 영역에서 양자화된 잔차의 수와 임계 값 사이의 비교에 기초하여 결정되는, 비디오 인코딩 방법.

청구항 3

제1항에 있어서,

상기 결정된 컨텍스트 모델에 기초하여 신택스 엘리먼트의 빈의 코딩된 비트를 포함하는 비트 스트림을 생성하는 상기 단계는,

BDPCM에 따라, 상기 결정된 컨텍스트 모델에 기초하여 신택스 엘리먼트를 인코딩하는 단계를 포함하고,

상기 컨텍스트 코딩된 빈의 최대 수에 도달한 때, 신택스 엘리먼트의 나머지 빈들이 우회(bypass) 모델에 기초하여 인코딩되는, 비디오 인코딩 방법.

청구항 4

제1항에 있어서,

상기 변환 스킵된 블록의 상기 영역에 대한 컨텍스트 코딩된 빈의 최대 수는 $B \times A$ 이고, A는 상기 영역에서의 양자화된 잔차의 수이고, B는 A에 의해 지시되는 상기 영역에서의 양자화된 잔차의 수에 기초한 양수인, 비디오 인코딩 방법.

청구항 5

제1항에 있어서,

상기 변환 스킵된 블록의 상기 영역에 대한 컨텍스트 코딩된 빈의 최대 수는 $B \times A$ 이고, A는 상기 영역에서의 잔차의 수이고, B는 양수인, 비디오 인코딩 방법.

청구항 6

제1항에 있어서,

컨텍스트 코딩되는 신택스 엘리먼트의 빈의 수는 상기 변환 스킵된 블록의 영역에 대한 신택스 엘리먼트의 빈의 총 수와 같거나 그보다 작은, 비디오 인코딩 방법.

청구항 7

제5항에 있어서,

상기 생성하는 단계는,

상기 결정된 컨텍스트 모델에 기초하여 컨텍스트 코딩되는, 신텍스 엘리먼트의 상기 다수의 빈의 코딩된 비트를 생성하는 단계; 및

동등의 확률 모델(equal probability model)에 기초하여 상기 변환 스킵된 블록의 영역에 대해 컨텍스트 코딩되지 않는 신텍스 엘리먼트의 남은 총 수의 빈의 코딩된 비트를 생성하는 단계;

를 포함하는, 비디오 인코딩 방법.

청구항 8

제4항 또는 제5항에 있어서,

B는 정수인 것;

B는 분수인 것;

A가 상기 임계 값보다 큰 경우 B는 제1 수이고, A가 상기 임계 값보다 크지 않은 경우 B는 제2 수인 것; 및

상기 변환 스킵된 블록의 복수의 영역의 각각에 대한 컨텍스트 코딩된 빈의 최대 수가 $B \times A$ 인 것

중 적어도 하나인, 비디오 인코딩 방법.

청구항 9

제1항 내지 제7항 중 어느 한 항에 따른 비디오 인코딩 방법을 수행하도록 구성된 처리 회로를 포함하는 비디오 인코딩 장치.

청구항 10

실행시에 제1항 내지 제7항 중 어느 한 항에 따른 비디오 인코딩 방법을 실현하는 컴퓨터 판독 가능한 명령을 저장하는 비 일시적 컴퓨터 판독 가능한 저장 매체.

청구항 11

시각 미디어 데이터를 처리하는 방법으로서,

포맷 규칙에 따라 시각 미디어 파일과 시각 미디어 데이터의 비트 스트림 간 변환을 수행하는 단계를 포함하고,

상기 비트 스트림은 신텍스 엘리먼트의 빈의 코딩된 비트를 포함하고,

상기 포맷 규칙은, 코딩된 화상 내 변환 스킵된 블록(transform skipped block)의 영역의 잔차(residue)에 대응하는 신텍스 엘리먼트의 다수의 빈 각각에 대한 컨텍스트 모델을 결정하기 위해 컨텍스트 모델링이 수행되고, 컨텍스트 코딩되는 신텍스 엘리먼트의 빈의 수는 상기 영역에 대해 설정된 컨텍스트 코딩된 빈의 최대 수를 초과하지 않는 것을 규정하고,

상기 포맷 규칙은 또, 상기 결정된 컨텍스트 모델에 기초하여 신텍스 엘리먼트의 빈의 코딩된 비트를 포함하는 비트 스트림이 생성되는 것을 규정하는,

방법.

발명의 설명**기술 분야**

[0001]

본 출원은, 2018년 12월 26일에 출원되고 발명의 명칭이 "블록 차동 펄스 코드 변조에서의 컨텍스트 모델링 및 디블로킹 필터를 위한 제약 조건"인 미국 가출원 번호 제62/785,056호에 대한 우선권을 주장하는, 2019년 12월 11일에 출원되고 발명의 명칭이 "비디오 코딩을 위한 방법 및 장치"인 미국 특허 출원 번호 제16/710,899호에

대한 우선권을 주장하는 바이다.

[0002] 본 개시는 일반적으로 비디오 코딩과 관련된 실시 예를 설명한다.

배경 기술

[0003] 여기서 제공된 배경 설명은 일반적으로 본 개시의 컨텍스트(context)를 제시하기(present) 위한 것이다. 현재 지명된 발명자의 저작물이 이 배경 섹션에 설명된 한도 내에서, 출원 당시 선행 기술로 달리 자격이 없을 수 있는 설명의 측면은 명시적으로 또는 묵시적으로 본 개시의 선행 기술로 인정되지 않는다.

[0004] 비디오 코딩 및 디코딩은 모션 보상(motion compensation)이 있는 인터 화상 예측(inter-picture prediction)을 사용하여 수행될 수 있다. 압축되지 않은 디지털 비디오는 일련의 화상을 포함할 수 있으며, 각 화상은 예를 들어 1920×1080 휘도 샘플 및 연관된 색차 샘플의 공간 차원을 갖는다. 일련의 화상은 예를 들어 초당 60 화상 또는 60Hz의 고정 또는 가변 화상 레이트(비공식적으로 프레임 레이트(frame rate)라고도 함)를 가질 수 있다. 압축되지 않은 비디오는 상당한 비트레이트 요건을 가진다. 예를 들어, 샘플 당 8 비트에서의 1080p60 4: 2: 0 비디오(60Hz 프레임 레이트의 1920×1080 휘도 샘플 해상도)는 거의 1.5Gbit/s 대역폭을 필요로 한다. 이러한 비디오의 한 시간은 600GB 이상의 저장 공간을 필요로 한다.

[0005] 비디오 코딩 및 디코딩의 한 가지 목적은 압축을 통해 입력 비디오 신호에서의 중복성(redundancy)을 줄이는 것이다. 압축은 앞서 언급한 대역폭 또는 저장 공간 요건을 일부 경우에는 2배 이상의 규모로 줄이는 데 도움이 될 수 있다. 무손실 압축과 손실 압축 모두뿐만 아니라 이들의 조합이 사용될 수 있다. 무손실 압축은 압축된 원본 신호(original signal)로부터 원본 신호의 정확한 사본을 재구성할 수 있는 기술을 말한다. 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 정도는 애플리케이션에 따라 다르며; 예를 들어, 특정 소비자 스트리밍 애플리케이션의 사용자는 텔레비전 배포 애플리케이션의 사용자보다 더 높은 왜곡을 허용할 수 있다. 달성 가능한 압축 비율은 허락/허용(allowable/tolerable) 왜곡이 높을수록 압축 비율이 높아질 수 있는 것을 반영할 수 있다.

[0006] 비디오 인코더 및 디코더는 예를 들어 모션 보상, 변환(transform), 양자화(quantization) 및 엔트로피 코딩(entropy coding)을 포함하는 여러 범주의 기술을 활용할 수 있다.

[0007] 비디오 코덱 기술은 인트라 코딩(intra coding)이라는 기술을 포함할 수 있다. 인트라 코딩에서, 샘플 값은 이전에 재구성된 참조 화상으로부터 샘플 또는 기타 데이터를 참조하지 않고 표현된다. 일부 비디오 코덱에서, 화상이 공간적으로 샘플 블록으로 세분화된다. 모든 샘플 블록이 인트라 모드로 코딩될 때, 대응하는 화상은 인트라 화상일 수 있다. 인트라 화상 및 독립 디코더 리프레시 화상(independent decoder refresh picture)과 같은 그 파생물은 디코더 상태를 재설정하는 데 사용될 수 있으며, 따라서 코딩된 비디오 비트스트림 및 비디오 세션에서 제1 화상 또는 정지 이미지(still image)로 사용될 수 있다. 인트라 블록의 샘플은 변환에 노출될 수 있으며, 변환 계수는 엔트로피 코딩 전에 양자화될 수 있다. 인트라 예측은 사전 변환 도메인(pre-transform domain)에서 샘플 값을 최소화하는 기술일 수 있다. 일부 경우에, 변환 후 DC 값이 더 작고, AC 계수가 더 작을수록 엔트로피 코딩 후 블록을 표현하기 위해, 주어진 양자화 단계 크기에서 필요한 비트가 더 적다.

[0008] 예를 들어, MPEG-2 생성 코딩 기술에서 알려진 것과 같은 전통적인 인트라 코딩은 인트라 예측을 사용하지 않는다. 그러나, 일부 새로운 비디오 압축 기술은, 예를 들어 주변 샘플 데이터 및/또는 데이터의 공간적으로 이웃하고 그리고 디코딩 순서에서 선행하는 블록의 인코딩/디코딩 동안 획득된 메타 데이터로부터 시도하는 기술을 포함한다. 이러한 기술은 이후 "인트라 예측" 기술이라고 한다. 적어도 일부 경우에, 인트라 예측은 참조 화상이 아닌 재구성중인 현재 화상의 참조 데이터만 사용한다.

[0009] 다양한 형태의 인트라 예측이 있을 수 있다. 이러한 기술 중 하나 이상이 주어진 비디오 코딩 기술에서 사용될 수 있을 때, 사용 중인 기술은 인트라 예측 모드에서 코딩될 수 있다. 특정 경우에, 모드는 서브 모드(submode) 및/또는 파라미터를 가질 수 있으며, 이들은 개별적으로 코딩되거나 모드 코드워드(codeword)에 포함될 수 있다. 주어진 모드/서브 모드/파라미터 조합에 사용할 코드워드는 인트라 예측을 통한 코딩 효율성 이득에 영향을 미칠 수 있으며, 코드워드를 비트스트림으로 변환하는 데 사용되는 엔트로피 코딩 기술도 마찬가지이다.

[0010] 특정 모드의 인트라 예측이 H.264로 도입되었고 H.265에서 개선되었으며 JEM(joint exploration model), VVC(versatile video coding) 및 BMS(benchmark set)와 같은 최신 코딩 기술에서 더욱 개선되었다. 이미 사용 가능한 샘플에 속하는 인접 샘플 값을 사용하여 예측자 블록(predictor block)을 형성할 수 있다. 인접 샘플의

샘플 값은 방향에 따라 예측자 블록에 복사된다. 사용 방향에 대한 참조는 비트스트림에서 코딩되거나 자체적으로 예측될 수 있다.

[0011] 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(indicated) 방향으로 공간적으로 시프트된 후, 새로 재구성된 화상 또는 화상 부분의 예측에 사용된다. 일부 경우에, 참조 화상은 현재 재구성중인 화상과 동일할 수 있다. MV는 X와 Y의 2차원 또는 3차원을 가질 수 있으며, 세 번째는 사용중인 참조 화상을 지시한다(후자는 간접적으로 시간 차원일 수 있음).

[0012] 일부 비디오 압축 기술에서, 샘플 데이터의 특정 영역에 적용 가능한 MV는 다른 MV, 예를 들어 재구성중인 영역에 공간적으로 인접한 샘플 데이터의 다른 영역과 관련되며 그리고 디코딩 순서에서 해당 MV보다 선행하는 MV로부터 예측될 수 있다. 이렇게 하면 MV를 코딩하는 데 필요한 데이터 양을 크게 감소시킬 수 있으므로, 중복성을 제거하고 압축을 증가시킬 수 있다. 예를 들어, 카메라에서 유도된 입력 비디오 신호(자연(natural) 비디오라고 함)를 코딩할 때 단일 MV가 적용되는 영역보다 큰 영역이 유사한 방향으로 이동할 수 있는 통계적 가능성이 있기 때문에, MV 예측이 효과적으로 작동할 수 있으며, 따라서 일부 경우에 이웃 영역의 MV에서 유도된 유사한 모션 벡터를 사용하여 예측할 수 있다. 그 결과 주어진 영역에 대해 발견된 MV가 주변 MV로부터 예측된 MV와 유사하거나 동일하며, 결국 엔트로피 코딩후, 직접 MV를 코딩하면 사용되는 비트보다 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 주변 MV로부터 예측자를 계산할 때 반올림 에러(rounding error) 때문에, MV 예측 자체가 손실될 수 있다.

[0013] 다양한 MV 예측 메커니즘은 H.265/HEVC(ITU-T Rec. H.265, "고효율 비디오 코딩", 2016년 12월)에 설명되어 있다.

[0014] 도 1을 참조하면, 현재 블록(101)은 공간적으로 시프트된 동일한 크기의 이전 블록으로부터 예측 가능한 모션 검색 프로세스 동안 인코더에 의해 발견된 샘플을 포함한다. MV를 직접 코딩하는 대신, MV는 A0, A1 및 B0, B1, B2(각각 102 ~ 106)으로 표시됨) 5개의 주변 샘플 중 하나와 연관된 MV를 사용하여, 예를 들어, 가장 최근(디코딩 순서에서) 참조 화상으로부터 하나 이상의 참조 화상과 연관된 메타 데이터로부터 유도될 수 있다. H.265에서, MV 예측은 이웃 블록이 사용하는 동일한 참조 화상으로부터의 예측자를 사용할 수 있다.

발명의 내용

[0015] 본 개시의 측면은 디코더에서의 비디오 코딩을 위한 방법 및 장치를 제공한다. 일 실시 예에서, 디코더에서의 비디오 코딩 방법이 제공된다. 상기 비디오 코딩 방법에서, 신택스 엘리먼트(syntax element)의 빈(bin)의 코딩된 비트를 포함하는 비트스트림이 수신된다. 상기 신택스 엘리먼트는 코딩된 화상에서 변환 스킵된(transform skipped) 블록의 영역(region)의 잔차(residue)에 대응한다. 콘텍스트 모델링(Context modeling)이 수행되어 상기 영역의 신택스 엘리먼트의 다수의 빈 각각에 대한 콘텍스트 모델을 결정한다. 콘텍스트 코딩된 신택스 엘리먼트의 빈의 수가 상기 영역에 대해 설정된 콘텍스트 코딩된 빈의 최대 수를 초과하지 않는다. 상기 신택스 엘리먼트의 다수의 빈의 코딩된 비트는 상기 결정된 콘텍스트 모델을 기반으로 디코딩된다.

[0016] 일 실시 예에서, 상기 콘텍스트 코딩된 신택스 엘리먼트의 빈의 수는 상기 변환 스킵된 블록의 영역에 대한 신택스 엘리먼트의 빈의 총 수보다 작거나 같다.

[0017] 일 실시 예에서, 상기 변환 스킵된 블록의 영역에 대한 콘텍스트 코딩된 빈의 최대 수는 $B \times A$ 이고, 여기서 B는 양수이고 A는 상기 영역의 양자화된 잔차의 수이다.

[0018] 일 실시 예에서, 상기 변환 스킵된 블록의 영역에 대해 콘텍스트 코딩되지 않은 신택스 엘리먼트의 나머지 총 수의 빈의 코딩된 비트는 동등의 확률 모델(equal probability model)에 기반하여 디코딩된다.

[0019] 일 실시 예에서, B는 정수이다.

[0020] 일 실시 예에서, B는 분수이다.

[0021] 일 실시 예에서, B는 A에 따라 설정된다.

[0022] 일 실시 예에서, 상기 변환 스킵된 블록은 상기 영역만을 포함하고, A는 상기 변환 스킵된 블록의 $W \times H$ 와 같으며, 여기서 W는 상기 변환 스킵된 블록의 너비이고 H는 상기 변환 스킵된 블록의 높이이다.

[0023] 일 실시 예에서, 상기 변환 스킵된 블록은 상기 영역을 포함하는 복수의 영역으로 나뉘지고, 상기 변환 스킵된

블록의 복수의 영역 각각에 대한 컨텍스트 코딩된 bin의 최대 수는 $B \times A$ 이며, 여기서 B는 양수이고, A는 상기 복수의 영역 각각에서 양자화된 잔차의 수이다.

- [0024] 일 실시 예에서, 디코더에서 비디오 코딩 방법이 제공된다. 상기 비디오 코딩 방법에서, 코딩된 비디오 비트스트림이 수신된다. 지시자(indicator)가 수신된다. 상기 지시자는 코딩된 비디오 비트스트림의 현재 블록 및 이웃 블록 중 적어도 하나가 BDPCM(Block Differential Pulse-code Modulation)으로 코딩되는지를 지시한다. 상기 현재 블록은 상기 이웃 블록에 인접해 있다. 상기 현재 블록과 상기 이웃 블록 중 적어도 하나가 BDPCM으로 코딩된 것으로 지시될 때, 상기 현재 블록의 현재 서브 블록과 상기 현재 블록의 이웃 서브 블록 사이의 경계에 적용되는 경계 강도(boundary strength)가 결정되고, 상기 현재 블록의 현재 서브 블록과 상기 이웃 블록의 이웃 서브 블록 사이의 경계에 대해 결정된 경계 강도에 따라, 디블로킹 필터를 사용하여 디블로킹을 수행한다.
- [0025] 일 실시 예에서, 상기 이웃 서브 블록은 BDPCM으로 코딩되지 않는다.
- [0026] 일 실시 예에서, 상기 이웃 서브 블록이 BDPCM으로 코딩될 때, 상기 디블로킹 필터의 경계 강도는 1 또는 2로 결정된다.
- [0027] 일 실시 예에서, 상기 현재 블록 및 상기 이웃 서브 블록이 BDPCM으로 코딩될 때, 상기 디블로킹은 상기 현재 서브 블록과 상기 이웃 서브 블록 사이의 양자화 파라미터 차이가 임계 값보다 크다는 결정에 기반하여, 상기 현재 블록의 현재 서브 블록과 상기 이웃 블록의 이웃 서브 블록 사이의 경계에서 수행된다.
- [0028] 일 실시 예에서, 상기 현재 블록 및 상기 이웃 서브 블록이 BDPCM으로 코딩될 때, 상기 디블로킹은 상기 현재 서브 블록과 상기 이웃 서브 블록은 서로 다른 BDPCM 예측 모드로 코딩된다는 결정에 기반하여, 상기 현재 블록의 현재 서브 블록과 상기 이웃 블록의 이웃 서브 블록 사이의 경계에서 수행된다.
- [0029] 일 실시 예에서, 상기 현재 블록 및 상기 이웃 서브 블록이 BDPCM으로 코딩될 때, 디블로킹은 현재 서브 블록과 이웃 서브 블록 중 적어도 하나는 0이 아닌 변환 계수를 갖는다는 결정에 기반하여, 다음과 같은 결정에 기반하여, 현재 블록의 현재 서브 블록과 이웃 블록의 이웃 서브 블록 사이의 경계에서 수행된다.
- [0030] 본 개시의 측면은 또한 컴퓨터에 의해 실행될 때 컴퓨터가 상기 방법 중 임의의 것을 수행하게 하는 명령을 저장하는, 컴퓨터가 판독 가능한 비 일시적 저장 매체(medium)를 제공한다.

도면의 간단한 설명

- [0031] 개시된 주제의 추가 특징, 본질 및 다양한 이점은 다음의 상세한 설명 및 첨부 도면으로부터 더욱 명백해질 것이다.
- 도 1은 하나의 예에서 현재 블록 및 그 주변 공간 병합 후보의 개략도이다.
- 도 2는 일 실시 예에 따른 통신 시스템의 단순화된 블록도의 개략도이다.
- 도 3은 일 실시 예에 따른 통신 시스템의 단순화된 블록도의 개략도이다.
- 도 4는 일 실시 예에 따른 디코더의 단순화된 블록도의 개략도이다.
- 도 5는 일 실시 예에 따른 인코더의 단순화된 블록도의 개략도이다.
- 도 6은 다른 실시 예에 따른 인코더의 블록도를 도시한다.
- 도 7은 다른 실시 예에 따른 디코더의 블록도를 도시한다.
- 도 8은 본 개시의 실시 예에 따른 경계 강도 값을 결정하는 예시적인 프로세스의 흐름도를 도시한다.
- 도 9a는 일 실시 예에 따른 블록 차동 펄스 코드 변조(block differential pulse-code modulation, BDPCM) 코딩된 블록의 예를 도시한다.
- 도 9b는 일 실시 예에 따른 BDPCM 코딩된 블록의 예를 도시한다.
- 도 10a는 일 실시 예에 따른 예시적인 컨텍스트 기반 적응성 이진 산술 코딩(context-based adaptive binary arithmetic coding, CABAC) 기반 엔트로피 인코더를 도시한다.
- 도 10b는 일 실시 예에 따른 예시적인 CABAC 기반 엔트로피 디코더를 도시한다.
- 도 11은 일 실시 예에 따른 예시적인 CABAC 기반 엔트로피 인코딩 프로세스를 도시한다.

