



등록특허 10-2548954



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년06월27일
(11) 등록번호 10-2548954
(24) 등록일자 2023년06월23일

- (51) 국제특허분류(Int. Cl.)
HO4N 19/517 (2014.01) *HO4N 19/115* (2014.01)
HO4N 19/132 (2014.01) *HO4N 19/156* (2014.01)
HO4N 19/176 (2014.01) *HO4N 19/44* (2014.01)
HO4N 19/523 (2014.01) *HO4N 19/70* (2014.01)

- (52) CPC특허분류
HO4N 19/517 (2015.01)
HO4N 19/115 (2015.01)

(21) 출원번호 10-2020-7027584

(22) 출원일자(국제) 2019년03월06일
심사청구일자 2020년09월24일

(85) 번역문제출일자 2020년09월24일

(65) 공개번호 10-2020-0118896

(43) 공개일자 2020년10월16일

(86) 국제출원번호 PCT/US2019/021008

(87) 국제공개번호 WO 2019/173510

국제공개일자 2019년09월12일

(30) 우선권주장

62/639,862 2018년03월07일 미국(US)

16/182,788 2018년11월07일 미국(US)

(56) 선행기술조사문헌

US20160337662 A1*

*는 심사관에 의하여 인용된 문헌

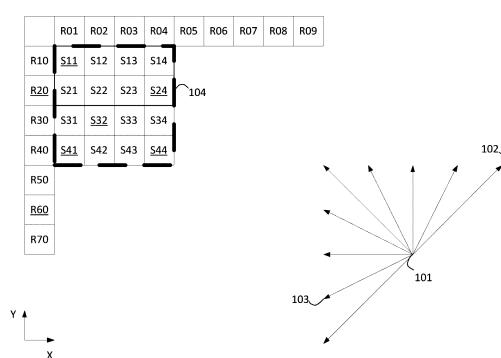
전체 청구항 수 : 총 19 항

심사관 : 박상철

(54) 발명의 명칭 비디오 코딩을 위한 방법 및 장치

(57) 요약

본 개시의 측면들은 비디오 코딩을 위한 방법 및 장치를 제공한다. 일부 예들에서, 장치는 비디오 디코딩을 위한 프로세싱 회로를 포함한다. 프로세싱 회로는, 코딩된 비디오 비트스트림으로부터 현재 블록의 예측 정보를 디코딩한다. 예측 정보는 인트라 블록 카피(intra block copy) 모드를 표시한다. 프로세싱 회로는, 다수의 후보 해상도 세트로부터, 현재 블록에 대한 블록 벡터 차이의 해상도를 선택하고, 블록 벡터 차이의 선택된 해상도 및 현재 블록의 블록 벡터 예측자에 따라 현재 블록의 블록 벡터를 결정한다. 다음으로, 프로세싱 회로는, 블록 벡터에 따라 현재 블록의 적어도 하나의 샘플을 재구성한다.

대 표 도 - 도1

(52) CPC특허분류

HO4N 19/132 (2015.01)

HO4N 19/156 (2015.01)

HO4N 19/176 (2015.01)

HO4N 19/44 (2015.01)

HO4N 19/523 (2015.01)

HO4N 19/70 (2015.01)

명세서

청구범위

청구항 1

비디오 디코딩을 위한 방법으로서,

코딩된 비디오 비트스트림으로부터 현재 블록의 예측 정보를 디코딩하는 단계 - 여기서, 상기 예측 정보는 인트라 블록 카피(intra block copy) 모드를 표시함 - ;

3개의 후보 해상도를 포함하는 세트로부터 상기 현재 블록에 대한 블록 벡터 차이의 해상도를 선택하는 단계 - 상기 선택하는 단계는,

제1 플래그에 기초하여 상기 세트에 포함된 제1 해상도가 사용되는지 여부를 결정하는 단계; 및

상기 제1 플래그가 상기 제1 해상도가 사용되지 않음을 지시할 때, 제2 플래그에 기초하여 상기 세트에 포함된 나머지 2개의 해상도 중 하나의 해상도를 선택하는 단계를 포함하고,

상기 제1 해상도와 상기 나머지 2개의 해상도는 3개의 서로 상이한 해상도이고, 상기 나머지 2개의 해상도는 상기 인트라 블록 카피 모드에 기초하는 1-integer-pel 및 4-integer-pel을 포함함 - ;

상기 블록 벡터 차이의 선택된 해상도 및 상기 현재 블록의 블록 벡터 예측자에 따라 상기 현재 블록의 블록 벡터를 결정하는 단계; 및

상기 블록 벡터에 따라 상기 현재 블록의 적어도 하나의 샘플을 재구성하는 단계를 포함하는

비디오 디코딩을 위한 방법.