도 12는 일 실시 예에 따른 예시적인 CABAC 기반 엔트로피 디코딩 프로세스를 도시한다.

도 13은 일 실시 예에 따른 엔트로피 디코딩 프로세스를 개략적으로 설명하는 흐름도를 도시한다.

도 14는 일 실시 예에 따른 디블로킹 필터링 프로세스를 개략적으로 설명하는 흐름도를 도시한다.

도 15는 일 실시 예에 따른 컴퓨터 시스템의 개략도이다.

발명을 실시하기 위한 구체적인 내용

I. 비디오 코딩 인코더 및 디코더(Video Coding Encoder and Decoder)

도 2는 본 개시의 일 실시 예에 따른 통신 시스템(200)의 단순화된 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 및 220)를 포함한다. 도 2 예에서, 제1 쌍의 단말 디바이스(210 및 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 복원하고, 복원된 비디오 데이터에 따라 비디오 화상을 디스플레이할 수 있다. 단방향 데이터 전송은 미디어(media) 서버(serving) 애플리케이션 등에서 일반적일 수 있다.

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 및 240)를 포함한다. 데이터의 양방향 전송의 경우, 예를 들어, 단말 디바이스들(230 및 240)의 각 단말 디바이스는, 네트워크(250)를 통해 단말 디바이스들(230 및 240) 중 다른 단말 디바이스로의 전송을 위해 비디오 데이터(예: 단말 디바이스에 의해 캡처된 비디오 화상의 스트림)를 코딩할 수 있다. 단말 디바이스들(230 및 240)의 각 단말 디바이스는 또한 단말 디바이스들(230 및 240) 중 다른 단말 디바이스가 전송한 코딩된 비디오 데이터를 수신할 수 있으며, 코딩된 비디오 데이터를 디코딩하여 비디오 화상을 복원할 수 있으며, 복원된 비디오 데이터에 따라 액세스 가능한 디스플레이 디바이스에 비디오 화상을 디스플레이할 수 있다.

도 2 예에서, 단말 디바이스(210, 220, 230, 240)는 서버, 개인용 컴퓨터 및 스마트 폰으로 예시될 수 있지만, 본 개시의 원리는 이에 제한되지 않을 수 있다. 본 개시의 실시 예는 랩톱 컴퓨터, 태블릿 컴퓨터, 미디어 플레이어 및/또는 전용 화상 회의 장비를 갖는 애플리케이션을 찾는다. 네트워크(250)는 예를 들어 유선(wireline)(wired) 및/또는 무선 통신 네트워크를 포함하며, 단말 디바이스들(210, 220, 230 및 240) 간에 코딩된 비디오 데이터를 전달하는 임의의 수의 네트워크를 나타낸다(represent). 통신 네트워크(250)는 회선 교환 및/또는 패킷 교환 채널에서 데이터를 교환할 수 있다. 대표적인 네트워크는 원격통신(telecommunications) 네트워크, 근거리 통신망, 광역 통신망 및/또는 인터넷을 포함한다. 본 논의의 목적을 위해, 네트워크(250)의 아키텍처 및 토폴로지는 아래에서 설명되지 않는 한 본 개시의 작동에 중요하지 않을 수 있다.

도 3은 개시된 주제에 대한 애플리케이션의 예로서 스트리밍 환경에서의 비디오 인코더 및 비디오 디코더의 배치를 예시한다. 개시된 주제는 예를 들어, 화상 회의, 디지털 TV, CD, DVD, 메모리 스틱 등을 포함하는 디지털 미디어에 압축된 비디오의 저장 등을 포함하는 다른 비디오 인에이블드(enabled) 애플리케이션(video enabled application)에도 동일하게 적용될 수 있다.

스트리밍 시스템은 비디오 소스(301), 예를 들어 디지털 카메라를 포함할 수 있는 캡처 서버 시스템(313)을 포함할 수 있으며, 이는 예를 들어 압축되지 않은 비디오 화상의 스트림(302)을 생성한다. 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 볼륨을 강조하기 위해 굵은 선으로 표시된 비디오 화상의 스트림(302)은, 비디오 소스(301)에 결합된(coupled) 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 표시된, 인코딩된 비디오 데이터(304)(또는 인코딩된 비디오 비트스트림(304))는 향후 사용을 위해 스트리밍 서버(305)에 저장될 수 있다. 도 3의 클라이언트 서버 시스템(306 및 308)과 같은 하나 이상의 스트리밍 클라이언트 서버 시스템은, 스트리밍 서버(305)에 액세스하여 인코딩된 비디오 데이터(304)의 사본(307 및 309)을 검색할(retrieve) 수 있다. 클라이언트 서버 시스템(306)은

예를 들어, 전자 디바이스(330)에서 비디오 디코더(310)를 포함할 수 있다. 비디오 디코더(310)는 인코딩된 비디오 데이터의 입력되는 사본(incoming copy)(307)을 디코딩하고, 디스플레이(312)(예: 디스플레이 화면) 또는 다른 렌더링 디바이스(도시되지 않음)에서 렌더링될 수 있는 비디오 화상의 출력 스트림(311)을 생성한다. 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 권고 H.265가 있다. 일 예에서, 개발중인 비디오 코딩 표준은 비공식적으로 VVC(Versatile Video Coding)로 알려져 있다. 개시된 주제는 VVC의 콘텍스트에서 사용될 수 있다.

[0038] 전자 디바이스(320 및 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)도 비디오 인코더(도시되지 않음)를 포함할 수 있다.

[0039] 도 4는 본 개시의 일 실시 예에 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 3 예에서 비디오 디코더(310) 대신에 사용될 수 있다.

[0040] 수신기(431)는 비디오 디코더(410)에 의해 디코딩될 하나 이상의 코딩된 비디오 시퀀스를 수신할 수 있고; 동일하거나 다른 실시 예에서, 한 번에 하나의 코딩된 비디오 시퀀스, 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 포워딩될 수 있는 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터(ancillary data) 스트림과 함께 인코딩된 비디오 데이터를 수신할 수 있다. 수신기(431)는 코딩된 비디오 시퀀스를 다른 데이터와 분리할 수 있다. 네트워크 지터(jitter)를 방지하기(combat) 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 경우, 예를 들어 네트워크 지터를 방지하기 위해 비디오 디코더(410) 외부에 버퍼 메모리(도시되지 않음)가 있을 수 있으며, 추가로 예를 들어, 플레이아웃 타이밍(playout timing)을 처리하기 위해 비디오 디코더(410) 내부에 다른 버퍼 메모리(415)가 있을 수 있다. 수신기(431)가 충분한 대역폭과 제어성(controllability)을 갖는 저장/포워딩 디바이스 또는 동시동기식 네트워크(isosynchronous network)로부터 데이터를 수신할 때, 버퍼 메모리(415)는 필요하지 않거나 작을 수 있다. 인터넷과 같은 최선형 패킷 네트워크에서 사용하기 위해, 버퍼 메모리(415)가 필요할 수 있고, 비교적 클 수 있고 적응형 크기가 유리할 수 있으며, 그리고 운영 체제 또는 비디오 디코더(410) 외부에서 유사한 엘리먼트에서 적어도 부분적으로 구현될 수 있다.

[0041] 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, 그리고 잠재적으로, 도 4에 도시된 바와 같이, 전자 디바이스(430)의 내부 부분이 아니지만 전자 디바이스(430)에 결합될 수 있는 렌더(render) 디바이스(412)(예: 디스플레이 화면)와 같은 렌더링 디바이스를 제어하기 위한 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 표준에 따를 수 있으며, 가변 길이 코딩, 호프만(Huffman) 코딩, 콘텍스트 민감도가 있거나 없는 산술 코딩 등을 포함한 다양한 원리를 따를 수 있다. 파서(420)는 그룹에 대응하는 적어도 하나의 파라미터에 기반하여, 비디오 디코더 내의 픽셀들의 서브 그룹 중 적어도 하나에 대한 서브 그룹 파라미터 세트를 코딩된 비디오 시퀀스로부터 추출할 수 있다. 서브 그룹은 GOP(Group of Pictures), 화상, 타일(tile), 슬라이스(slice), 매크로 블록, CU(Coding Unit), 블록, TU(Transform Unit), PU(Prediction Unit) 등을 포함할 수 있다. 파서(420)는 또한 변환 계수, 양자화기(quantizer) 파라미터 값, 모션 벡터(motion vector) 등과 같은 코딩된 비디오 시퀀스 정보로부터 추출할 수 있다.

[0042] 파서(420)는 버퍼 메모리(415)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩/파싱 작동을 수행하여 심볼(421)을 생성할 수 있다.

[0043] 심볼의 재구성(421)은 코딩된 비디오 화상 또는 그 일부(예: 인터 및 인트라 화상, 인터 및 인트라 블록)의 유형 및 기타 팩터에 따라 여러 개의 상이한 유닛을 포함할 수 있다. 파서(420)에 의해 코딩된 비디오 시퀀스로부

터 과싱된 서브 그룹 제어 정보에 의해 어떤 유닛이 관련되고 그리고 어떻게 제어될 수 있다. 파서(420)와 아래의 다중 유닛 사이의 이러한 서브 그룹 제어 정보의 흐름(flow)은 명료성(clarity)을 위해 표시되지 않는다.

[0044] 이미 언급된 기능 블록 이상으로, 비디오 디코더(410)는 아래에 설명된 바와 같이 개념적으로 다수의 기능 유닛으로 세분될 수 있다. 상업적 제약 하에서 작동하는 실제 구현에서, 이러한 유닛 중 다수는 서로 밀접하게 상호 작용하며 적어도 부분적으로는 서로 통합될 수 있다. 그러나 개시된 주제를 설명하기 위해서는 아래의 기능 유닛으로 개념적으로 세분화하는 것이 적절하다.

[0045] 제1 유닛은 스케일러(scaler)/역 변환(inverse transform) 유닛(451)이다. 스케일러/역 변환 유닛(451)은 사용할 변환, 블록 크기, 양자화 팩터, 양자화 스케일링 매트릭스 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.

[0046] 일부 경우에, 스케일러/역 변환(451)의 출력 샘플은 인트라 코딩된 블록 즉, 이전에 재구성된 화상으로부터의 예측 정보(predictive information)를 사용하지 않지만, 현재 블록의 이전에 재구성된 부분으로부터의 예측 정보를 사용할 수 있는 블록에 관계될(pertain) 수 있다. 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) 이미 재구성된 주변 정보를 사용하여, 재구성중인 블록과 동일한 크기 및 모양의 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.

[0047] 다른 경우에, 스케일러/역 변환 유닛(451)의 출력 샘플은 인트라 코딩되고 잠재적으로 모션 보상된 블록에 관계될 수 있다. 이러한 경우, 모션 보상 예측 유닛(453)은 참조 화상 메모리(457)에 액세스하여, 예측에 사용되는 샘플을 페치할 수 있다. 블록에 관계된 심볼(421)에 따라 페치된 샘플을 모션 보정한 후, 출력 샘플 정보를 생성하기 위해, 이러한 샘플은 애그리게이터(455)에 의해 스케일러/역 변환 유닛(451)의 출력(이 경우에는 잔차 샘플 또는 잔차 신호라고 함)에 추가될 수 있다. 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 이용 가능하다. 모션 보상은 또한 서브 샘플 정확한 모션 벡터가 사용 중일 때 참조 화상 메모리(457)로부터 페치된 샘플 값의 보간, 모션 벡터 예측 메커니즘 등을 포함할 수 있다.

[0048] 애그리게이터(455)의 출력 샘플은 루프 필터 유닛(456)에서 다양한 루프 필터링 기술을 적용받을 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 (디코딩 순서에서) 이전 부분을 디코딩하는 동안 획득된 메타 정보에 반응할 수 있을 뿐만 아니라, 이전에 재구성되고 루프 필터링된 샘플 값에 반응할 수도 있다.

[0049] 루프 필터 유닛(456)의 출력은 렌더 디바이스(412)로 출력될 수 있을 뿐만 아니라 미래의 인트라 화상 예측에 사용하기 위해 참조 화상 메모리(457)에 저장될 수 있는 샘플 스트림일 수 있다.

[0050] 완전히 재구성되면 특정 코딩된 화상은 미래의 예측을 위한 참조 화상으로 사용될 수 있다. 예를 들어, 현재 화상에 대응하는 코딩된 화상이 완전히 재구성되고 코딩된 화상이 참조 화상으로 식별되면(예를 들어 파서(420)에 의해), 현재 화상 버퍼(458)는 참조 화상 메모리(457)의 일부가 될 수 있으며, 새로운(fresh) 현재 화상 버퍼는 다음 코딩된 화상의 재구성을 시작하기 전에 재할당될 수 있다.

[0051] 비디오 디코더(410)는 ITU-T Rec. H.265와 같은 표준에서 미리 정해진 비디오 압축 기술에 따라 디코딩 작동을 수행할 수 있다. 코딩된 비디오 시퀀스가 비디오 압축 기술 또는 표준의 선택스와 비디오 압축 기술 또는 표준에 문서화된 프로파일(profile)을 모두 준수한다는 점에서, 코딩된 비디오 시퀀스가 사용중인 비디오 압축 기술 또는 표준에 의해 지정된 선택스를 따를 수 있다. 특히, 프로파일은 비디오 압축 기술 또는 표준에서 사용할 수 있는 모든 도구로부터, 특정 도구를 해당 프로파일 하에서 사용할 수 있는 유일한 도구로 선택할 수 있다. 또한, 규정 준수(compliance)에 필요한 것은 코딩된 비디오 시퀀스의 복잡성이 비디오 압축 기술 또는 표준의 수준에 정의된 범위 내에 있어야 한다는 것이다. 일부 경우에, 레벨은 최대 화상 크기, 최대 프레임 레이트, 최대 재구성 샘플 레이트(예를 들어, 초당 메가 샘플로 측정됨), 최대 참조 화상 크기 등을 제한한다. 레벨별로 설정된

제한은 일부 경우에, 코딩된 비디오 시퀀스에서 시그널링된 HRD(Hypothetical Reference Decoder) 사양 및 HRD 버퍼 관리를 위한 메타 데이터를 통해 추가로 제한될 수 있다.

[0052] 일 실시 예에서, 수신기(431)는 인코딩된 비디오와 함께 추가(중복) 데이터를 수신할 수 있다. 추가 데이터(additional data)는 코딩된 비디오 시퀀스(들)의 일부로 포함될 수 있다. 추가 데이터는 데이터를 적절하게 디코딩하거나 및/또는 원본 비디오 데이터를 더 정확하게 재구성하기 위해 비디오 디코더(410)에 의해 사용될 수 있다. 추가 데이터는 예를 들어 시간적, 공간적 또는 신호 잡음비(signal noise ratio, SNR) 향상 계층, 중복 슬라이스, 중복 화상, 순방향 에러 정정 코드(forward error correction code) 등의 형태일 수 있다.

[0053] 도 5는 본 개시의 일 실시 예에 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 3 예에서의 비디오 인코더(303) 대신에 사용될 수 있다.

[0054] 비디오 인코더(503)는 비디오 인코더(503)에 의해 코딩될 비디오 이미지(들)를 캡처할 수 있는 비디오 소스(501)(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.

[0055] 비디오 소스(501)는 임의의 적절한 비트 깊이(예를 들어, 8 비트, 10 비트, 12 비트, ...), 임의 색 공간(예를 들어, BT.601 Y CrCb, RGB, ...) 및 임의 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 시스템에서, 비디오 소스(501)는 미리 준비된 비디오를 저장하는 저장 디바이스일 수 있다. 화상 회의 시스템에서, 비디오 소스(501)는 로컬 이미지 정보를 비디오 시퀀스로 캡처하는 카메라일 수 있다. 비디오 데이터는 순차적으로 볼 때, 모션을 부여하는(impart) 복수의 개별 화상으로 제공될 수 있다. 화상 자체는 픽셀의 공간 어레이로 구성될 수 있으며, 각 픽셀은 사용중인 샘플링 구조, 색 공간 등에 따라 하나 이상의 샘플을 포함할 수 있다. 당업자는 픽셀과 샘플 사이의 관계를 쉽게 이해할 수 있다. 아래 설명은 샘플에 중점을 둔다.

[0056] 일 실시 예에 따르면, 비디오 인코더(503)는 실시간으로 또는 애플리케이션에 의해 요구되는 임의의 다른 시간 제약하에 소스 비디오 시퀀스의 화상을 코딩된 비디오 시퀀스(543)로 코딩하고 압축할 수 있다. 적절한 코딩 속도를 적용하는 것은 컨트롤러(550)의 기능 중 하나이다. 일부 실시 예에서, 컨트롤러(550)는 후술되는 바와 같이 다른 기능 유닛을 제어하고 다른 기능 유닛에 기능적으로 결합된다. 명료성을 위해 이 결합(coupling)은 묘사되지 않는다. 컨트롤러(550)에 의해 설정되는 파라미터는 레이트 제어 관련 파라미터(화상 스킵, 양자화기, 레이트 왜곡 최적화 기술의 람다(lambda) 값, ...), 화상 크기, GOP(group of pictures) 레이아웃, 최대 모션 벡터 검색 범위 및 등을 포함할 수 있다. 컨트롤러(550)는 특정 시스템 설계에 최적화된 비디오 인코더(503)에 관계된 다른 적절한 기능을 갖도록 구성될 수 있다.

[0057] 일부 실시 예에서, 비디오 인코더(503)는 코딩 루프에서 작동하도록 구성된다. 지나치게 단순화된 설명으로서, 일 예에서, 코딩 루프는 소스 코더(530)(예: 코딩된 입력 화상 및 참조 화상(들)에 기반하여 심볼 스트림과 같은 심볼 및 참조 화상(들)을 생성하는 것을 담당함) 및 비디오 인코더(503)에 내장된(로컬) 디코더(533)를 포함할 수 있다. 디코더(533)는 심볼을 재구성하여(심볼과 코딩된 비디오 비트스트림 간의 압축은 개시된 주제에서 고려되는 비디오 압축 기술에서 무손실이기 때문에)(원격) 디코더가 생성하는 것과 유사한 방식으로 샘플 데이터를 생성한다. 재구성된 샘플 스트림(샘플 데이터)은 참조 화상 메모리(534)에 입력된다. 심볼 스트림의 디코딩이 디코더 위치(로컬 또는 원격)와 무관하게 비트-정확한(bit-exact) 결과로 이어지므로, 참조 화상 메모리(534)의 콘텐츠는 또한 로컬 인코더와 원격 인코더 간에 비트 정확하다. 다시 말해서, 인코더의 예측 부분은 디코딩 중에 예측을 사용할 때 디코더가 "보는(see)" 것과 정확히 동일한 샘플 값을 참조 화상 샘플링으로 "본다(see)". 참조 화상 동기성(synchronicity)의 이 기본 원리(채널 에러 등으로 동기성을 유지할 수 없으면, 결과적으로 드리프트(drift)가 발생)는 일부 관련 기술에서도 사용된다.

[0058] "로컬" 디코더(533)의 작동은 비디오 디코더(410)와 같은 "원격" 디코더의 작동과 동일할 수 있으며, 이는 이미 도 4와 관련하여 상세하게 설명되었다. 간단히 도 4를 참조하면, 심볼이 이용 가능하고, 심볼을 엔트로피 코더(545) 및 파서(420)에 의해 코딩된 비디오 시퀀스로 인코딩/디코딩하는 것이 무손실일 수 있으므로, 버퍼 메모리(415)를 포함하는 비디오 디코더(410)의 엔트로피 디코딩 부분 및 파서(420)는 로컬 디코더(533)에서 완전히 구현되지 않을 수 있다.

[0059] 이 시점에서 이루어질 수 있는 관찰은 디코더에 존재하는 파싱/엔트로피 디코딩을 제외한 모든 디코더 기술이

반드시 대응하는 인코더에 실질적으로 동일한 기능적 형태로 존재해야 한다는 것이다. 이러한 이유로, 개시된 주제는 디코더 작동에 중점을 둔다. 인코더 기술에 대한 설명은 포괄적으로 설명된 디코더 기술의 반대이므로 생략될 수 있다. 특정 영역에서만 더 자세한 설명이 필요하며 아래에서 제공된다.