청구항 2

제1항에 있어서,

상기 제1 플래그 또는 상기 제2 플래그는 1-bin 플래그인, 비디오 디코딩을 위한 방법.

청구항 3

제1항에 있어서,

상기 현재 블록에 대한 블록 벡터 차이의 해상도를 선택하는 단계는,

상기 블록 벡터 차이의 제1 성분에 대한 제1 해상도를 선택하는 단계; 및

상기 블록 벡터 차이의 제2 성분에 대한 제2 해상도를 선택하는 단계를 포함하는 비디오 디코딩을 위한 방법.

청구항 4

제3항에 있어서,

상기 블록 벡터 차이의 제1 성분에 대한 제1 해상도를 선택하는 단계는,

상기 제1 플래그에 기초하여 상기 블록 벡터 차이의 상기 제1 성분에 대한 상기 제1 해상도를 선택하는 단계를 포함하고;

상기 블록 벡터 차이의 제2 성분에 대한 제2 해상도를 선택하는 단계는,

상기 제2 플래그에 기초하여 상기 블록 벡터 차이의 상기 제2 성분에 대한 상기 제2 해상도를 선택하는 단계를 포함하는 비디오 디코딩을 위한 방법.

청구항 5

제3항에 있어서,

상기 블록 벡터 차이의 제1 성분에 대한 제1 해상도를 선택하는 단계는,

상기 블록 벡터 예측자의 제1 성분에 기초하여 상기 블록 벡터 차이의 상기 제1 성분에 대한 상기 제1 해상도를 선택하는 단계를 포함하고;

상기 블록 벡터 차이의 제2 성분에 대한 제2 해상도를 선택하는 단계는,

상기 블록 벡터 예측자의 제2 성분에 기초하여 상기 블록 벡터 차이의 상기 제2 성분에 대한 상기 제2 해상도를 선택하는 단계를 포함하는 비디오 디코딩을 위한 방법.

청구항 6

제5항에 있어서,

상기 블록 벡터 예측자의 제1 성분에 기초하여 상기 블록 벡터 차이의 상기 제1 성분에 대한 상기 제1 해상도를 선택하는 단계는,

상기 블록 벡터 예측자의 상기 제1 성분이 문턱값보다 작은 경우, 2 개의 해상도 세트로부터 디폴트 해상도를 상기 제1 해상도로서 선택하는 단계를 포함하고;

상기 블록 벡터 예측자의 제2 성분에 기초하여 상기 블록 벡터 차이의 상기 제2 성분에 대한 상기 제2 해상도를 선택하는 단계는,

상기 블록 벡터 예측자의 상기 제1 성분이 문턱값보다 큰 경우, 상기 2 개의 해상도 세트로부터 더 작은 해상도를 상기 제1 해상도로서 선택하는 단계를 포함하는 비디오 디코딩을 위한 방법.

청구항 7

제1항에 있어서,

상기 블록 벡터 예측자가, 상기 선택된 해상도와 다른 해상도를 사용한 경우, 상기 블록 벡터를 계산하기 위해 상기 블록 벡터 예측자를 반올림하지 않고, 상기 블록 벡터 예측자에 상기 블록 벡터 차이를 추가하는 단계를 더 포함하는 비디오 디코딩을 위한 방법.

청구항 8

제1항에 있어서,

상기 블록 벡터 예측자가, 상기 선택된 해상도와 다른 해상도를 갖는 경우, 상기 현재 블록의 상기 블록 벡터 예측자를 상기 선택된 해상도로 반올림하는 단계; 및

상기 블록 벡터를 계산하기 위해 상기 반올림된 블록 벡터 예측자에 블록 벡터 차이를 추가하는 단계를 더 포함하는 비디오 디코딩을 위한 방법.

청구항 9

제1항에 있어서,

유효 영역에서 상기 블록 벡터를 제한하기 위해, 상기 블록 벡터 차이 및 상기 블록 벡터 예측자 중 적어도 하나를 수정하는 단계를 더 포함하는 비디오 디코딩을 위한 방법.

청구항 10

비디오 디코딩을 위한 장치로서,

코딩된 비디오 비트스트림으로부터 현재 블록의 예측 정보를 디코딩하고 - 여기서, 상기 예측 정보는 인트라 블록 카피(intra block copy) 모드를 표시함 -;

3개의 후보 해상도를 포함하는 세트로부터 상기 현재 블록에 대한 블록 벡터 차이의 해상도를 선택하고 - 상기 선택하는 것은,

제1 플래그에 기초하여 상기 세트에 포함된 제1 해상도가 사용되는지 여부를 결정하는 것; 및

상기 제1 플래그가 상기 제1 해상도가 사용되지 않음을 지시할 때, 제2 플래그에 기초하여 상기 세트에

포함된 나머지 2개의 해상도 중 하나의 해상도를 선택하는 것을 포함하고,

상기 제1 해상도와 상기 나머지 2개의 해상도는 3개의 서로 상이한 해상도이고, 상기 나머지 2개의 해상도는 상기 인트라 블록 카피 모드에 기초하는 1-integer-pel 및 4-integer-pel을 포함함 - ;

상기 블록 벡터 차이의 선택된 해상도 및 상기 현재 블록의 블록 벡터 예측자에 따라 상기 현재 블록의 블록 벡터를 결정하고; 그리고

상기 블록 벡터에 따라 상기 현재 블록의 적어도 하나의 샘플을 재구성하도록 구성되는 프로세싱 회로를 포함하는

비디오 디코딩을 위한 장치.

청구항 11

제10항에 있어서,

상기 제1 플래그 또는 상기 제2 플래그는 1-bin 플래그인, 비디오 디코딩을 위한 장치.

청구항 12

제10항에 있어서,

상기 프로세싱 회로는,

상기 블록 벡터 차이의 제1 성분에 대한 제1 해상도를 선택하고;

상기 블록 벡터 차이의 제2 성분에 대한 제2 해상도를 선택하도록 추가로 구성되는, 비디오 디코딩을 위한 장치.

청구항 13

제12항에 있어서,

상기 프로세싱 회로는,

상기 제1 플래그에 기초하여 상기 블록 벡터 차이의 상기 제1 성분에 대한 상기 제1 해상도를 선택하고;

상기 제2 플래그에 기초하여 상기 블록 벡터 차이의 상기 제2 성분에 대한 상기 제2 해상도를 선택하도록 구성되는, 비디오 디코딩을 위한 장치.

청구항 14

제12항에 있어서,

상기 프로세싱 회로는,

상기 블록 벡터 예측자의 제1 성분에 기초하여 상기 블록 벡터 차이의 상기 제1 성분에 대한 상기 제1 해상도를 선택하고;

상기 블록 벡터 예측자의 제2 성분에 기초하여 상기 블록 벡터 차이의 상기 제2 성분에 대한 상기 제2 해상도를 선택하도록 추가로 구성되는, 비디오 디코딩을 위한 장치.

청구항 15

제14항에 있어서,

상기 프로세싱 회로는,

상기 블록 벡터 예측자의 상기 제1 성분이 문턱값보다 작은 경우, 2 개의 해상도 세트로부터 디폴트 해상도를 상기 제1 해상도로서 선택하고;

상기 블록 벡터 예측자의 상기 제1 성분이 문턱값보다 큰 경우, 상기 2 개의 해상도 세트로부터 더 작은 해상도를 상기 제1 해상도로서 선택하도록 추가로 구성되는, 비디오 디코딩을 위한 장치.

청구항 16

제10항에 있어서,

상기 프로세싱 회로는,

상기 블록 벡터 예측자가, 상기 선택된 해상도와 다른 해상도를 사용한 경우, 상기 블록 벡터를 계산하기 위해 상기 블록 벡터 예측자를 반올림하지 않고, 상기 블록 벡터 예측자에 상기 블록 벡터 차이를 추가하도록 추가로 구성되는, 비디오 디코딩을 위한 장치.

청구항 17

제10항에 있어서,

상기 프로세싱 회로는,

상기 블록 벡터 예측자가, 상기 선택된 해상도와 다른 해상도를 갖는 경우, 상기 현재 블록의 상기 블록 벡터 예측자를 상기 선택된 해상도로 반올림하고;

상기 블록 벡터를 계산하기 위해 상기 반올림된 블록 벡터 예측자에 블록 벡터 차이를 추가하도록 추가로 구성되는, 비디오 디코딩을 위한 장치.

청구항 18

제10항에 있어서,

상기 프로세싱 회로는,

유효 영역에서 상기 블록 벡터를 제한하기 위해, 상기 블록 벡터 차이 및 상기 블록 벡터 예측자 중 적어도 하나를 수정하도록 추가로 구성되는, 비디오 디코딩을 위한 장치.