- [0060] 작동 동안, 일부 예에서, 소스 코더(530)는 "참조 화상"으로 지정된 비디오 시퀀스로부터 하나 이상의 이전에 코딩된 화상을 참조하여 입력 화상을 예측적으로 코딩하는 모션 보상된 예측 코딩을 수행할 수 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.
- [0061] 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 프로세스일 수 있다. 코딩된 비디오 데이터가 비디오 디코더(도 5에 도시되지 않음)에서 디코딩될 수 있을 때, 재구성된 비디오 시퀀스는 일반적으로 약간의 에러가 있는 소스 비디오 시퀀스의 복제본(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 디코더(전송 에러 없음)에 의해 획득될 재구성된 참조 화상으로서, 공통 콘텐츠를 갖는 재구성된 참조 화상의 사본을 국부적으로 저장할 수 있다.
- [0062] 예측자(predictor)(535)는 코딩 엔진(532)에 대한 예측 검색을 수행할 수 있다. 즉, 코딩된 신규 화상의 경우, 예측자(535)는 참조 화상 메모리(534)에서, 신규 화상에 대한 적절한 예측 참조 역할을 할 수 있는, 샘플 데이터(후보 참조 픽셀 블록) 또는 참조 화상 모션 벡터, 블록 모양 등과 같은 특정 메타 데이터를 검색할 수 있다. 예측자(535)는 적절한 예측 참조를 찾기 위해 샘플 블록 대 픽셀 블록 단위(sample block-by-pixel block basis)로 작동할 수 있다. 일부 경우에, 예측자(535)에 의해 획득된 검색 결과에 의해 결정된 바와 같이, 입력 화상은 참조 화상 메모리(534)에 저장된 다수의 참조 화상으로부터 추출된 예측 참조를 가질 수 있다.
- [0063] 컨트롤러(550)는 예를 들어, 비디오 데이터를 인코딩하기 위해 사용되는 파라미터 및 서브 그룹 파라미터의 설정을 포함하는, 소스 코더(530)의 코딩 작동을 관리할 수 있다.
- [0064] 진술한 모든 기능 유닛의 출력은 엔트로피 코더(545)에서 엔트로피 코딩을 받을 수 있다. 엔트로피 코더(545)는 호프만 코딩, 가변 길이 코딩, 산술 코딩 등과 같은 기술에 따라 심볼을 무손실 압축하는 것에 의해, 다양한 기능 유닛에 의해 생성된 심볼을 코딩된 비디오 시퀀스로 변환한다.
- [0065] 송신기(540)는 엔트로피 코더(545)에 의해 생성된 코딩된 비디오 시퀀스(들)를 버퍼링하여, 인코딩된 비디오 데이터를 저장할 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는 통신 채널(560)을 통한 전송을 준비할 수 있다. 송신기(540)는 비디오 코더(503)로부터 코딩된 비디오 데이터를 전송될 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터(ancillary data) 스트림(소스는 도시되지 않음)과 병합할 수 있다.
- [0066] 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 지정될 수 있다.
- [0067] 인트라 화상(Intra Picture)(I 화상)은 시퀀스의 다른 화상을 예측의 소스로 사용하지 않고 코딩 및 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱은 예를 들어 "IDR(Independent Decoder Refresh)" 화상을 포함하여 다양한 유형의 인트라 화상을 허용한다. 당업자는 I 화상의 이러한 변형 및 각각의 애플리케이션 및 특징을 알고 있다.
- [0068] 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 코딩 및 디코딩될 수 있는 것일 수 있다.
- [0069] 양방향 예측 화상(bi-directionally predictive picture)(B 화상)은 각 블록의 샘플 값을 예측하기 위해 최대 2개의 모션 벡터 및 참조 인덱스를 사용하는 인트라 예측 또는 인터 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다. 유사하게, 다중 예측 화상은 단일 블록의 재구성을 위해 2개 이상의 참조 화상과 관련 메타 데이터를 사용할 수 있다.
- [0070] 소스 화상은 보통 공간적으로 복수의 샘플 블록(예를 들어, 각각 4×4 , 8×8 , 4×8 또는 16×16 샘플 블록)으로 세분화되고 블록 단위로 코딩될 수 있다. 블록은 블록의 각각의 화상에 적용된 코딩 할당에 의해 결정되는 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 (공간 예측 또는 인트라 예측) 코딩될 수 있다. P 화상의 픽셀 블록은 하나의 이전에 코딩된 참조 화상을 참조하여, 공간 예측을 통해 또는 시간적 예측을 통해 예측적으로 코딩될 수 있다. B 화상의 블록은 하나 또는 2개의 이전에 코딩된 참조 화상을 참조하여, 공간 예측 또는 시간적 예측을 통해 예측적으로 코딩될 수 있다.

[0071] 비디오 인코더(503)는 ITU-T Rec. H.265와 같은 미리 정해진 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 수 있다.

[0072] 일 실시 예에서, 송신기(540)는 인코딩된 비디오와 함께 추가 데이터를 전송할 수 있다. 소스 코더(530)는 이러한 데이터를 코딩된 비디오 시퀀스의 일부로서 포함할 수 있다. 추가 데이터는 시간/공간/SNR 향상 계층, 중복 화상 및 슬라이스와 같은 다른 형태의 중복 데이터, SEI 메시지, VUI 파라미터 세트 프래그먼트 등을 포함할 수 있다.

[0073] 비디오는 시간적 시퀀스에서 복수의 소스 화상(비디오 화상)으로 캡처될 수 있다. 인트라 화상 예측(종종 인트라 예측으로 축약됨)은 주어진 화상에서 공간적 상관 관계를 사용하고, 인터 화상 예측은 화상 간의 (시간적 또는 기타) 상관 관계를 사용한다. 일 예에서, 현재 화상이라고 불리는 인코딩/디코딩중인 특정 화상은 블록으로 파티셔닝된다(partitioned). 현재 화상에서의 블록이 이전에 코딩되고 비디오에서 여전히 버퍼링된 참조 화상의 참조 블록과 유사할 때, 현재 화상에서의 블록은 모션 벡터라고 하는 벡터로 코딩될 수 있다. 모션 벡터는 참조 화상에서 참조 블록을 가리키며, 여러 참조 화상이 사용 중인 경우에 참조 화상을 식별하는 3차원을 가질 수 있다.

[0074] 일부 실시 예에서, 이중 예측(bi-prediction) 기술이 인터 화상 예측에 사용될 수 있다. 이중 예측 기법에 따르면, 비디오에서 현재 화상에 대한 디코딩 순서에서 모두 선행되는 제1 참조 화상 및 제2 참조 화상과 같은 2개의 참조 화상(그러나 디스플레이 순서에서 각각 과거 및 미래일 수 있음)이 사용된다. 현재 화상의 블록은 제1 참조 화상에서 제1 참조 블록을 가리키는 제1 모션 벡터 및 제2 참조 화상에서 제2 참조 블록을 가리키는 제2 모션 벡터에 의해 코딩될 수 있다. 블록은 제1 참조 블록과 제2 참조 블록의 조합에 의해 예측될 수 있다.

[0075] 또한, 인터 화상 예측에 병합 모드 기술을 사용하여 코딩 효율을 높일 수 있다.

[0076] 본 개시의 일부 실시 예에 따르면, 인터 화상 예측 및 인트라 화상 예측과 같은 예측은 블록의 단위로 수행된다. 예를 들어, HEVC 표준에 따르면, 비디오 화상 시퀀스에서의 화상은 압축을 위해 코딩 트리 유닛(coding tree unit, CTU)으로 파티셔닝되고, 화상에서의 CTU는 64×64 픽셀, 32×32 픽셀 또는 16×16 픽셀과 같은 동일한 크기를 갖는다. 일반적으로 CTU는 3개의 코딩 트리 블록(coding tree block, CTB)을 포함하며, 이는 1개의 루마 CTB와 2개의 크로마 CTB이다. 각 CTU는 반복적으로 하나 또는 여러 코딩 유닛(coding unit, CU)으로 쿼드트리(quadtree) 분할될(split) 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. 일 예에서, 각 CU는 인터 예측 유형 또는 인트라 예측 유형과 같은 CU에 대한 예측 유형을 결정하기 위해 분석된다. CU는 시간적 및/또는 공간적 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 실시 예에서, 코딩(인코딩/디코딩)에서의 예측 작동은 예측 블록의 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 값의 행렬(예: 루마 값)을 포함한다.

[0077] 도 6은 본 개시의 다른 실시 예에 따른 비디오 인코더(603)의 다이어그램을 도시한다. 비디오 인코더(603)는 비디오 화상의 시퀀스에서 현재 비디오 화상 내의 샘플 값의 처리 블록(예: 예측 블록)을 수신하고, 처리 블록을 코딩된 비디오 시퀀스의 부분인 코딩된 화상으로 인코딩하도록 구성된다. 일 예에서, 비디오 인코더(603)는 도 3 예의 비디오 인코더(303) 대신에 사용된다.

[0078] HEVC 예에서, 비디오 인코더(603)는 8×8 샘플의 예측 블록 등과 같은 처리 블록에 대한 샘플 값의 행렬을 수신한다. 비디오 인코더(603)는 처리 블록이 예를 들어 레이트 왜곡 최적화를 사용하여 인트라 모드, 인터 모드 또는 이중 예측 모드를 사용하여 최상의 코딩되는지를 판정한다. 처리 블록이 인트라 모드로 코딩될 때, 비디오 인코더(603)는 인트라 예측 기술을 사용하여 처리 블록을 코딩된 화상으로 인코딩할 수 있으며; 처리 블록이 인터 모드 또는 이중 예측 모드로 코딩될 때, 비디오 인코더(603)는 각각 인터 예측 또는 이중 예측 기술을 사용하여, 처리 블록을 코딩된 화상으로 인코딩할 수 있다. 특정 비디오 코딩 기술에서, 병합 모드는 예측자 외부의

코딩된 모션 벡터 컴포넌트의 이점 없이 모션 벡터가 하나 이상의 모션 벡터 예측자로부터 유도되는 인터 화상 예측 서브 모드일 수 있다. 특정 다른 비디오 코딩 기술에서, 주제 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.

[0079] 도 6 예에서, 비디오 인코더(603)는 도 6에서 도시된 바와 같이 함께 결합된, 인터 인코더(630), 인트라 인코더(622), 잔차 계산기(623), 스위치(626), 잔차 인코더(624), 일반 컨트롤러(621) 및 엔트로피 인코더(625)를 포함한다.

[0080] 인터 인코더(630)는 현재 블록(예: 처리 블록)의 샘플을 수신하고, 블록을 참조 화상의 하나 이상의 참조 블록((예: 이전 화상 및 이후 화상에서의 블록)과 비교하며, 인터 예측 정보(예: 인터 인코딩 기술에 따른 중복 정보의 설명, 모션 벡터, 병합 모드 정보)를 생성하고, 임의의 적절한 기술을 사용하여 인터 예측 정보를 기반으로 인터 예측 결과(예: 예측된 블록)를 계산한다. 일부 예에서, 참조 화상은 인코딩된 비디오 정보에 기반하여 디코딩되는, 디코딩된 참조 화상이다.

[0081] 인트라 인코더(622)는 현재 블록(예: 처리 블록)의 샘플을 수신하고, 일부 경우에 블록을 동일한 화상에서 이미 코딩된 블록과 비교하며, 변환 후 양자화된 계수를 생성하고, 일부 경우에는 또한 인트라 예측 정보(예: 하나 이상의 인트라 인코딩 기술에 따른 인트라 예측 방향 정보)를 생성한다. 일 예에서, 인트라 인코더(622)는 또한 동일한 화상에서의 인트라 예측 정보 및 참조 블록에 기반하여 인트라 예측 결과(예를 들어, 예측 블록)를 계산한다.

[0082] 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, 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 예측 정보를 선택하고 인트라 예측 정보를 비트스트림에 포함시키며; 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 선택하고, 인터 예측 정보를 비트스트림에 포함시킨다.

[0083] 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 양자화된 변환 계수를 획득하기 위해 양자화 처리를 거친다. 다양한 실시 예에서, 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 적절하게 처리되어 디코딩된 화상을 생성하며, 디코딩된 화상은 메모리 회로(도시되지 않음)에 버퍼링될 수 있고 일부 예에서 참조 화상으로서 사용될 수 있다.

[0084] 엔트로피 인코더(625)는 인코딩된 블록을 포함하도록 비트스트림을 포맷하도록 구성된다. 엔트로피 인코더(625)는 HEVC 표준과 같은 적합한 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 서브 모드에서 블록을 코딩할 때, 잔차 정보가 없음을 유의한다.

[0085] 도 7은 본 개시의 다른 실시 예에 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 3 예의 비디오 디코더(310) 대신에 사용된다.

[0086] 도 7 예에서, 비디오 디코더(710)는 도 7에서 도시된 바와 같이 함께 결합된, 엔트로피 디코더(771), 인터 디코더(780), 잔차 디코더(773), 재구성 모듈(774) 및 인트라 디코더(772)를 포함한다.

[0087] 엔트로피 디코더(771)는 코딩된 화상으로부터, 코딩된 화상이 구성되는 신택스 엘리먼트를 나타내는 특정 심볼을 재구성하도록 구성될 수 있다. 이러한 심볼은 예를 들어 블록이 코딩되는 모드(예를 들어, 인트라 모드, 인

터 모드, 이중 예측 모드, 병합 서브 모드 또는 다른 서브 모드에서의 후자 두 개), 인트라 디코더(772) 또는 인터 디코더(780)에 의해 각각 예측에 사용되는 특정 샘플 또는 메타 데이터를 식별할 수 있는 예측 정보(예를 들어, 인트라 예측 정보 또는 인터 예측 정보), 예를 들어, 양자화된 변환 계수 등과 같은 형태의 잔차 정보를 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 때, 인트라 예측 정보는 인트라 디코더(772)에 제공된다. 잔차 정보는 역 양자화(inverse quantization) 대상이 될 수 있으며 잔차 디코더(773)에 제공된다.

[0088] 인터 디코더(780)는 인터 예측 정보를 수신하고 인터 예측 정보에 기반하여 인터 예측 결과를 생성하도록 구성된다.

[0089] 인트라 디코더(772)는 인트라 예측 정보를 수신하고 인트라 예측 정보에 기반하여 예측 결과를 생성하도록 구성된다.

[0090] 잔차 디코더(773)는 역 양자화를 수행하여 역양자화된(de-quantized) 변환 계수를 추출하고, 역양자화된 변환 계수를 처리하여 잔차를 주파수 도메인에서 공간 도메인으로 변환하도록 구성된다. 잔차 디코더(773)는 또한 양자화기 파라미터(Quantizer Parameter, QP)를 포함하기 위해 특정 제어 정보를 요구할 수 있으며, 그 정보는 엔트로피 디코더(771)에 의해 제공될 수 있다(데이터 경로는 낮은 볼륨 제어 정보일 수 있으므로 표시되지 않음).

[0091] 재구성 모듈(774)은 공간 도메인에서 잔차 디코더(773)에 의해 출력된 잔차 및 예측 결과(경우에 따라 인터 또는 인트라 예측 모듈에 의해 출력)를 조합하여(combine), 재구성된 화상의 일부일 수 있는 재구성된 블록을 형성하며, 이는 결국 재구성된 비디오의 일부일 수 있다. 디블로킹(deblocking) 작동 등과 같은 다른 적절한 작동이 시각적 품질을 향상시키기 위해 수행될 수 있다는 점에 유의한다.

[0092] 비디오 인코더(303, 503 및 603) 및 비디오 디코더(310, 410 및 710)는 임의의 적절한 기술을 사용하여 구현될 수 있음에 유의한다. 일 실시 예에서, 비디오 인코더(303, 503 및 603) 및 비디오 디코더(310, 410 및 710)는 하나 이상의 집적 회로를 사용하여 구현될 수 있다. 다른 실시 예에서, 비디오 인코더(303, 503 및 503) 및 비디오 디코더(310, 410 및 710)는 소프트웨어 명령을 실행하는 하나 이상의 프로세서를 사용하여 구현될 수 있다.

[0093] II. 디블로킹에서 경계 강도 유도(Boundary Strength Derivation)

[0094] HEVC에서, 디코딩 프로세스와 동일한 순서로 각 CU에 대해 디블로킹 필터링 프로세스가 수행된다. 수직 에지(edge)가 먼저 필터링되고(수평 필터링) 그 다음에 수평 에지가 필터링된다(수직 필터링). 8×8 블록 경계가 필터링되는 것으로 결정될 때, 루마 컴포넌트 및 크로마 컴포넌트 모두에 대해 8×8 블록 경계에 필터링이 적용될 수 있다. 복잡성을 감소시키기 위해 4×4 블록 경계는 처리되지 않을 수 있다. 경계 강도(boundary strength, Bs)는 경계에 사용될 수 있는 디블로킹 필터링 프로세스의 정도(degree) 또는 강도를 지시하는 데 사용될 수 있다. 일 실시 예에서, Bs에 대한 값 2는 강력한 필터링을 지시하고, 1은 약한 필터링을 지시하며, 0은 디블로킹 필터링이 없음을 지시한다.

[0095] 일 실시 예에서, Bs는 4×4 샘플 그리드(grid) 기반으로 계산되지만, 8×8 샘플 그리드로 다시 매핑될 수 있다. 일 예에서, 8×8 블록은 4개의 4×4 블록을 포함하므로, 8×8 블록의 경계는 2개의 인접한 4×4 블록의 두 변(side)을 포함한다. 4×4 그리드에서 선(line)을 형성하는 8개의 픽셀에 대응하는 Bs의 두 값 중 최대 값을 8×8 그리드의 경계에 대한 Bs로 선택할 수 있다.

[0096] 도 8은 본 개시의 실시 예에 따라 Bs 값을 결정하기 위한 예시적인 프로세스(800)의 흐름도를 도시한다. 도 8의 단계의 순서는 다른 실시 예에서 순서가 변경되거나 하나 이상의 단계가 생략될 수 있음을 유의한다.

[0097] 도 8에서, P 및 Q는 그들 사이에 경계가 있는 인접한 두 블록이다. 수직 에지의 경우, P는 경계의 왼쪽에 있는 블록을 나타내고, Q는 경계의 오른쪽에 있는 블록을 나타낼 수 있다. 수평 에지의 경우, P는 경계 위에 있는 블록을 나타내고, Q는 경계 아래에 있는 블록을 나타낼 수 있다.

[0098] 도 8에 도시된 바와 같이, Bs 값은 예측 모드(예: 인트라 코딩 모드), 0이 아닌(non-zero) 변환 계수(예: 0이 아닌 변환 계수의 존재), 참조 화상, 모션 벡터의 수(a number of motion vectors) 및 모션 벡터 차이에 기반하여 결정될 수 있다.

[0099] 단계(S810)에서, 프로세스(800)는 P 또는 Q가 인트라 예측 모드로 코딩되었는지를 판정한다. P 및 Q 중 적어도 하나가 인트라 예측 모드로 코딩되는 것으로 결정될 때, 프로세스(800)는 Bs에 대한 제1 값(예: 2)을 결정한다.

그렇지 않으면, 프로세스(800)는 단계(S820)로 진행한다.

[0100] 단계(S820)에서, 프로세스(800)는 P 또는 Q가 0이 아닌 변환 계수를 갖는지를 판정한다. P 및 Q 중 적어도 하나가 0이 아닌 변환 계수를 갖는 것으로 결정될 때, 프로세스(800)는 Bs에 대한 제2 값(예: 1)을 결정한다. 그렇지 않으면, 프로세스(800)는 단계(S830)로 진행한다.

[0101] 단계(S830)에서, 프로세스(800)는 P와 Q가 서로 다른 참조 화상을 가지고 있는지를 판정한다. P와 Q가 서로 다른 참조 화상을 갖는 것으로 결정될 때, 프로세스(800)는 Bs에 대한 제3 값(예: 1)을 결정한다. 그렇지 않으면, 프로세스(800)는 단계(S840)로 진행한다.

[0102] 단계(S840)에서, 프로세스(800)는 P와 Q가 서로 다른 수의 모션 벡터를 가지고 있는지를 판단한다. P와 Q가 서로 다른 수의 모션 벡터를 갖는 것으로 결정될 때, 프로세스(800)는 Bs에 대한 제4 값(예: 1)을 결정한다. 그렇지 않으면, 프로세스(800)는 단계(S850)로 진행한다.

[0103] 단계(S850)에서, 프로세스(800)는 P와 Q 사이의 모션 벡터 차이가 임계 값 T보다 크거나 같은지를 판정한다. P와 Q 사이의 모션 벡터 차이가 임계 값 T보다 크거나 같다고 결정될 때, 프로세스(800)는 Bs에 대한 제5 값(예: 1)을 결정한다. 그렇지 않으면, 프로세스(800)는 Bs에 대한 제6 값(예: 0)을 결정한다. 일 실시 예에서, 임계 값 T는 1 픽셀로 설정된다. 일 예에서, MV 정밀도(precision)는 1/4 픽셀이고 MV 차이 임계 값은 4로 설정될 수 있다. 다른 예에서, MV 정밀도는 1/16이고 MV 차이 값은 16으로 설정될 수 있다.

[0104] III. 블록 차동 펄스 코드 변조 모드(Block Differential Pulse-code Modulation Mode)

[0105] 블록 차동 펄스 코드 변조(Block Differential Pulse-code Modulation Mode, BDPCM)는 블록 레벨에서 차동 펄스 코드 변조(differential pulse-code modulation, DPCM) 접근 방식을 사용하는 인트라 코딩 도구이다. bdpcm_flag는 CU가 각 차원이 32보다 작거나 같은 루마 인트라 코딩된 CU일 때마다 CU 레벨에서 전송될 수 있다. 이 플래그는 일반 인트라 코딩(regular intra coding) 또는 DPCM이 사용되는지를 지시하며, 그리고 단일 CABAC(Context-based adaptive binary arithmetic coding) 콘텍스트를 사용하여 인코딩된다.

[0106] BDPCM은 LOCO-I의 MED(Median Edge Detector)(예: JPEG-LS에서 사용됨)를 사용할 수 있다. 특히, 픽셀 A를 왼쪽 이웃으로, 픽셀 B를 상단(top) 이웃으로, C를 왼쪽 상단(top-left) 이웃으로 갖는 현재 픽셀 X의 경우, 현재 픽셀 X의 예측 P(X)는 다음 수식:

[0107]
$$P(X) = \min(A, B) \text{ if } C \geq \max(A, B)$$

[0108]
$$\max(A, B) \text{ if } C \leq \min(A, B)$$

[0109]
$$A+B-C \text{ otherwise.}$$

[0110] 에 의해 결정된다.

[0111] 픽셀 예측자는 CU의 상단 행과 왼쪽 열에서 예측할 때 필터링되지 않은 참조 픽셀을 사용한다. 그런 다음 예측자는 나머지 CU에 대해 재구성된 픽셀을 사용한다. 픽셀은 CU 내부에서 래스터 스캔 순서(raster-scan order)로 처리된다. 예측 에러(prediction error)는 리스케일링(rescaling) 후, 변환 스킵 양자화와 동일한 방식으로 공간 도메인에서 양자화될 수 있다. 각 픽셀은 역양자화된 예측 에러를 예측에 추가하는 것에 의해 재구성될 수 있다. 따라서, 재구성된 픽셀을 사용하여 래스터 스캔 순서에서의 다음 픽셀을 예측할 수 있다. 양자화된 예측 에러의 진폭(amplitude)과 부호(sign)는 별도로 인코딩된다. cbf_bdpcm_flag도 코딩된다. cbf_bdpcm_flag가 0이면, 블록의 모든 진폭은 0으로 디코딩된다. cbf_bdpcm_flag가 1이면, 블록의 모든 진폭이 래스터 스캔 순서로 개별적으로 인코딩된다. 복잡성을 낮게 유지하기 위해, 진폭을 최대 31(포함)로 제한할 수 있다. 진폭은 제1 bin(bin)에 대해 3개의 콘텍스트, 제12 bin까지 각 추가 bin에 대해 하나의 콘텍스트, 그리고 나머지 모든 bin에 대해 하나의 콘텍스트를 가지는 단항 이진화(unary binarization)를 사용하여, 인코딩될 수 있다. 부호는 각 제로 잔차에 대해 우회(bypass) 모델로 인코딩될 수 있다.

[0112] 일반 인트라 모드 예측(regular intra mode prediction)의 일관성을 유지하기 위해, 인트라 모드 예측의 MPM(most probable mode) 리스트에서의 제1 모드는 BDPCM CU(전송되지 않음)와 연관되며 후속 블록을 위한 MPM 생성에 사용될 수 있다.

[0113] 디블로킹 필터는 BDPCM 코딩된 블록 중 어느 것도 통상 블로킹 아티팩트(artifact)에 책임이 있는 변환을 수행하지 않기 때문에 2개의 BDPCM 코딩된 블록 사이의 보더(border)/경계(boundary)에서 비활성화될 수 있다. 또한, BDPCM은 여기에 설명된 단계를 제외하고는 다른 단계를 사용할 수 없다. 특히, BDPCM은 전술한 바와 같이

잔차 코딩에서 어떠한 변환도 수행하지 않을 수 있다.

- [0114] BDPCM의 처리량 향상과 다른 SCC(Screen Content Coding) 도구와의 상호 작용을 조사하기 위해 BDPCM과 관련된 여러 테스트가 수행되었다.
- [0115] 도 9a는 일 실시 예에 따른 BDPCM 코딩된 블록의 예를 도시한다. 도 9a에 도시된 예는 테스트 1과 관련된다. 도 9a에 도시된 바와 같이, 처리량을 높이기 위해, 더 작은 블록(예: 4×4 , 4×8 및 8×4 크기)은 블록을 효과적으로 2개의 반으로 분할하는 대각선(예: 계단 모양의 파티션(stair-case shaped partition))을 사용하여, 2개의 독립적으로 디코딩 가능한 영역으로 나눌 수 있다.
- [0116] 일 실시 예에서, 전반부(first half)의 한 영역의 픽셀은 예측을 계산하기 위해 후반부(second half)의 다른 영역의 픽셀을 사용하는 것이 허용되지 않을 수 있다. 한 영역의 픽셀이 예측을 계산하기 위해 다른 영역의 픽셀을 사용해야 하면, 대신 참조 픽셀이 사용된다. 예를 들어, 다른 영역의 픽셀은 가장 가까운 참조 픽셀로 대체될 수 있다. 예를 들어, 왼쪽 이웃은 동일한 행의 왼쪽 참조 픽셀로 대체될 수 있고, 상단 이웃은 동일한 열의 왼쪽 참조 픽셀로 대체될 수 있으며, 왼쪽 상단 이웃은 가장 가까운 참조 픽셀로 대체될 수 있다. 따라서, 두 영역을 병렬로 처리할 수 있다.
- [0117] 도 9a는 또한 상이한 크기를 갖는 각 블록의 예시적인 처리량을 제공한다. 예를 들어, 2개의 독립적으로 디코딩 가능한 영역이 있는 4×4 블록의 경우, 처리량은 사이클당 4 픽셀일 수 있다. 2개의 독립적으로 디코딩 가능한 영역이 있는 4×8 또는 8×4 블록의 경우, 처리량은 사이클당 5.33 픽셀일 수 있다. 독립적으로 디코딩 가능한 영역이 없는 8×8 블록의 경우, 처리량은 사이클당 4.26 픽셀일 수 있다. 독립적으로 디코딩 가능한 영역이 없는 16×16 블록의 경우, 처리량은 사이클당 8.25 픽셀일 수 있다.
- [0118] 도 9b는 일 실시 예에 따른 BDPCM 코딩된 블록의 예를 도시한다. 도 9b에 도시된 예는 테스트 2와 관련된다. 도 9b에서, 블록은 JPEG-LS 예측자를 대체하는 수직 또는 수평 예측자를 사용하여 나뉘질 수 있다. 수직 또는 수평 예측자는 블록 레벨에서 선택되고 시그널링될 수 있다. 독립적으로 디코딩 가능한 영역의 모양은 예측자의 기하학적 구조(geometry)를 반영한다. 현재 픽셀의 예측을 위해 왼쪽 또는 상단 픽셀을 사용하는 수평 또는 수직 예측자의 모양으로 인해, 블록을 처리하는 가장 처리량이 효율적인 방법은 하나의 열 또는 행의 모든 픽셀을 병렬로 처리하고, 이러한 열 또는 행을 순차적으로 처리하는 것이다. 예를 들어, 처리량을 증가시키기 위해, 이 블록에서 선택한 예측자가 수직일 때 너비 4의 블록은 수평 경계가 있는 2개의 반으로 나뉘지고, 이 블록에서 선택한 예측자가 수평일 때 높이 4의 블록은 수직 경계가 있는 2개의 반으로 나뉘진다. 2개의 독립적으로 디코딩 가능한 영역이 있는 4×4 블록, 8×4 또는 4×8 블록의 경우, 처리량은 사이클당 8 픽셀일 수 있다. 독립적으로 디코딩 가능한 영역이 없는 4×8 블록, 8×4 블록 또는 8×8 블록의 경우, 처리량은 사이클당 8 픽셀일 수 있다. 독립적으로 디코딩 가능한 영역이 없는 16×16 블록의 경우, 처리량은 사이클당 16 픽셀일 수 있다.
- [0119] 테스트 3에서, 본 개시의 실시 예에 따르면, BDPCM 잔차 진폭은 28로 제한되고, 진폭은 처음 12개 bin에 대해 잘린 단항 이진화(truncated unary binarization)로 인코딩된 다음, 나머지에 대해 2차 Exp-Golomb 동등의 확률 bin(equal probability bin)으로 인코딩된다(예: encodeRemAbsEP() 함수 사용).
- [0120] **IV. 변환 계수 코딩(Transform Coefficient Coding)**
- [0121] 엔트로피 코딩은 비디오 신호가 일련의 선택스 엘리먼트로 감소된 후 비디오 코딩의 마지막 스테이지(stage)(또는 비디오 디코딩의 제1 스테이지)에서 수행될 수 있다. 엔트로피 코딩은 데이터를 나타내는 데 사용되는 비트 수가 데이터의 확률에 대수적으로 비례하도록 데이터를 압축하기 위해 통계 속성을 사용하는 무손실 압축 방식일 수 있다. 예를 들어, 선택스 엘리먼트 세트에 대해 엔트로피 코딩을 수행하는 것에 의해, 선택스 엘리먼트(bin이라고 함)를 나타내는 비트를 비트스트림에서 더 적은 비트(코딩된 비트라고 함)로 변환할 수 있다. CABAC는 엔트로피 코딩의 한 형태이다. CABAC에서, 확률 추정치(probability estimate)를 제공하는 콘텍스트 모델은 각 bin과 연관된 콘텍스트에 기반하여 bin의 시퀀스에서의 각 bin에 대해 결정될 수 있다. 이어서, 이진 산술 코딩(binary arithmetic coding) 프로세스가 확률 추정치를 사용하여 수행되어 bin의 시퀀스를 비트스트림에서의 코딩된 비트로 인코딩할 수 있다. 또한, 콘텍스트 모델은 코딩된 bin을 기반으로 신규 확률 추정치로 업데이트된다.
- [0122] 도 10a는 일 실시 예에 따른 예시적인 CABAC 기반 엔트로피 인코더(1000A)를 도시한다. 예를 들어, 엔트로피 인코더(1000A)는 도 5 예의 엔트로피 코더(545), 또는 도 6 예의 엔트로피 인코더(625)에서 구현될 수 있다. 엔트로피 인코더(1000A)는 콘텍스트 모델러(context modeler)(1010) 및 이진 산술 인코더(1020)를 포함할 수 있다. 일 예에서, 다양한 유형의 선택스 엘리먼트가 엔트로피 인코더(1000A)에 대한 입력으로 제공된다. 예를 들어,

이진 값 선택스 엘리먼트(binary valued syntax element)의 빈은 컨텍스트 모델러(1010)에 직접 입력될 수 있는 반면, 이진 값이 아닌 선택스 엘리먼트(non-binary valued syntax element)는 빈 스트링(string)의 빈이 콘텍스트 모델러(1010)에 입력되기 전에 빈 스트링으로 이진화될 수 있다.

[0123] 일 예에서, 컨텍스트 모델러(1010)는 선택스 엘리먼트의 빈을 수신하고, 컨텍스트 모델링 프로세스를 수행하여 수신된 각 빈에 대한 컨텍스트 모델을 선택한다. 예를 들어, 변환 블록에서 변환 계수의 이진 선택스 엘리먼트의 빈이 수신된다. 변환 블록은 현재 블록이 예측을 위해 BDPCM으로 코딩될 때 변환 스킵된 블록일 수 있다. 예를 들어, 선택스 엘리먼트의 유형, 변환 컴포넌트의 컬러 컴포넌트 유형, 변환 계수의 위치 및 이전에 처리된 이웃 변환 계수 등에 기반하여, 이 빈에 대한 컨텍스트 모델이 결정될 수 있다. 컨텍스트 모델은 이 빈에 대한 확률 추정치를 제공할 수 있다.

[0124] 일 예에서, 하나 이상의 선택스 엘리먼트 유형에 대해 컨텍스트 모델 세트를 구성할 수 있다. 이러한 컨텍스트 모델은 도 10a에 도시된 바와 같이 메모리(1001)에 저장되는 컨텍스트 모델 리스트(1002)에 배열될 수 있다. 컨텍스트 모델 리스트(1002)의 각 항목(entry)은 컨텍스트 모델을 나타낼 수 있다. 리스트의 각 컨텍스트 모델은 컨텍스트 모델 인덱스 또는 컨텍스트 인덱스라고 하는 인덱스가 할당될 수 있다. 또한, 각 컨텍스트 모델은 확률 추정치 또는 확률 추정치를 지시하는 파라미터를 포함할 수 있다. 확률 추정치는 빈이 0 또는 1 일 가능성을 지시할 수 있다. 예를 들어, 컨텍스트 모델링 프로세스에서, 컨텍스트 모델러(1010)는 빈에 대한 컨텍스트 인덱스를 계산할 수 있으며, 컨텍스트 인덱스에 따라 컨텍스트 모델 리스트(1002)로부터 컨텍스트 모델을 선택하여 빈에 할당할 수 있다.

[0125] 더욱이, 컨텍스트 모델 리스트의 확률 추정치는 엔트로피 인코더(1000A)의 작동을 시작할 때 초기화될 수 있다. 컨텍스트 모델 리스트(1002)의 컨텍스트 모델이 빈에 할당되고 빈을 인코딩하는 데 사용된 후, 컨텍스트 모델은 업데이트된 확률 추정치를 사용하여 빈의 값에 따라 후속적으로 업데이트될 수 있다.

[0126] 일 예에서, 이진 산술 인코더(1020)는 빈과 빈에 할당된 컨텍스트 모델(예: 확률 추정치)을 수신하고, 이에 따라 이진 산술 코딩 프로세스를 수행한다. 결과적으로, 코딩된 비트가 생성되어 비트스트림으로 전송된다.

[0127] 도 10b는 일 실시 예에 따른 예시적인 CABAC 기반 엔트로피 디코더(1000B)를 도시한다. 예를 들어, 엔트로피 디코더(1000B)는 도 4 예의 파서(420) 또는 도 7 예의 엔트로피 디코더(771)에서 구현될 수 있다. 엔트로피 디코더(1000B)는 이진 산술 디코더(1030) 및 컨텍스트 모델러(1040)를 포함할 수 있다. 이진 산술 디코더(1030)는 비트스트림으로부터 코딩된 비트를 수신하고, 이진 산술 디코딩 프로세스를 수행하여 코딩된 비트로부터 빈을 복원한다. 컨텍스트 모델러(1040)는 컨텍스트 모델러(1010)와 유사하게 작동할 수 있다. 예를 들어, 컨텍스트 모델러(1040)는 메모리(1003)에 저장된 컨텍스트 모델 리스트(1004)로부터 컨텍스트 모델을 선택하고, 선택된 컨텍스트 모델을 이진 산술 디코더(1030)에 제공할 수 있다. 컨텍스트 모델러(1040)는 이진 산술 디코더(1030)로부터 복원된 빈에 기반하여 컨텍스트 모델을 결정할 수 있다. 예를 들어, 복원된 빈을 기반으로 컨텍스트 모델러(1040)는 다음 디코딩될 빈의 선택스 엘리먼트 유형 및 이전에 디코딩된 선택스 엘리먼트의 값을 알 수 있다. 이 정보는 다음 디코딩될 빈에 대한 컨텍스트 모델을 결정하는 데 사용된다.

[0128] V. 컨텍스트 코딩된 빈 수에 대한 제한(Limitations on a number of Context Coded Bins)

[0129] 전술한 바와 같이, 각 픽셀과 연관된 잔차는 그 부호를 나타내는 빈과 그 절대 값을 나타내는 일련의 빈으로 이진화될 수 있다. 절대 값은 주어진 숫자(예: 31 또는 28)보다 작거나 같도록 양자화될 수 있다. 하나 이상의 컨텍스트 모델 또는 동등의 확률(equal probability, EP) 모델(즉, 우회 모델)이 각 빈과 연관될 수 있다. 각 잔차 심볼의 경우, 최대 13개의 빈이 하나 이상의 컨텍스트 모델과 연관될 수 있다. 많은 수의 컨텍스트 코딩된 빈(예: 적어도 컨텍스트 모델과 연관된 빈)은 선택스 코딩 동안 엔트로피 인코더 또는 디코더의 처리량을 감소시킬 수 있으며, 특정 코덱 구현(예: 일부 하드웨어 코덱)에 대한 부담이 될 수 있다. 따라서, 컨텍스트 코딩된 빈의 최대 수를 설정하면 예를 들어 코딩 속도를 높일 수 있을 뿐만 아니라 필요한 메모리 크기와 컨텍스트 모델 유지 비용을 줄일 수 있다.

[0130] 따라서, 본 개시의 측면들은 변환 계수 엔트로피 코딩에 사용되는 컨텍스트 코딩된 빈의 수를 제한하는 방법을 포함한다. 예를 들어, BDPCM 모드에서, 양자화된 잔차의 이진 표현에서의 빈의 수가 컨텍스트 모델과 연관될 수 있다. 일부 실시 예에서 컨텍스트 모델과 연관된 빈의 수에는 하나 이상의 제한이 있다.

[0131] $W \times H$ 블록은 가상으로 복수의 서브 영역(sub-region)으로 나뉘질 수 있다. 복수의 서브 영역의 서브 세트는 서로 병렬로 BDPCM으로 코딩될 수 있다. 일 예에서, 복수의 서브 영역들 각각은 서로 병렬로 BDPCM으로 코딩될 수 있다. $W \times H$ 블록은 변환 블록 또는 변환 스킵된 블록일 수 있다. 각 서브 영역에 대한 컨텍스트 코딩된 빈(예:

MaxCpbs)의 최대 수는 $B \times A$ 로 설정될 수 있으며, 여기서 A는 서브 영역의 양자화된 잔차의 수이고, B는 양수이다. B는 예를 들어 인코더에서 디코더로 비트스트림으로 시그널링되거나 시그널링되지 않을 수 있다. 서브 영역의 빈 수가 제한(예: $B \times A$)을 초과하지 않을 때, 서브 영역의 각 빈이 컨텍스트 모델로 코딩될 수 있다. 서브 영역의 빈 수가 제한을 초과할 때, 컨텍스트 코딩되지 않은 나머지 빈은 다른 모델(예: EP 모델)로 코딩될 수 있다.

[0132] 일 실시 예에서, B는 루마 컴포넌트 및 크로마 컴포넌트에 대해 동일하다. 예를 들어, $B = 2$ 이다. 일 실시 예에서, B는 루마 컴포넌트 및 크로마 컴포넌트에 대해 상이하다. 예를 들어, 루마 컴포넌트의 경우 $B = 2$ 이고, 크로마 컴포넌트의 경우 $B = 1$ 이다.

[0133] 일 실시 예에서, B는 0.5 또는 1.2와 같은 분수(fractional number)일 수 있다. 일부 예에서, B는 A에 의존한다. A는 위에 정의된 바와 같이 서브 영역의 양자화된 잔차의 수이다. 예를 들어, A가 임계 값보다 작을 때, B는 특정 값(예: 1)과 같을(equal) 수 있다. 그렇지 않으면, B는 다른 값(예: 2)과 같을 수 있다. 임계 값은 예를 들어 4, 8, 16, 32, 64, 128 또는 256일 수 있다.

[0134] 일 실시 예에서, B는 2개의 임계 값에 기반하여 결정될 수 있다. 예를 들어, A가 제1 임계 값보다 작을 때 B는 제1 값(예: 1)과 같을 수 있다. A가 제2 임계 값보다 크거나 같을 때 B는 제2 값(예: 2)과 같을 수 있다. 예를 들어, 제1 임계 값 및 제2 임계 값은 4, 8, 16, 32, 64, 128 및/또는 256 값 중 하나일 수 있다.

[0135] 일 실시 예에서, B는 전술한 것과 유사한 방식으로 2개 이상의(예를 들어, 3 개) 임계 값에 기반하여 결정될 수 있다. 예를 들어, A가 제1 임계 값보다 작을 때 B는 제1 값(예: 0.5)과 같을 수 있다. A가 제1 임계 값보다 크거나 같고 제2 임계 값보다 작을 때, B는 제2 값(예를 들어, 1)과 같을 수 있다. A가 제3 임계 값보다 크거나 같을 때 B는 제3 값(예: 2)과 같을 수 있다. 예를 들어, 제1 임계 값, 제2 임계 값 및 제3 임계 값은 4, 8, 16, 32, 64, 128 및/또는 256 값 중 하나일 수 있다.

[0136] 일 실시 예에서, $W \times H$ 블록은 서브 영역으로 나뉘지 않고 하나의 영역만을 포함한다. 영역에 대한 컨텍스트 코딩된 빈의 최대 수는 또한 $B \times A$ 로 설정될 수 있다. 이 경우, A는 $W \times H$ 와 같을 수 있으며, B는 양수이고, 컨텍스트 코딩된 빈의 최대 수(예: MaxCpbs)는 $B \times W \times H$ 와 같을 수 있다. 블록에서의 빈 수가 제한(예: $B \times W \times H$)을 초과하지 않을 때, 블록에서의 각 빈은 컨텍스트 모델로 코딩될 수 있다. 블록에서의 빈 수가 제한을 초과할 때, 코딩되지 않는 나머지 빈은 다른 모델(예: EP 모델)로 코딩될 수 있다.

[0137] 일 실시 예에서, $W \times H$ 블록은 H개의 서브 영역으로 나뉘지고 각 서브 영역은 W개 잔차의 한 행(one row of W residues)을 포함한다. 이 경우, A는 W와 같을 수 있으며, 각 행에 대한 컨텍스트 코딩된 빈(예: MaxCpbs)의 최대 수는 $B \times W$ 이다. 행에서의 빈 수가 제한(예: $B \times W$)을 초과하지 않을 때, 행에서의 각 빈은 컨텍스트 모델로 코딩될 수 있다. 행에서의 빈 수가 제한을 초과할 때, 컨텍스트 모델과 연결되지 않은 나머지 빈이 다른 모델(예: EP 모델)로 코딩될 수 있다.

[0138] 일 실시 예에서, $W \times H$ 블록은 W개의 서브 영역으로 나뉘지고 각 서브 영역은 H개 잔차의 한 열을 포함한다. 이 경우, A는 H와 같을 수 있으며 각 열에 대한 컨텍스트 코딩된 빈(예: MaxCpbs)의 최대 수는 $B \times H$ 와 같을 수 있다. 서브 영역에서의 빈 수가 제한(예: $B \times H$)을 초과하지 않을 때, 열에서의 각 빈은 컨텍스트 모델로 코딩될 수 있다. 열에서의 빈 수가 제한을 초과할 때, 컨텍스트 모델과 연결되지 않은 나머지 빈이 다른 모델(예: EP 모델)로 코딩될 수 있다.

[0139] 일 실시 예에서, 임의의 파티션이 도 9a에 설명된 계단형 파티션 또는 도 9b에 설명된 수평/수직 분할과 같은 서브 영역을 생성하는 데 사용될 수 있다. 또한, 위에서 설명된 다른 실시 예에서 2개 이상의 파티션이 사용될 수 있다.

[0140] 도 11은 컨텍스트 코딩된 빈의 최대 수에 기반한 계수 인코딩의 예시적인 방법을 도시한다. 컨텍스트 코딩된 빈의 최대 수는 예를 들어, 위에서 설명된 하나 이상의 실시 예에 기반하여 설정될 수 있다.

[0141] 전술한 바와 같이, 선택스 엘리먼트(예: 빈)를 나타내는 비트는 선택스 엘리먼트에 대해 엔트로피 코딩을 수행하는 것에 의해 비트스트림에서 더 적은 비트로 변환될 수 있다. 도 11에서, 양자화된 잔차(1110)는 이진 코딩을 사용하여 이진 스트링(1120)으로 변환될 수 있다. 예를 들어, 변환된 이진 스트링(1120)은 11010010001111100011101010010101010010101010111000111000101101011010101110 일 수 있다. 이진 스트링(1120)은 4개의 세그먼트를 포함할 수 있으며, 2개의 선택스 엘리먼트를 나타낼 수 있다. 제1 선택스 엘리먼트의 빈은 110100100011111000111010100101010이고, 제2 선택스 엘리먼트의 빈은 10010101010111000111000101101011010101110이다. 도 6의 엔트로피 인코더(545) 또는 도 6의 엔트로피 인코더

(625)와 같은 CABAC 기반 엔트로피 인코더는, 콘텍스트 코딩된 bin의 최대 수에 기반하여 이진 스트림에 대한 콘텍스트 모델링을 수행할 수 있다. 예를 들어, 블록 W의 너비와 블록 H의 높이가 모두 4와 같고 블록이 하나의 영역만 포함할 때, A는 $4 \times 4 = 16$ 과 같고, B는 2로 설정될 수 있다. 따라서, 블록에서 콘텍스트 코딩된 bin의 최대 수는 $B \times A = 32$ 이다. 따라서, 도 11에 도시된 바와 같이, 블록(굵은 글꼴 및 밑줄 글꼴로 표시됨)에 대한 콘텍스트 코딩된 bin의 수가 (이진 스트림의 처음 3개의 세그먼트에 표시된 바와 같이) 32에 도달하지 않을 때, 엔트로피 인코더가 BDPCM 알고리즘에 따라, 콘텍스트 모델링 사용 여부를 자유롭게 선택하여 bin을 코딩한다. 콘텍스트 코딩된 bin(예: MaxCbs)의 최대 수에 도달할 때(제4 세그먼트에 표시된 바와 같이) 나머지 bin은 콘텍스트 모델링으로 코딩될 수 없다. 대신에, bin은 EP 모델로 코딩될 수 있다. 일부 예들에서, 엔트로피 인코더는 카운터를 포함할 수 있고, 카운터는 콘텍스트 코딩된 bin의 수를 카운팅한다. 카운터 출력은 초기에 0으로 설정될 수 있으며, 0부터 콘텍스트 코딩된 bin의 최대 수까지 카운팅할 수 있다. 카운터가 콘텍스트 코딩된 bin의 최대 수에 도달할 때, 나머지 bin은 콘텍스트 모델링으로 코딩되지 않을 수 있다.

[0142] 일 실시 예에서, 블록 W의 너비와 블록 H의 높이가 모두 8과 같고 블록이 동일한 크기의 4개의 서브 영역으로 분할될 때, 각 서브 영역의 크기는 4×4 이다. 따라서, A는 16과 같다. 이 경우, B는 2일 수 있고, 콘텍스트 코딩된 bin의 최대 수는 32이다. 따라서, 도 11에 설명된 예는 본 실시 예에서도 적용될 수 있다.

[0143] 도 12는 콘텍스트 코딩된 bin의 최대 수에 기반한 계수 디코딩의 예시적인 방법을 도시한다. 일 실시 예에 따라 CABAC 기반 엔트로피 디코더가 제공될 수 있다. 엔트로피 디코더는 도 4 예의 파서(420) 또는 도 7의 엔트로피 디코더(771)에서 구현될 수 있다. 전술한 바와 같이, 엔트로피 디코더의 이진 산술 디코더는 비트스트림으로부터 코딩된 비트(1210)를 수신하고, 이진 산술 디코딩 프로세스를 수행하여 코딩된 bin으로부터 bin(1220)을 복원한다. 엔트로피 인코더와 유사하게, 엔트로피 디코더는 블록 또는 분할된 서브 영역의 크기에 기반하여 콘텍스트 코딩된 bin의 최대 수를 결정할 수도 있다. 예를 들어, 블록 W의 너비와 블록 H의 높이가 모두 4와 같고 블록이 하나의 영역만 포함할 때, A는 16과 같고 B는 2일 수 있다. 따라서, 콘텍스트 코딩된 bin의 최대 수는 $B \times A = 32$ 이다. 따라서, 도 12에 도시된 바와 같이, 이 블록(굵게 및 밑줄 글꼴로 표시됨)에 대한 콘텍스트 코딩된 bin의 수가 32(이진 스트림의 처음 3개의 세그먼트에 표시된 바와 같이)에 도달하지 않을 때, 엔트로피 디코더가 BDPCM 알고리즘에 따라, 콘텍스트 모델링 사용 여부를 자유롭게 선택하여 bin을 코딩한다. 콘텍스트 코딩된 bin의 최대 수에 도달할 때(제4 세그먼트에 표시된 바와 같이), bin은 콘텍스트 모델링으로 디코딩될 수 없다. 대신, bin은 EP 모델로 디코딩된다. 디코딩된 이진 스트림(1220)은 비이진화(debinarization) 프로세스를 사용하여 양자화된 잔차(1230)로 변환될 수 있다.

[0144] VI. BDPCM 코딩된 블록에 디블로킹 필터의 적용(Application of a Deblocking Filter on BDPCM Coded Blocks)

[0145] 전술한 바와 같이, 두 BDPCM 코딩된 블록 사이의 경계에서 디블로킹 필터는 비활성화될 수 있는데, 이는 예를 들어 BDPCM 코딩된 블록 중 어느 것도 블로킹 아티팩트를 유발하는 변환을 수행하지 않기 때문이다. 블록 아티팩트를 피하거나 최소화하기 위해, 블록 중 적어도 하나가 BDPCM으로 코딩되어 지각적 아티팩트(perceptual artifact) 피하거나 최소화하는 두 블록 사이에 디블로킹 필터가 적용될 수 있다.

[0146] 일 실시 예에서, 디블로킹 필터는 BDPCM 코딩된 블록과 비-BDPCM(non-BDPCM) 코딩된 블록 사이에서 항상 활성화될 수 있다. 일부 예에서, 경계 강도(Bs)는 BDPCM 코딩된 블록과 비-BDPCM 코딩된 블록 사이에서 고정된 값으로 설정된다. 고정 값은 0, 1 또는 2일 수 있다. 예를 들어, 값 0은 디블로킹 필터가 비활성화되었음을 지시하고, 1은 약한 필터링이 적용됨을 지시하며, 2는 강력한 필터링이 적용됨을 지시한다.

[0147] 일 실시 예에서, 디블로킹 필터는 2개의 인접한 BDPCM 코딩된 블록 사이에서 항상 활성화될 수 있다. Bs는 두 BDPCM 코딩된 블록의 예측 모드에 따라 서로 다른 값으로 설정될 수 있다. 일부 예에서, 하나의 블록이 수평 BDPCM으로 코딩되고 다른 블록이 수직 BDPCM으로 코딩될 때, Bs는 2로 설정된다. 다른 예에서, 두 블록이 수평 예측을 사용하여 코딩될 때 Bs는 1로 설정된다.

[0148] 일부 실시 예에서, 디블로킹 필터는 2개의 인접한 BDPCM 코딩된 블록 사이에서 조건부로 활성화될 수 있다. 일부 예에서, 디블로킹 필터는 2개의 인접한 BDPCM 코딩된 블록의 QP 차이가 임계 값보다 클 때만 활성화된다. 예를 들어, 임계 값은 0, 1, 2 또는 3일 수 있다.

[0149] 일부 예에서, 디블로킹 필터는 2개의 BDPCM 코딩된 블록이 서로 다른 예측 모드를 가질 때만 활성화된다. 예를 들어, 한 블록이 수평 BDPCM으로 코딩되고 다른 블록이 수직 BDPCM으로 코딩될 때 디블로킹 필터가 활성화된다.

[0150] 일부 예에서, BDPCM은 2개의 인접한 BDPCM 모드 간의 경계 강도 유도에서 인트라 모드로 간주되지 않는다. 경계 강도 유도에서의 참조 및 MV 검사는 인접한 두 BDPCM 블록에 대해 스킵된다. 예를 들어, 도 8에 설명된 경계 강

도 유도 프로세스가 적용되고, 2개의 인접한 BDPCM 코딩된 블록 중 적어도 하나가 0이 아닌 변환 계수를 가질 때, Bs는 1로 설정되고 디블로킹 필터가 활성화된다. 인접한 2개의 BDPCM 코딩된 블록 중 어느 것도 0이 아닌 변환 계수를 가지지 않을 때, Bs는 0으로 설정되고 디블로킹 필터는 비활성화된다.

[0151] VII. 예시적인 디코딩 프로세스

[0152] 도 13은 본 개시의 일부 실시 예들에 따른 엔트로피 디코딩 프로세스(1300)를 개략적으로 설명하는 흐름도를 도시한다. 프로세스(1300)는 여기에 개시된 컨텍스트 코딩된 bin의 최대 수에 기반하여 여러 유형의 변환 계수 선택스 엘리먼트의 엔트로피 디코딩에 사용될 수 있다. 다양한 실시 예에서, 프로세스(1300)는 단말 디바이스(210, 220, 230 및 240)의 처리 회로, 비디오 디코더(310)의 기능을 수행하는 처리 회로, 비디오 디코더(410)의 기능을 수행하는 처리 회로 등과 같은 처리 회로에 의해 실행될 수 있다. 일부 실시 예에서, 프로세스(1300)는 소프트웨어 명령으로 구현되고, 따라서 처리 회로가 소프트웨어 명령을 실행할 때, 처리 회로는 프로세스(1300)를 수행한다. 프로세스는 (S1301)에서 시작하여 (S1310)으로 진행된다.

[0153] (S1310)에서, 코딩된 비트를 포함하는 비트스트림이 수신될 수 있다. 코딩된 bin은 코딩된 화상에서 변환 스킵된 블록의 잔차에 대응하는 다양한 유형의 선택스 엘리먼트의 bin으로부터 코딩될 수 있다. 예를 들어, 다양한 유형의 선택스 엘리먼트는 중요성(significance) 선택스 엘리먼트, 패리티(parity) 선택스 엘리먼트, 1개 이상의 선택스 엘리먼트(greater than 1 syntax elements) 및/또는 2개 이상의 선택스 엘리먼트(greater than 2 syntax elements)를 포함할 수 있다. 중요성 선택스 엘리먼트(예: sig_coeff_flag)는 현재 변환 계수(absLevel)의 절대 값이 0보다 크다는 것을 지시할 수 있다. 패리티 선택스 엘리먼트(예: par_level_flag)는 absLevel의 패리티를 지시할 수 있다. 1개 이상의 선택스 엘리먼트(예: rem_abs_gt1_flag)는 absLevel-1이 0보다 크다는 것을 지시할 수 있다. 2개 이상의 선택스 엘리먼트(예: rem_abs_gt2_flag)는 absLevel-4가 0보다 크다는 것을 지시할 수 있다. 변환 스킵된 블록은 변환 블록에 대해 변환이 수행되지 않았음을 지시할 수 있다. 예를 들어, 현재 블록이 BDPCM으로 코딩될 때, 변환 블록에서 변환이 수행되지 않는다.

[0154] (S1320)에서, 영역의 선택스 엘리먼트의 다수의 bin에 대한 컨텍스트 모델을 결정하기 위해 컨텍스트 모델링이 수행될 수 있다. 컨텍스트 코딩된 선택스 엘리먼트의 bin 수는 영역에 대해 설정된 컨텍스트 코딩된 bin의 최대 수를 초과할 수 없다. 선택스 엘리먼트의 bin의 수는 변환 스킵된 블록의 영역에 대한 선택스 엘리먼트의 총 bin 수보다 작거나 같다. 컨텍스트 코딩된 bin의 최대 수는 전술한 실시 예 중 하나 이상에 따라 결정될 수 있다. 예를 들어, 변환 스킵된 블록의 영역에 대한 컨텍스트 코딩된 bin(예: MaxCcb)의 최대 수는 $B \times A$ 로 설정되며, 여기서 B는 양수이고 A는 영역의 양자화된 잔차의 수이다. B는 정수 또는 분수와 같은 미리 결정될 수 있을 수 있다. 다르게는, B는 A에 의존할 수 있다. B는 인코더에서 디코더로 시그널링되어, 디코더가 $B \times A$ 의 컨텍스트 코딩된 bin의 최대 수를 결정할 수도 있다.

[0155] (S1330)에서, 결정된 컨텍스트 모델에 기반하여 선택스 엘리먼트의 다수의 bin의 코딩된 비트를 디코딩할 수 있다. 변환 스킵된 블록의 영역에 대한 선택스 엘리먼트의 나머지 총 수의 bin의 코딩된 비트는 EP 모델(즉, 우회 모델)에 기반하여 디코딩될 수 있다. 복원된 bin을 기반으로 변환 계수의 변환 계수 레벨이 재구성될 수 있다. 프로세스(1300)는 (S1399)로 진행하여 종료한다.

[0156] 도 14는 본 개시의 일부 실시 예에 따른 디블로킹 필터링 프로세스(1400)를 개략적으로 설명하는 흐름도를 도시한다. 프로세스(1400)는 코딩된 비디오 비트스트림의 현재 블록 및 이웃 블록 중 적어도 하나가 BDPCM(block Differential Pulse-code Modulation)으로 코딩될 때 사용될 수 있다. 다양한 실시 예에서, 프로세스(1400)는 단말 디바이스(210, 220, 230 및 240)의 처리 회로, 비디오 디코더(310)의 기능을 수행하는 처리 회로, 비디오 디코더(410)의 기능을 수행하는 처리 회로 등과 같은 처리 회로에 의해 실행될 수 있다. 일부 실시 예에서, 프로세스(1400)는 소프트웨어 명령으로 구현되고, 따라서 처리 회로가 소프트웨어 명령을 실행할 때, 처리 회로는 프로세스(1400)를 수행한다. 프로세스는 (S1401)에서 시작하여 (S1410)으로 진행된다.

[0157] (S1410)에서, 코딩된 비트를 포함하는 비트스트림을 수신할 수 있다.

[0158] (S1420)에서, 인코더로부터 지시자(indicator)를 수신할 수 있다. 지시자는 코딩된 비디오 비트스트림의 현재 블록 및 이웃 블록 중 적어도 하나가 BDPCM(block Differential Pulse-code Modulation)으로 코딩되는지를 지시할 수 있다. 현재 블록은 이웃 블록에 인접할 수 있다. 현재 블록은 현재 CU에 있으며, 이웃 블록은 동일한 CU에 있거나 현재 CU에 인접한 다른 CU에 있을 수 있다. 전술한 바와 같이, CU가 32보다 작거나 같은 차원을 갖는 루마 인트라 CU일 때마다 bdpcm_flag는 CU 레벨에서 전송될 수 있다. 따라서, 디코더에 의해 수신된 bdpcm_flag가 CU가 BDPCM으로 코딩되었음을 지시할 때, 현재 블록은 BDPCM으로 코딩된다. 따라서, CU에서의 현

재 블록 및 수신된 코딩된 비디오 비트스트림의 이웃 블록 중 적어도 하나는 BDPCM으로 코딩된다. 유사하게, 디코더에 의해 수신된 `bdpcm_flag`가 다른 CU가 BDPCM으로 코딩되고 이웃 블록이 다른 CU에 있음을 지시할 때, 이웃 블록은 BDPCM으로 코딩된다. 따라서, CU 내의 현재 블록 및 수신된 코딩된 비디오 비트스트림의 다른 CU 내의 이웃 블록 중 적어도 하나는 BDPCM으로 코딩된다.

- [0159] (S1430)에서, 프로세스(1400)는 현재 블록 및 이웃 블록 중 적어도 하나가 BDPCM으로 코딩된 것으로 지시되는지를 판정한다. 현재 블록 및 이웃 블록 중 적어도 하나가 BDPCM으로 코딩된 것으로 지시될 때, 프로세스(1400)는 단계(S1440)로 진행한다.
- [0160] (S1440)에서, 현재 블록의 현재 서브 블록과 이웃 블록의 이웃 서브 블록 사이의 경계에 적용할 경계 강도(B_s)를 결정할 수 있다.
- [0161] (S1450)에서, 결정된 경계 강도에 따라 디블로킹 필터를 사용하여 현재 블록의 현재 서브 블록과 이웃 블록의 이웃 서브 블록 사이의 경계에 대해 디블로킹을 수행할 수 있다. 디블로킹 필터는 전술한 실시 예 중 하나 이상에 따라 적용될 수 있다.
- [0162] 예를 들어, 현재 블록 및 이웃 블록 중 적어도 하나가 BDPCM으로 코딩된 것으로 지시될 때, 현재 블록의 현재 서브 블록과 이웃 블록에서의 이웃 서브 블록 사이에 디블로킹 필터가 항상 활성화 또는 비활성화될 수 있다. B_s 는 현재 블록의 현재 서브 블록과 이웃 블록의 이웃 서브 블록 사이의 경계에 적용될 것으로 결정될 수 있다. 현재 블록의 현재 서브 블록과 이웃 블록의 이웃 서브 블록은 서로 인접할 수 있다. B_s 의 결정이 도 8에 설명된 방법을 사용할 때, 현재 블록의 현재 서브 블록과 이웃 블록의 이웃 서브 블록 중 적어도 하나가 인트라 코딩된 경우에, B_s 는 2의 고정 값으로 결정된다. 일 실시 예에서, B_s 의 결정은 도 8에 설명된 방법을 사용하지 않을 수 있다. 대신, B_s 는 0, 1 또는 2와 같은 유연한 값으로 결정될 수 있다.
- [0163] 일 실시 예에서, 이웃 블록의 이웃 서브 블록도 BDPCM으로 코딩될 때, 현재 서브 블록과 이웃 서브 블록 사이에 디블로킹 필터가 항상 활성화될 수 있다.
- [0164] 일 실시 예에서, 이웃 블록의 이웃 서브 블록도 BDPCM으로 코딩될 때, 디블로킹 필터는 현재 서브 블록과 이웃 서브 블록 사이에서 조건부로 활성화될 수 있다. 조건은 2개의 서브 블록 간의 QP 차이, 현재 블록과 이웃 블록의 예측 모드, 그리고 도 8에서 설명한 방법에 따라 B 를 결정할 때 BDPCM이 인트라 모드로 간주되는지 여부를 포함할 수 있다.
- [0165] 프로세스(1400)는 (S1499)로 진행하여 종료한다.

[0167] *VIII. 컴퓨터 시스템

- [0168] 전술한 기술은 컴퓨터가 판독 가능한 명령을 사용하여 컴퓨터 소프트웨어로 구현될 수 있고, 하나 이상의 컴퓨터가 판독 가능한 미디어에 물리적으로 저장될 수 있다. 예를 들어, 도 15는 개시된 주제의 특정 실시 예를 구현하기에 적합한 컴퓨터 시스템(1500)을 도시한다.
- [0169] 컴퓨터 소프트웨어는 어셈블리, 컴파일레이션(compilation), 링킹(linking) 또는 유사한 메커니즘에 종속되어, 하나 이상의 컴퓨터 중앙 처리 유닛(central processing unit, CPU), 그래픽 처리 유닛(Graphics Processing Unit, GPU) 등에 의해, 직접 또는 해석(interpretation), 마이크로 코드 실행(micro-code execution) 등을 통해 실행될 수 있는 명령을 포함하는 코드를 생성하는, 임의의 적절한 기계 코드 또는 컴퓨터 언어를 사용하여 코딩될 수 있다.
- [0170] 명령은 예를 들어 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트 폰, 게임 디바이스, 사물 인터넷 디바이스 등을 포함하는 다양한 유형의 컴퓨터 또는 그의 컴포넌트에서 실행될 수 있다.
- [0171] 컴퓨터 시스템(1500)에 대한 도 15에 도시된 컴포넌트는 본질적으로 예시적이며, 본 개시의 실시 예를 구현하는 컴퓨터 소프트웨어의 사용 또는 기능의 범위에 대한 어떠한 제한도 제안하려는 것이 아니다. 컴포넌트의 구성이 컴퓨터 시스템(1500)의 예시적인 실시 예에 예시된 컴포넌트의 임의의 하나 또는 조합과 관련된 임의의 종속성 또는 요건을 갖는 것으로 해석되어서는 안된다.
- [0172] 컴퓨터 시스템(1500)은 특정 휴먼 인터페이스 입력 디바이스를 포함할 수 있다. 이러한 휴먼 인터페이스 입력 디바이스는 촉각 입력(예: 키 입력(keystroke), 스와이프(swipe), 데이터 글러브(glove) 모션), 오디오 입력(예: 음성(voice), 박수), 시각적 입력(예: 제스처), 후각 입력(도시되지 않음)을 통해 한 명 이상의 인간

사용자의 입력에 응답할 수 있다. 휴먼 인터페이스 디바이스는 오디오(예: 음성(speech), 음악, 주변 소리), 이미지(예: 스캔한 이미지, 정지 이미지 카메라로부터 획득된 사진 이미지), 비디오(예: 2차원 비디오, 입체 비디오를 포함한 3차원 비디오)와 같이 사람의 의식적 입력과 직접 관련이 없는 특정 미디어를 캡처하는데도 사용할 수 있다.

[0173] 입력 휴먼 인터페이스 디바이스는 키보드(1501), 마우스(1502), 트랙패드(trackpad)(1503), 터치 스크린(1510), 데이터 글러브(도시되지 않음), 조이스틱(1505), 마이크(1506), 스캐너(1507), 카메라(1508) 중 하나 이상을 포함할 수 있다.

[0174] 컴퓨터 시스템(1500)은 또한 특정 휴먼 인터페이스 출력 디바이스를 포함할 수 있다. 이러한 휴먼 인터페이스 출력 디바이스는 예를 들어 촉각 출력, 소리, 빛 및 냄새/맛을 통해 한 명 이상의 인간 사용자의 감각을 자극할 수 있다. 이러한 휴먼 인터페이스 출력 디바이스는 촉각 출력 디바이스(예를 들어, 터치 스크린(1510), 데이터 글러브(도시되지 않음) 또는 조이스틱(1505)에 의한 촉각 피드백을 포함할 수 있지만, 입력 디바이스로서 기능하지 않는 촉각 피드백 디바이스도 있을 수 있음), 오디오 출력 디바이스(예: 스피커(1509), 헤드폰(도시되지 않음)), 시각적 출력 디바이스(예: CRT 화면, LCD 화면, 플라즈마 화면, OLED 화면을 포함하는 화면(1510), 각각은 터치 스크린 입력 기능이 있거나 없으며, 각각은 촉각 피드백 기능이 있거나 없음 - 일부는 입체 출력, 가상 현실 안경(도시되지 않음), 홀로그램 디스플레이 및 스모크 탱크(smoke tank)(도시되지 않음)와 같은 수단을 통해 2차원 시각 출력 또는 3차원 출력을 출력할 수 있음), 및 프린터(도시되지 않음)를 포함할 수 있다.

[0175] 컴퓨터 시스템(1500)은 또한 사람이 액세스할 수 있는 저장 디바이스 그리고, CD/DVD를 가지는 CD/DVD ROM/RW(1520) 또는 유사한 미디어(media)(1521)를 포함하는 광학 미디어, 썸 드라이브(thumb-drive)(1522), 제거 가능한 하드 드라이브 또는 솔리드 스테이트 드라이브(1523), 테이프 및 플로피 디스크(도시되지 않음)와 같은 레거시 자기 미디어, 보안 동글(security dongle)(도시되지 않음)과 같은 특수 ROM/ASIC/PLD 기반 디바이스 등과 같은 관련 미디어를 포함할 수 있다.

[0176] 당업자는 또한 현재 개시된 주제와 관련하여 사용되는 용어 "컴퓨터가 판독 가능한 미디어"가 전송 미디어, 반송파 또는 다른 일시적인 신호를 포함하지 않는다는 것을 이해해야 한다.

[0177] 컴퓨터 시스템(1500)은 또한 하나 이상의 통신 네트워크에 대한 인터페이스를 포함할 수 있다. 네트워크는 예를 들어 무선, 유선, 광일 수 있다. 네트워크는 또한 로컬, 광역, 대도시, 차량 및 산업, 실시간, 지연 허용(delay-tolerant) 등일 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 네트워크와 같은 근거리 통신망, CANBus를 포함하는 차량 및 산업용 등을 포함한다. 특정 네트워크는 일반적으로 특정 범용 데이터 포트 또는 주변 장치 버스(1549)(예: 컴퓨터 시스템의 USB 포트(1500))에 연결된 외부 네트워크 인터페이스 어댑터를 필요로 하며; 다른 것들은 공통적으로, 아래에 설명된 바와 같이 부착(attachment)에 의해 컴퓨터 시스템(1500)의 코어에 통합된다(예를 들어, PC 컴퓨터 시스템에 대한 이더넷 인터페이스 또는 스마트폰 컴퓨터 시스템에 대한 셀룰러 네트워크 인터페이스). 이러한 네트워크를 사용하여 컴퓨터 시스템(1500)은 다른 엔티티와 통신할 수 있다. 이러한 통신은 단방향, 수신 전용(receive only)(예를 들어, 방송 TV), 단방향 전송 전용(uni-directional send-only)(예를 들어, CANbus에서 특정 CANbus 디바이스로) 또는 양방향(예를 들어, 로컬 또는 광역 디지털 네트워크를 사용하는 다른 컴퓨터 시스템)일 수 있다. 특정 프로토콜 및 프로토콜 스택은 위에서 설명한 바와 같이 각 네트워크 및 네트워크 인터페이스에서 사용될 수 있다.

[0178] 앞서 언급한 휴먼 인터페이스 디바이스, 사람이 액세스할 수 있는 저장 디바이스 및 네트워크 인터페이스는 컴퓨터 시스템(1500)의 코어(1540)에 부착될 수 있다.

[0179] 코어(1540)는 하나 이상의 중앙 처리 유닛(Central Processing Unit, CPU)(1541), 그래픽 처리 유닛(Graphics Processing Unit, GPU)(1542), FPGA(Field Programmable Gate Areas)(1543) 형태의 특수 프로그램 가능 처리 유닛, 특정 태스크(task)에 대한 가속기(accelerator)(1544) 등을 포함할 수 있다. 읽기 전용 메모리(Read-only memory, ROM)(1545), 랜덤 액세스 메모리(1546), 내부의 비 사용자(non-user) 액세스 하드 드라이브, SSD 등과 같은 내부 대용량 스토리지(1547)와 함께 이러한 디바이스는 시스템 버스(1548)를 통해 연결될 수 있다. 일부 컴퓨터 시스템에서, 시스템 버스(1548)는 추가 CPU, GPU 등에 의한 확장을 가능하게 하는 하나 이상의 물리적 플러그의 형태로 액세스 가능할 수 있다. 주변 디바이스는 직접 또는 주변 버스(1549)를 통해 코어의 시스템 서브(1548)에 부착될 수 있다. 주변 버스의 아키텍처는 PCI, USB 등을 포함한다.

[0180] CPU(1541), GPU(1542), FPGA(1543) 및 가속기(1544)는 조합하여 앞서 언급한 컴퓨터 코드를 구성할 수 있는 특

정 명령을 실행할 수 있다. 이 컴퓨터 코드는 ROM(1545) 또는 RAM(1546)에 저장될 수 있다. 임시 데이터는 RAM(1546)에 저장될 수도 있지만 영구 데이터는 내부 대용량 스토리지(mass storage)(1547)에 저장될 수 있다. 하나 이상의 CPU(1541), GPU(1542), 대용량 스토리지(1547), ROM(1545), RAM(1546) 등에 밀접하게 연관될 수 있는 캐시 메모리를 사용하여 모든 메모리 디바이스에 대한 빠른 저장 및 검색을 활성화할 수 있다.

[0181] 컴퓨터가 판독 가능한 미디어는 다양한 컴퓨터 구현 작동을 수행하기 위한 컴퓨터 코드를 가질 수 있다. 미디어 및 컴퓨터 코드는 본 개시의 목적을 위해 특별히 설계되고 구성된 것일 수 있거나, 컴퓨터 소프트웨어 분야의 숙련자에게 잘 알려져 있고 이용 가능한 종류일 수 있다.

[0182] 제한이 아닌 예로서, 아키텍처(1500), 특히 코어(1540)를 갖는 컴퓨터 시스템은 하나 이상의 실재하는 컴퓨터가 판독 가능한 미디어에 구현된 소프트웨어를 실행하는 프로세서(CPU, GPU, FPGA, 가속기 등 포함)의 결과로서 기능을 제공할 수 있다. 이러한 컴퓨터가 판독 가능한 미디어는 위에서 소개한 사용자 액세스 가능 대용량 스토리지와 관련된 미디어일 수 있으며, 코어 내부 대용량 스토리지(1547) 또는 ROM(1545)과 같은 비 일시적 특성인 코어(1540)의 특정 스토리지일 수 있다. 본 개시의 다양한 실시 예를 구현하는 소프트웨어는 이러한 디바이스에 저장되고 코어(1540)에 의해 실행될 수 있다. 컴퓨터가 판독 가능한 매체는 특정 요구에 따라 하나 이상의 메모리 디바이스 또는 칩을 포함할 수 있다. 소프트웨어는 코어(1540) 및 특히 그 안의 프로세서(CPU, GPU, FPGA 등 포함)가, RAM(1546)에 저장된 데이터 구조를 정의하는 것을 포함하고 소프트웨어에서 정의한 프로세스에 따라 이러한 데이터 구조를 수정한, 여기에 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하게 할 수 있다. 추가로 또는 대안으로, 컴퓨터 시스템은 여기에 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하기 위해 소프트웨어 대신 또는 소프트웨어와 함께 작동할 수 있는 회로(예: 가속기(1544))에 고정되거나 다른 방식으로 구현된 로직의 결과로 기능을 제공할 수 있다. 소프트웨어에 대한 참조는 로직을 포함할 수 있으며, 그 반대의 경우도 마찬가지이다. 컴퓨터가 판독 가능한 미디어에 대한 참조는 실행을 위한 소프트웨어를 저장하는 회로(예: 집적 회로(IC)), 실행을 위한 로직을 구현하는 회로 또는 적절한 경우 둘 다를 포함할 수 있다. 본 개시는 하드웨어 및 소프트웨어의 임의의 적절한 조합을 포함한다.

[0183] 부록 A: 약어

[0184] JEM: joint exploration model

[0185] VVC: versatile video coding

[0186] BMS: benchmark set

[0187] MV: Motion Vector

[0188] HEVC: High Efficiency Video Coding

[0189] SEI: Supplementary Enhancement Information

[0190] VUI: Video Usability Information

[0191] GOPs: Groups of Pictures

[0192] TUs: Transform Units

[0193] PUs: Prediction Units

[0194] CTUs: Coding Tree Units

[0195] CTBs: Coding Tree Blocks

[0196] PBs: Prediction Blocks

[0197] HRD: Hypothetical Reference Decoder

[0198] SNR: Signal Noise Ratio

[0199] CPUs: Central Processing Units

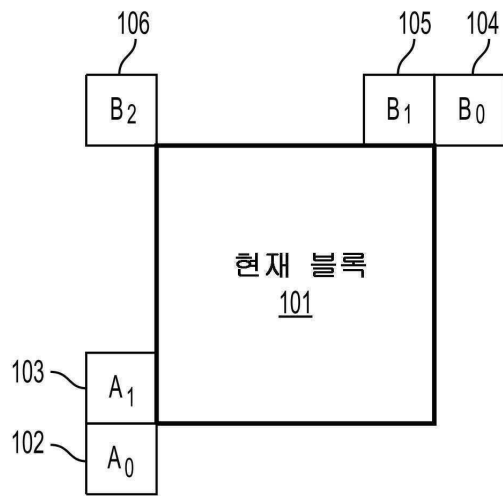
[0200] GPUs: Graphics Processing Units

[0201] CRT: Cathode Ray Tube

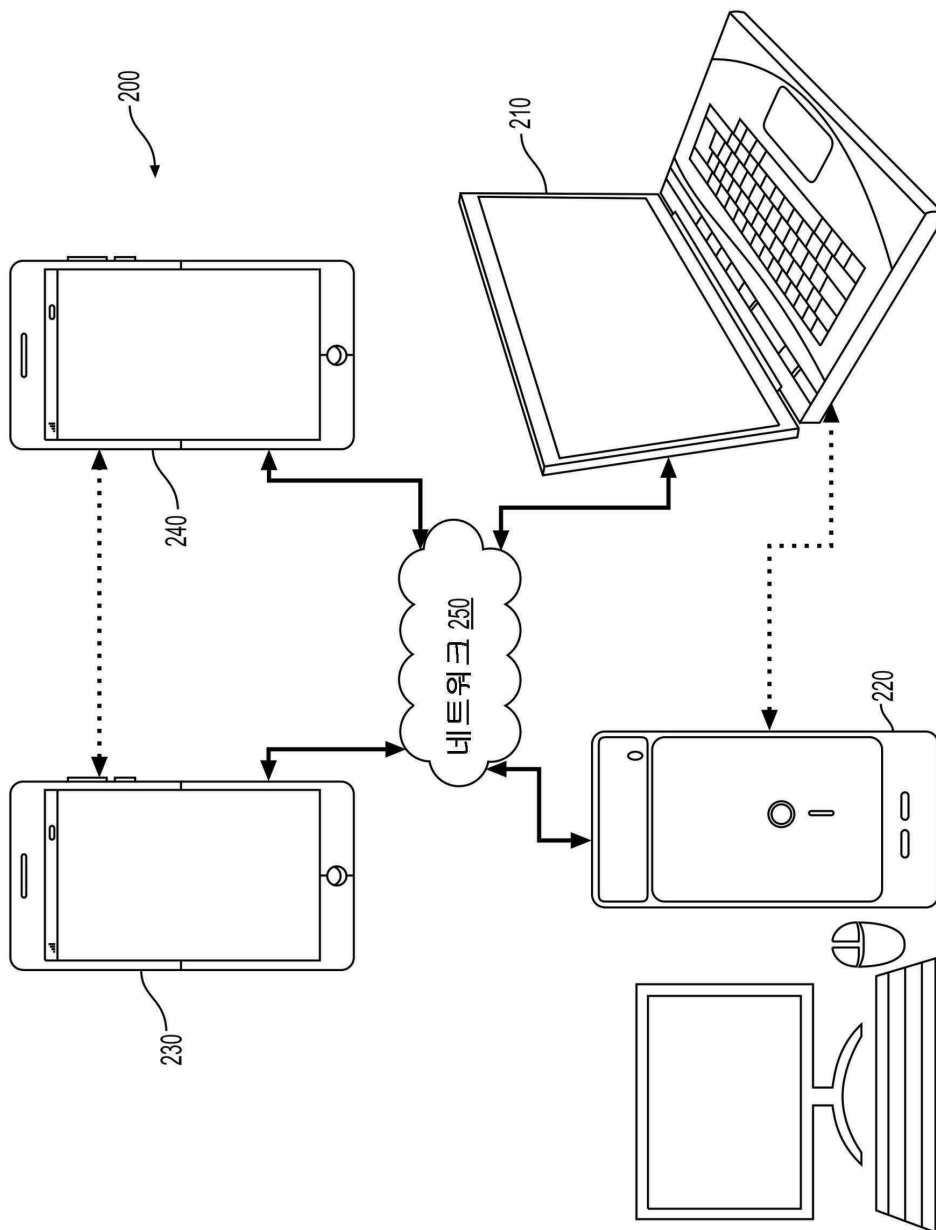
[0202]	LCD: Liquid-Crystal Display
[0203]	OLED: Organic Light-Emitting Diode
[0204]	CD: Compact Disc
[0205]	DVD: Digital Video Disc
[0206]	ROM: Read-Only Memory
[0207]	RAM: Random Access Memory
[0208]	ASIC: Application-Specific Integrated Circuit
[0209]	PLD: Programmable Logic Device
[0210]	LAN: Local Area Network
[0211]	GSM: Global System for Mobile communications
[0212]	LTE: Long-Term Evolution
[0213]	CANBus: Controller Area Network Bus
[0214]	USB: Universal Serial Bus
[0215]	PCI: Peripheral Component Interconnect
[0216]	FPGA: Field Programmable Gate Areas
[0217]	SSD: solid-state drive
[0218]	IC: Integrated Circuit
[0219]	CU: Coding Unit
[0220]	DPCM: Differential Pulse-code Modulation
[0221]	BDPCM: Block Differential Pulse-code Modulation
[0222]	SCC: Screen Content Coding
[0223]	Bs: Boundary Strength
[0224]	본 개시가 여러 예시적인 실시 예를 설명했지만, 본 개시의 범위 내에 속하는 변경, 순열 및 다양한 대체 등 가 물이 있다.

도면

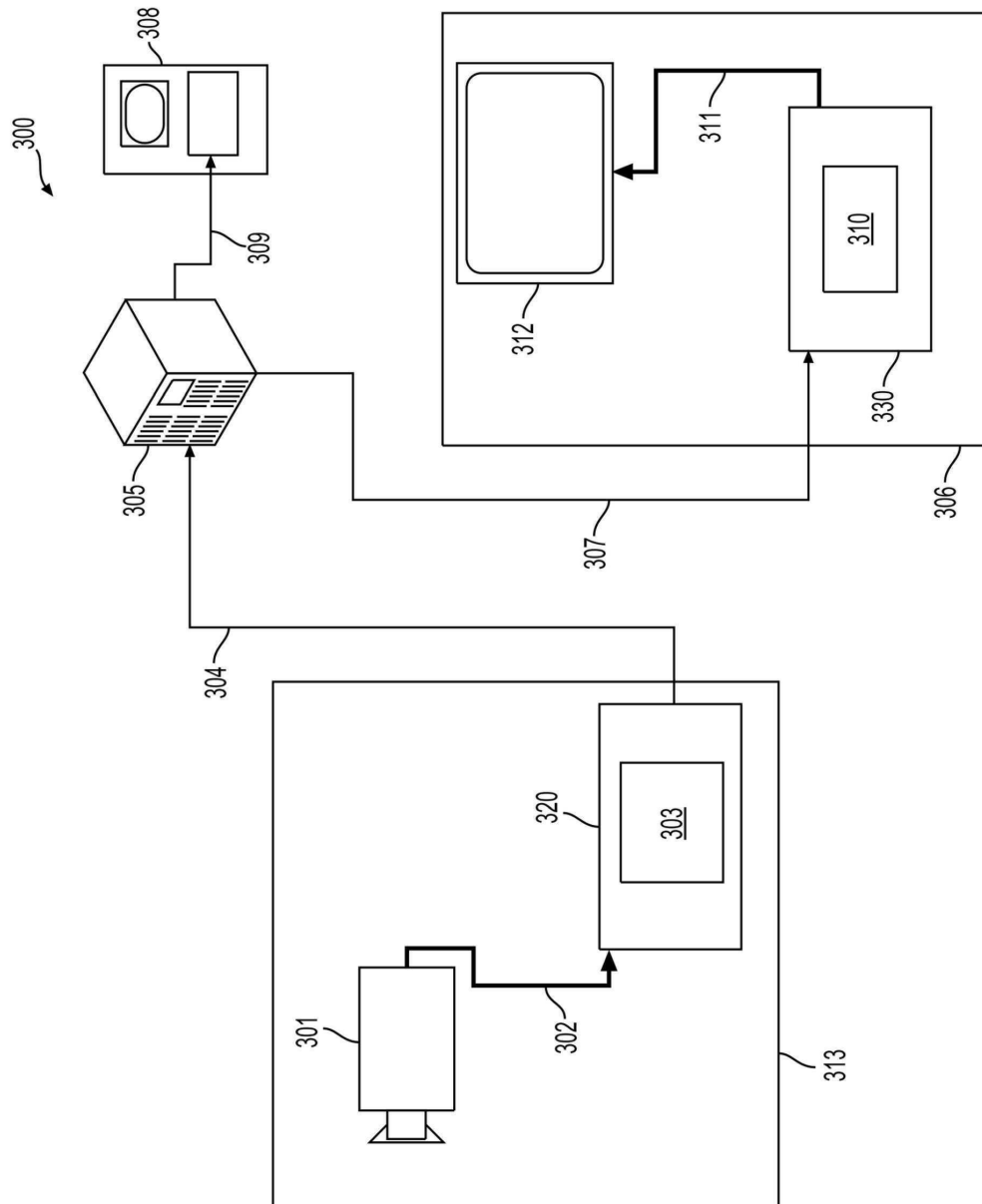
도면1



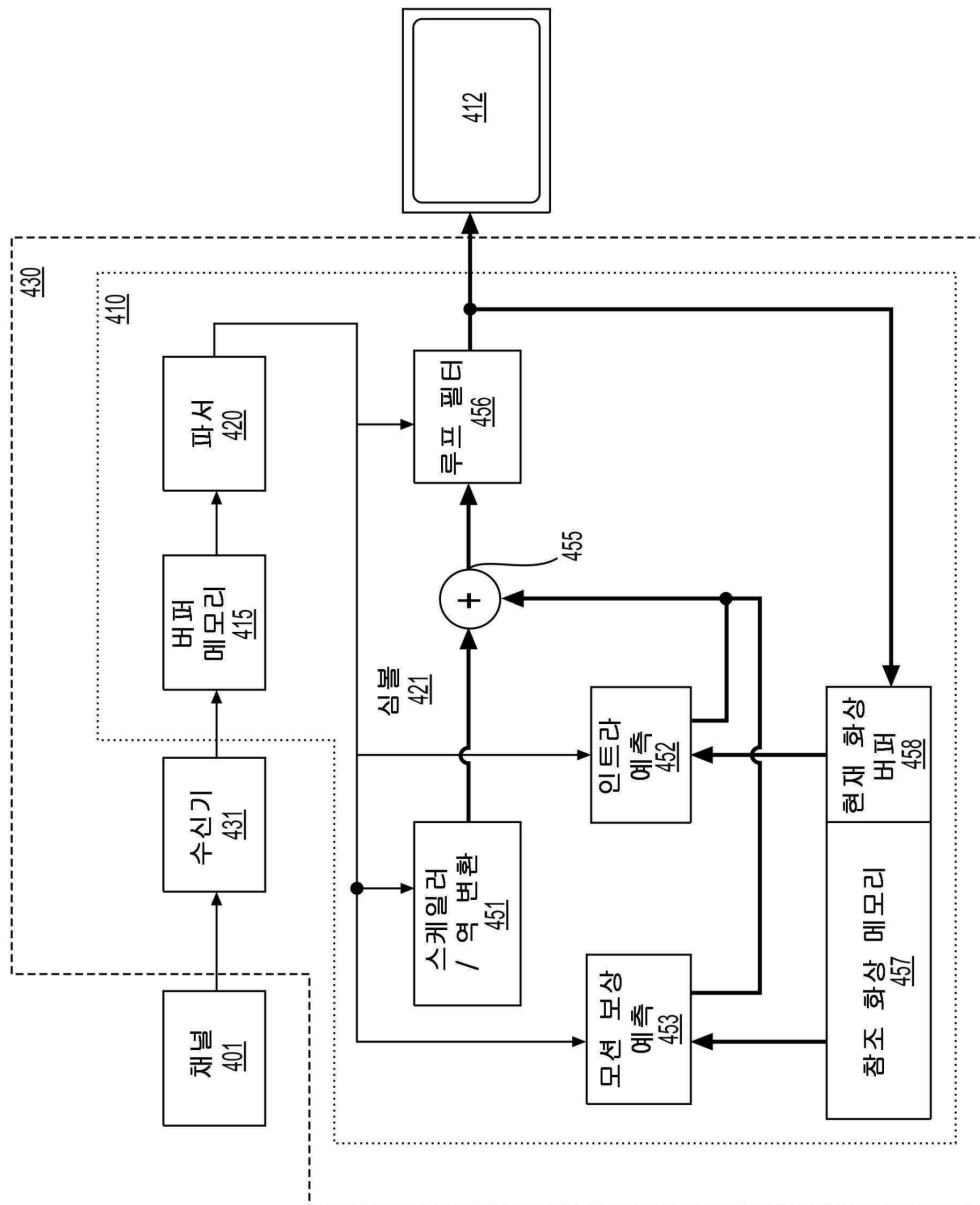
도면2



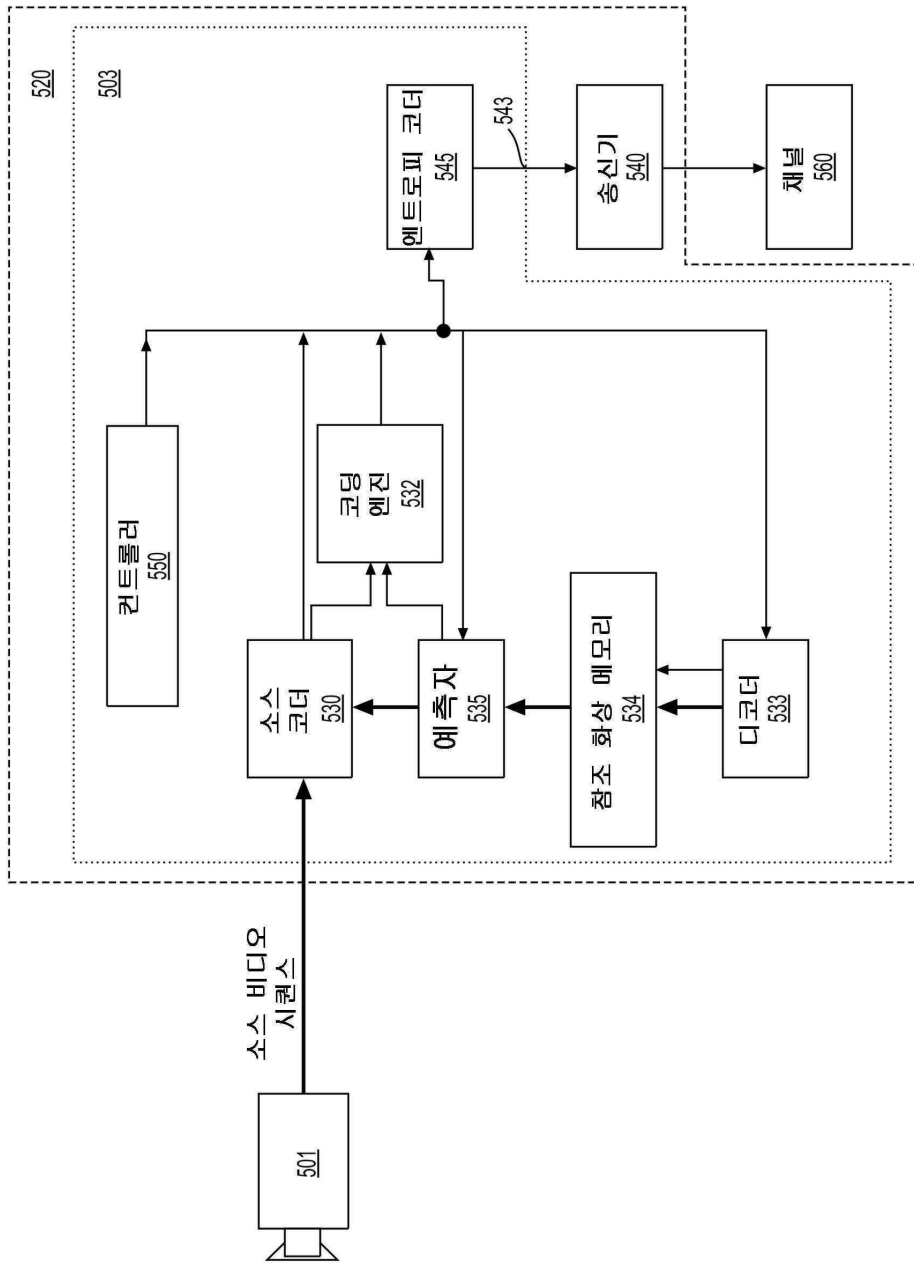
도면3



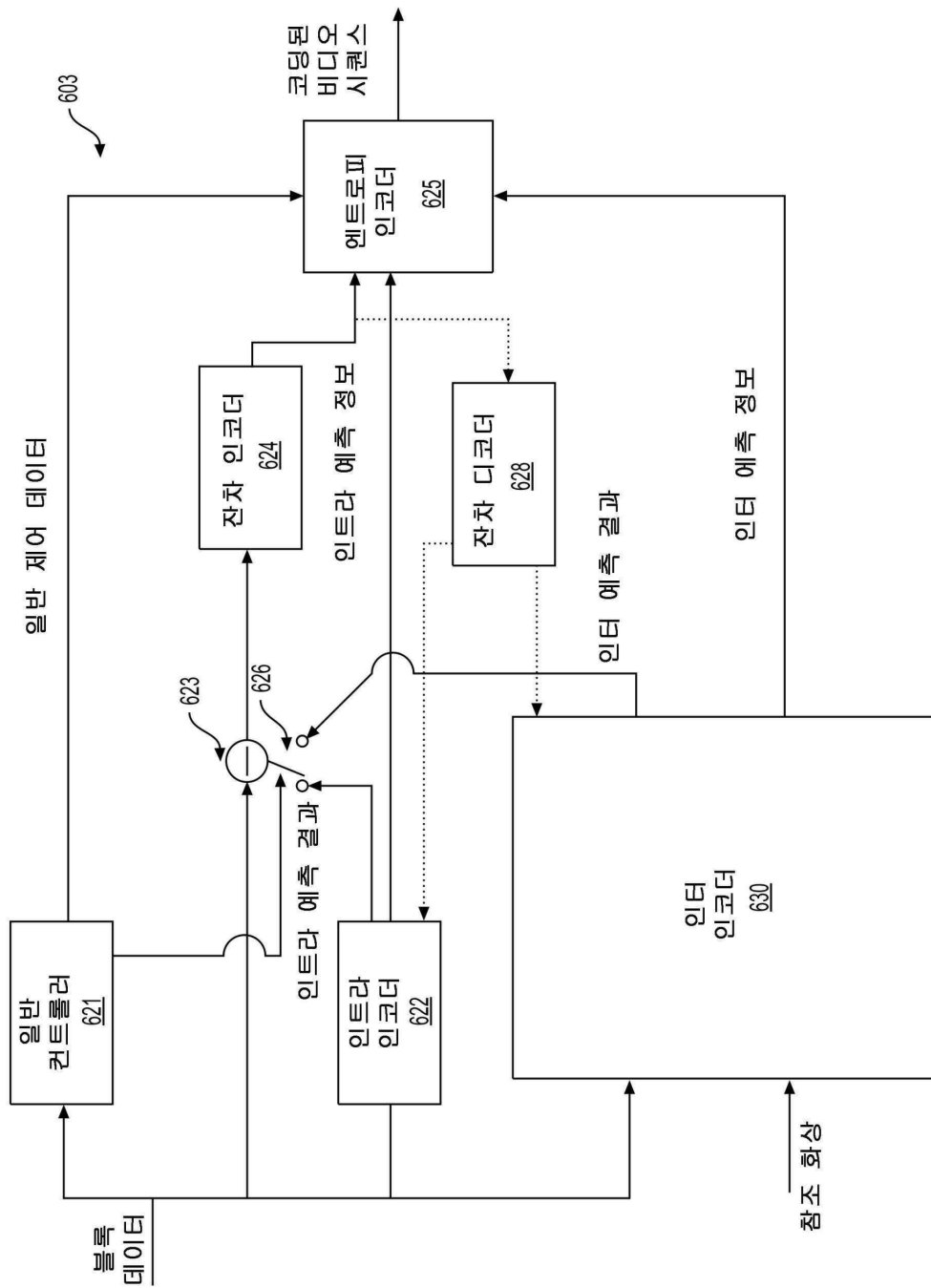
도면4



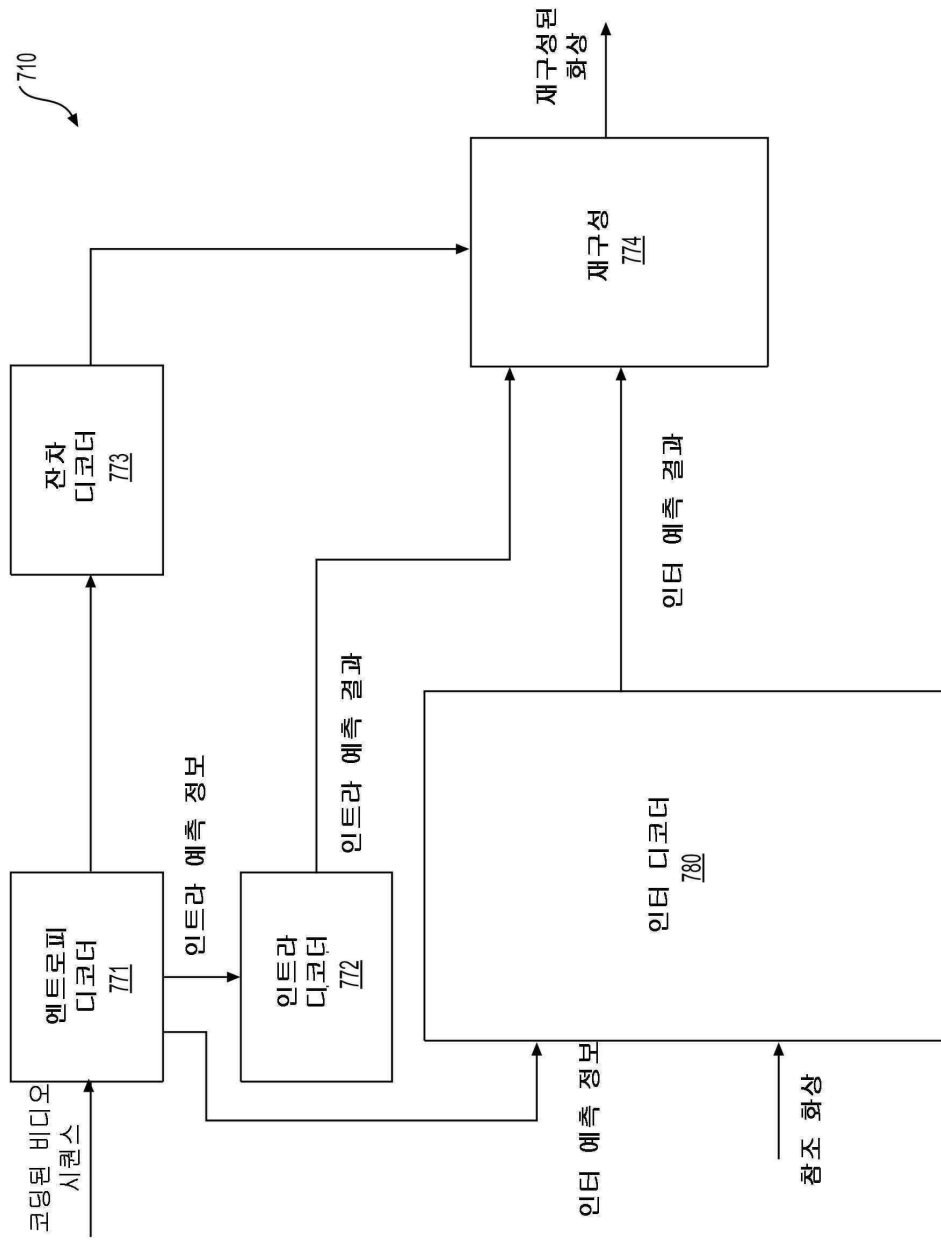
도면5



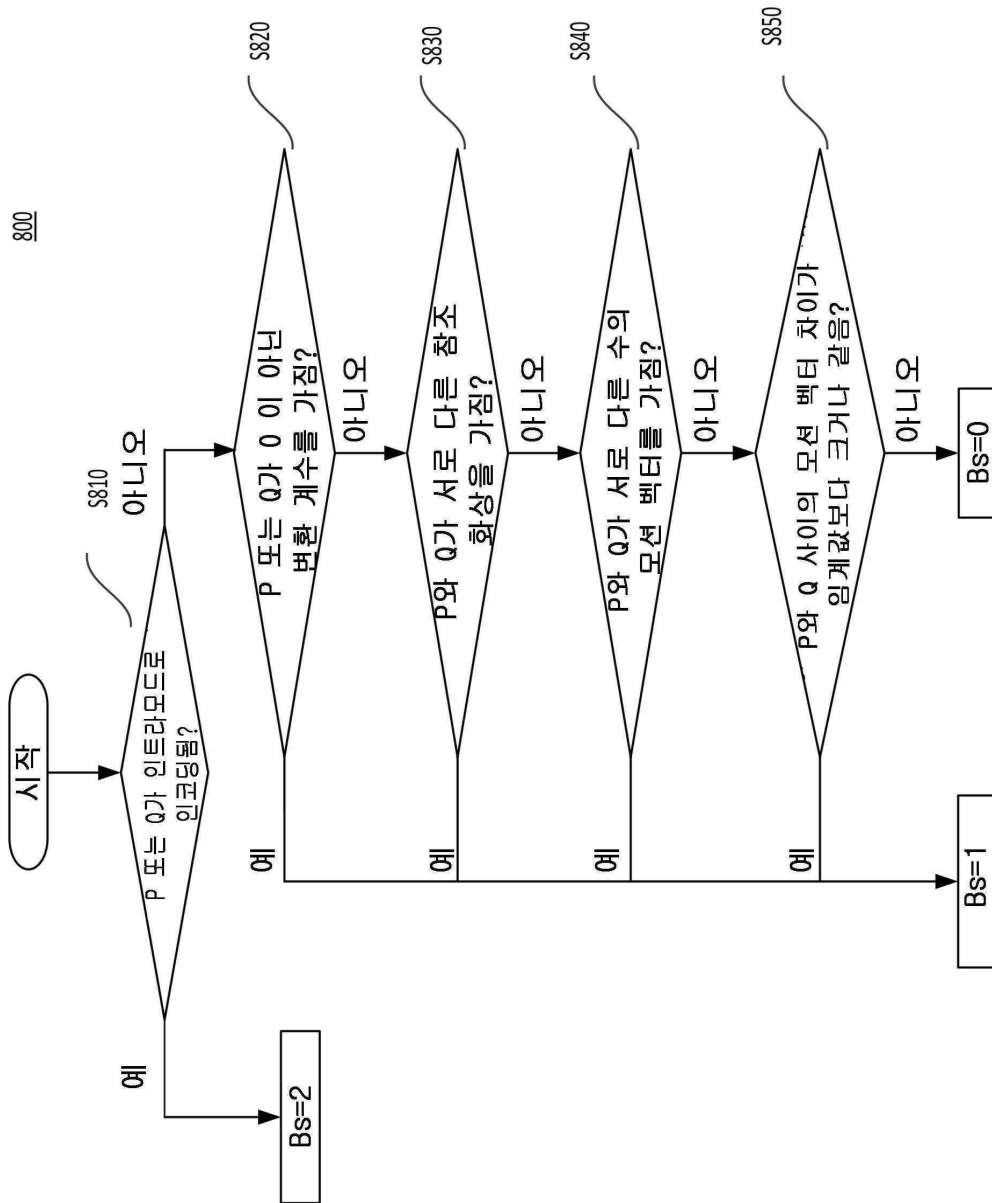
도면6



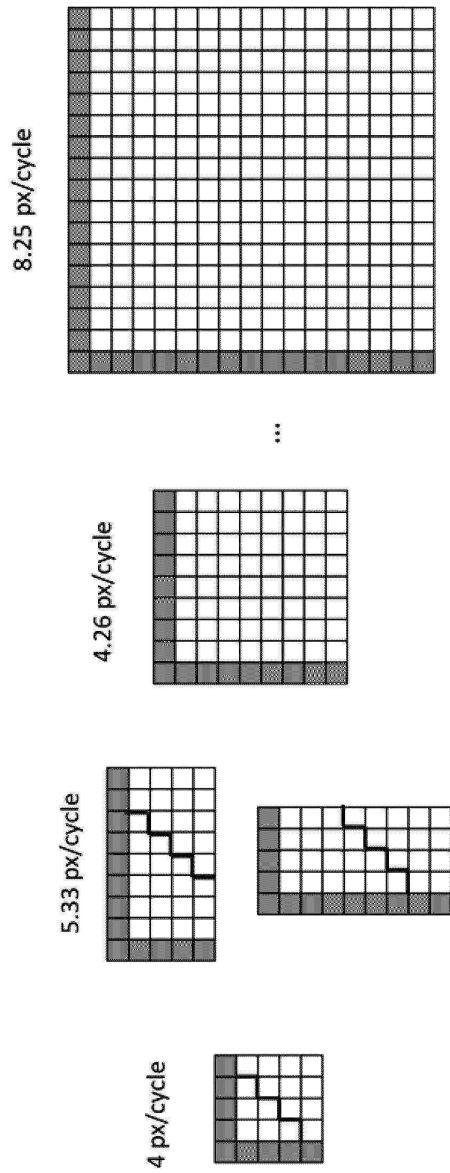
도면7



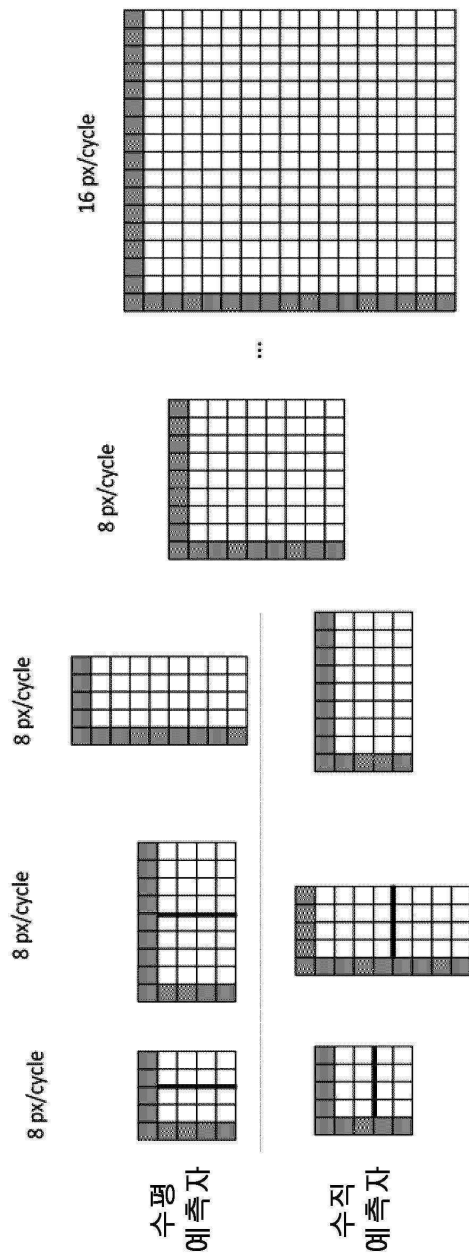
도면8



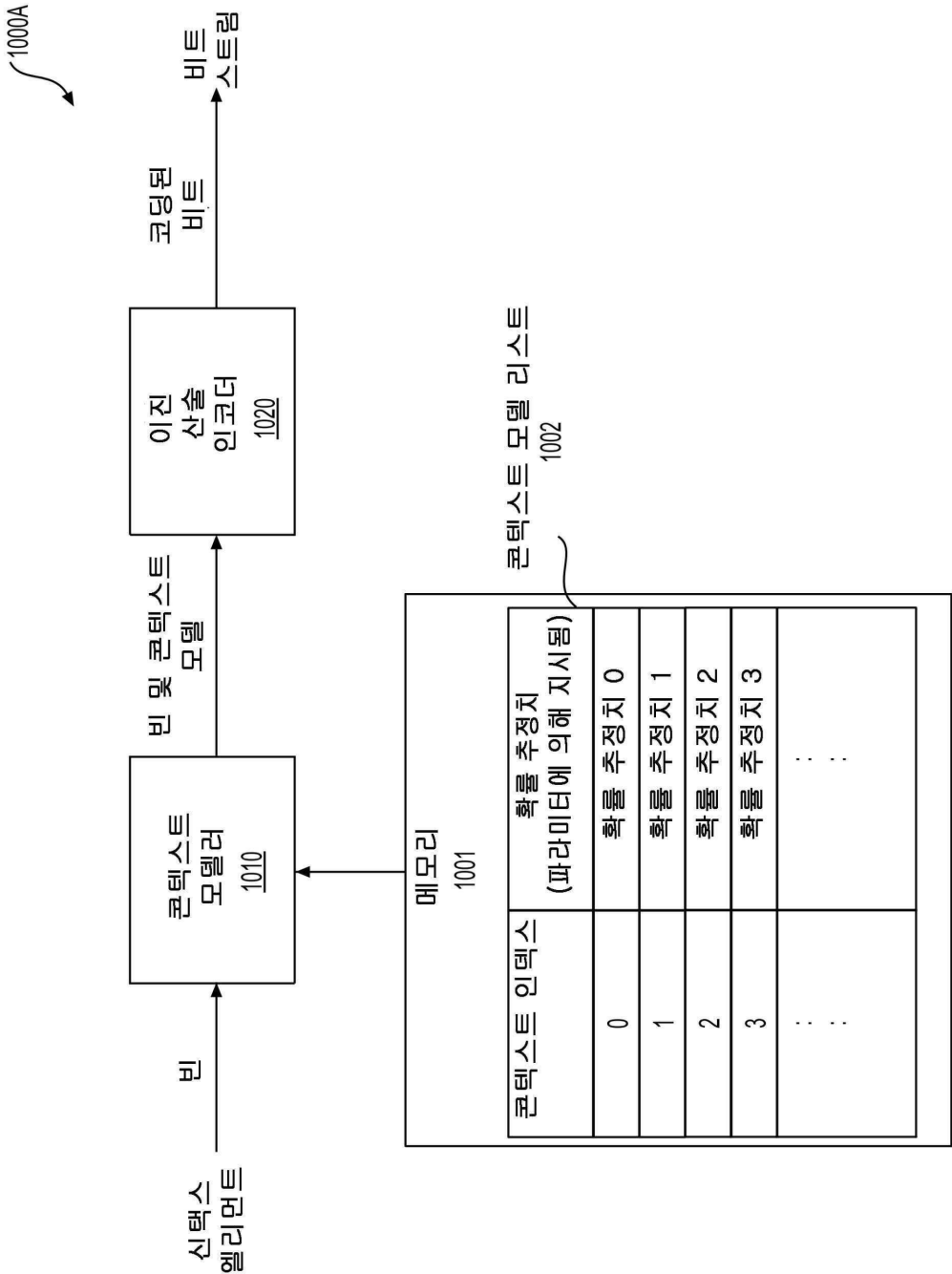
도면9a



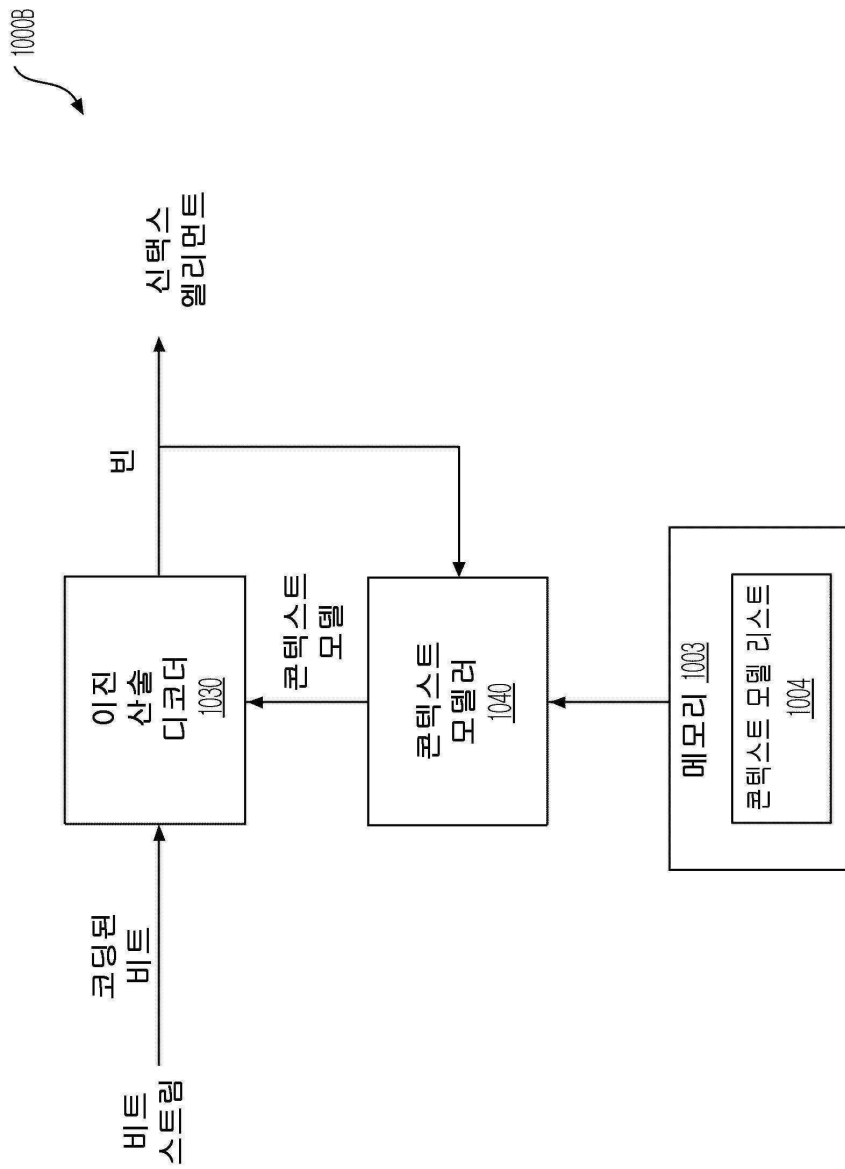
도면9b



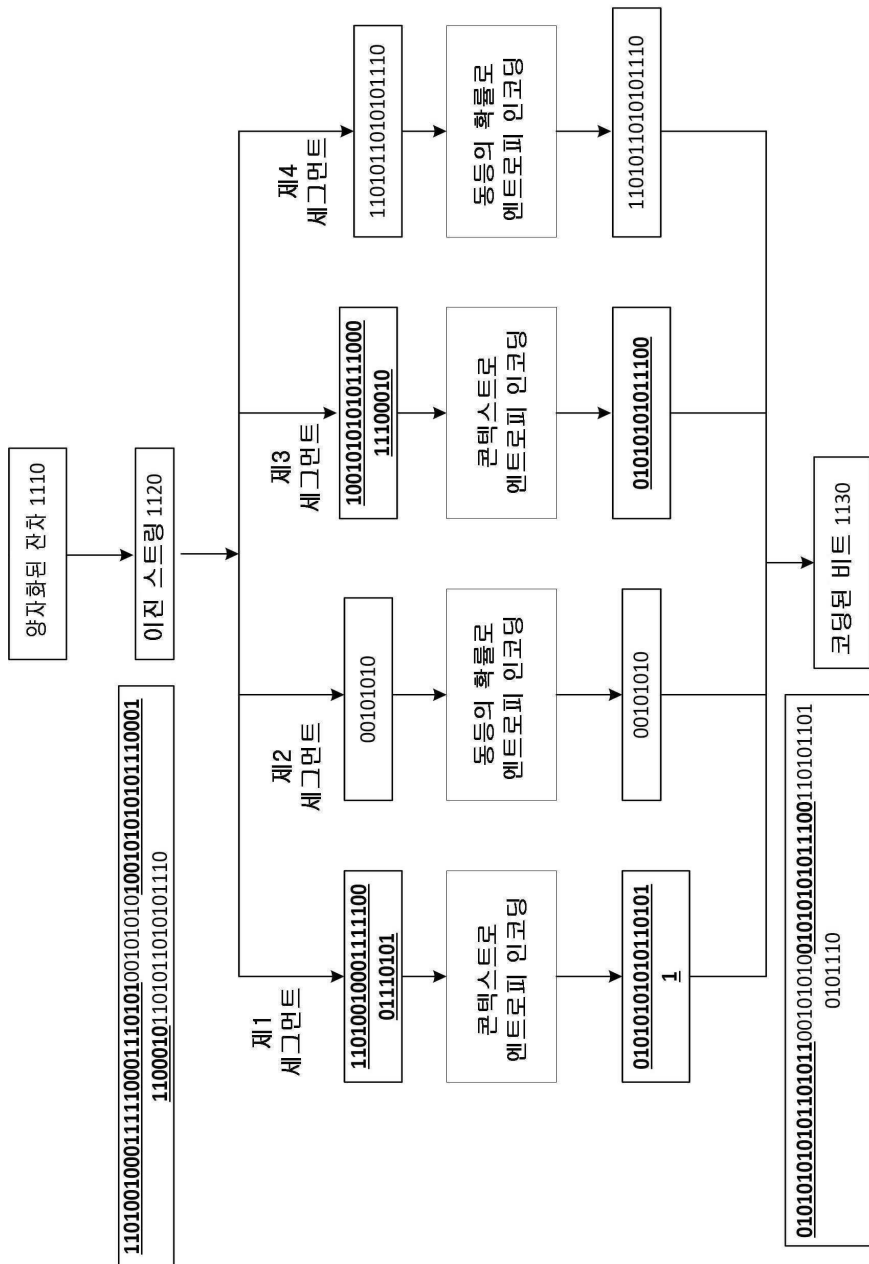
도면10a



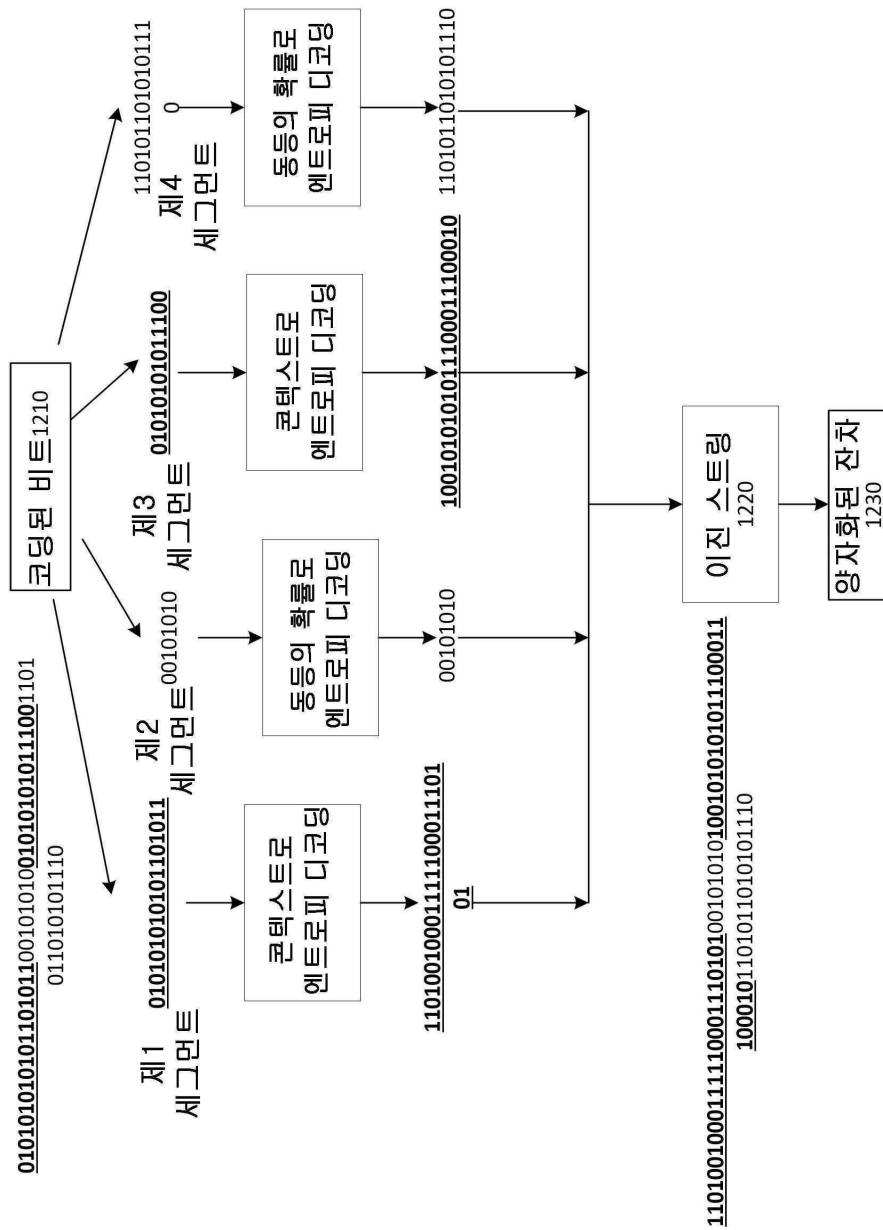
도면10b



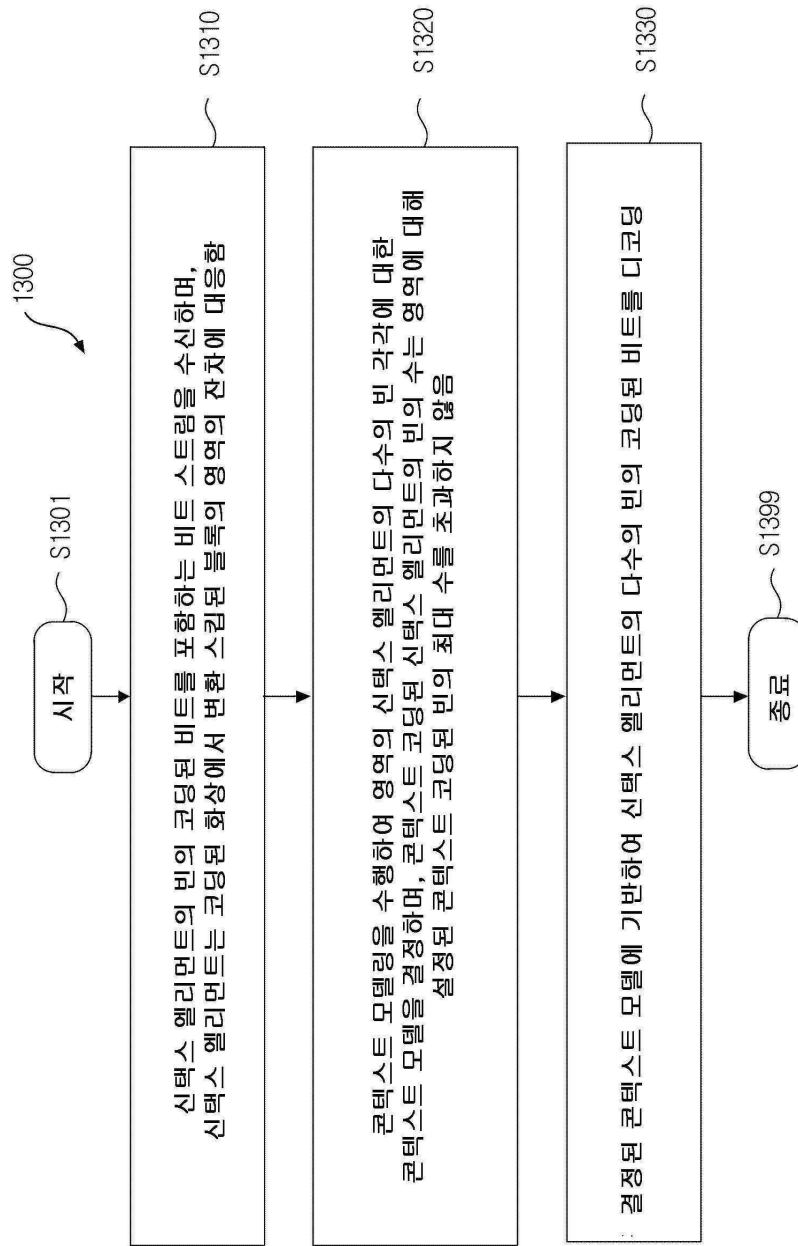
도면11



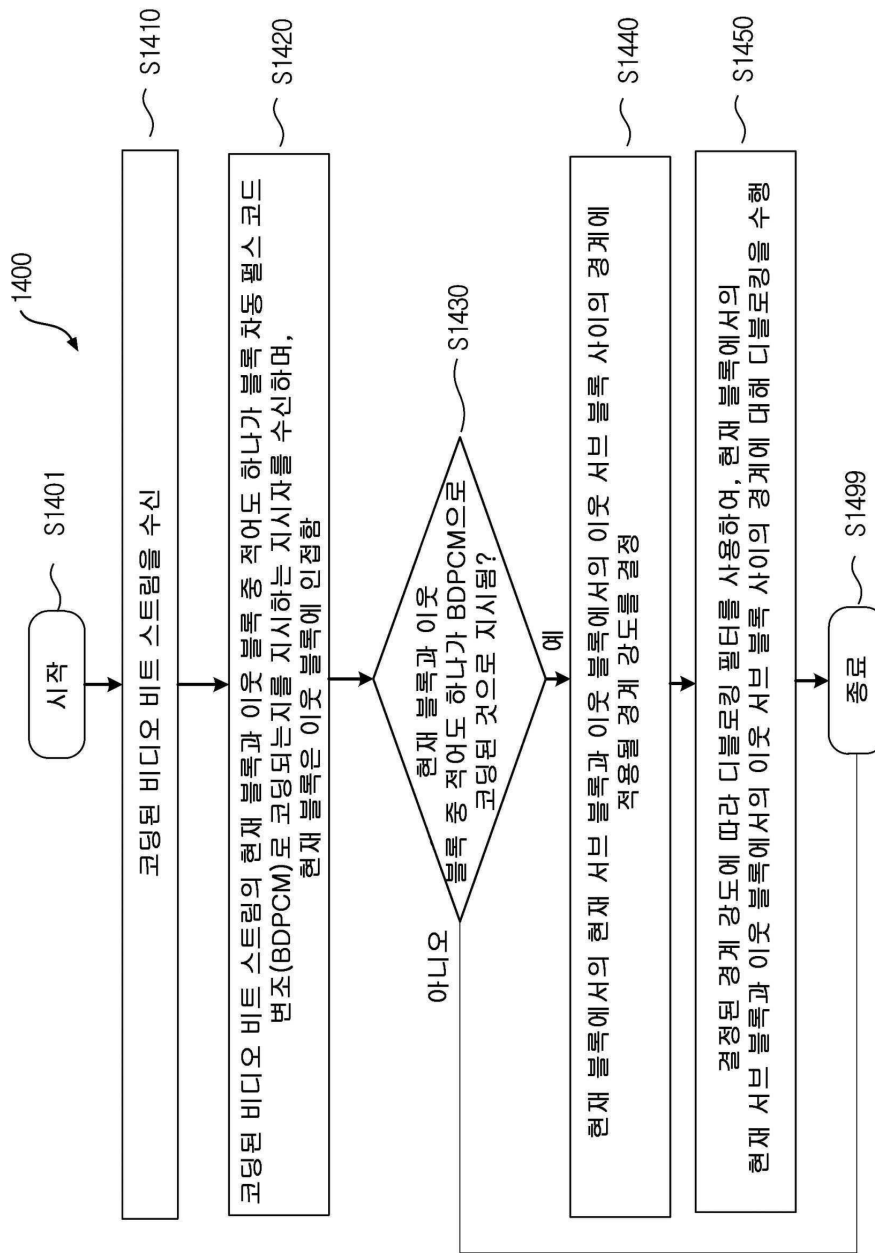
도면12



도면13



도면14



도면15