청구항 19

명령을 저장하는 비 일시적인 컴퓨터로 판독 가능한 매체로서,

상기 명령은, 비디오 디코딩을 위해 컴퓨터에 의해 실행될 때 상기 컴퓨터로 하여금 제1항 내지 제9항 중 어느 한 항에 따른 방법을 수행하도록 하는,

비 일시적인 컴퓨터로 판독 가능한 매체.

청구항 20

삭제

발명의 설명**기술 분야**

[0001] 본 개시는 2018년 3월 7일에 출원된 "METHODS FOR ADAPTIVE MOTION AND BLOCK VECTOR RESOLUTIONS IN IMAGE AND VIDEO COMPRESSION" 명칭을 갖는 미국 가출원 제62/639,862호 및 2018년 11월 7일에 출원된 "METHOD AND APPARATUS FOR VIDEO CODING" 명칭을 갖는 미국 출원 제16/182,788호에 대한 우선권을 주장하며, 그 내용 전부는 참조에 의해 본 명세서에 포함된다.

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

배경 기술

[0003] 본 명세서에서 제공되는 배경기술 설명은 일반적으로 본 개시의 맥락을 제시하기 위한 것이다. 성과물이 배경기술 섹션에서 설명되는 범위까지, 현재 지명된 발명자들의 성과물 및 출원 시 선행 기술로서 달리 인정되지 않을 수 있는 설명의 측면들은, 본 개시에 대한 선행 기술로서 명시적 또는 묵시적으로 인정되지 않는다.

[0004] 비디오 코딩 및 디코딩은 모션 보상이 있는 인터 화상 예측(inter-picture prediction)을 사용하여 수행될 수 있다. 압축되지 않은 디지털 비디오는 일련의 화상들을 포함할 수 있으며, 각 화상은, 예를 들어 1920 x 1080

휘도(luminance) 샘플 및 연관된 색차(chrominance) 샘플의 공간 차원을 갖는다. 일련의 화상들은 고정 또는 가변 화상 레이트(비공식적으로 프레임 레이트라고도 함), 예를 들어 초당 60 개 화상 또는 60 Hz를 가질 수 있다. 압축되지 않은 비디오는 상당한 비트레이트(bitrate) 요구 사항을 갖는다. 예를 들어, 샘플 당 8 비트의 1080p60 4:2:0 비디오(60 Hz 프레임 레이트의 1920x1080 휘도 샘플 해상도)는 거의 1.5 Gbit/s 대역폭을 필요로 한다. 1 시간 길이의 이러한 비디오는 600 GByte 보다 큰 저장 공간을 필요로 한다.

[0005] 비디오 코딩 및 디코딩의 한 가지 목적은, 압축을 통해, 입력 비디오 신호의 중복성(redundancy)을 줄이는 것이 될 수 있다. 압축은, 경우에 따라 전술한 대역폭 또는 저장 공간 요구 사항을 2 배 이상 줄이는 데 도움이 될 수 있다. 무손실 압축과 손실 압축, 그리고 이들의 조합이 모두 사용될 수 있다. 무손실 압축은, 원본 신호의 정확한 사본아 압축된 원본 신호로부터 재구성될 수 있는 기술을 말한다. 손실 압축을 사용하는 경우, 재구성된 신호는 원본 신호와 동일하지 않을 수 있지만, 원본 신호와 재구성된 신호 사이의 왜곡은, 재구성된 신호가 의도하는 애플리케이션에 유용할 만큼 충분히 작다. 비디오의 경우, 손실 압축이 널리 사용된다. 허용되는 왜곡의 양은 응용에 따라 다르다; 예를 들어, 특정 소비자 스트리밍 애플리케이션의 사용자는, 텔레비전 기부 애플리케이션의 사용자보다 더 높은 왜곡을 견딜 수 있다. 달성 가능한 압축 비율은: 허용 가능한/견딜 수 있는 왜곡이 더 높을수록 더 높은 압축비를 얻을 수 있음을 반영할 수 있다.

[0006] 비디오 인코더 및 디코더는, 예를 들어 모션 보상, 변환, 양자화 및 엔트로피 코딩을 포함하여 여러 넓은 범주의 기술을 활용할 수 있다.

[0007] 비디오 코덱 기술에는 인트라 코딩이라는 기술이 포함될 수 있다. 인트라 코딩에서, 샘플 값은, 이전에 재구성된 참조 화상으로부터 샘플 또는 기타 데이터를 참조하지 않고 표현된다. 일부 비디오 코덱에서는, 화상이 샘플 블록으로 공간적으로 세분화된다. 모든 샘플 블록이 인트라 모드에서 코딩되는 경우, 해당 화상은 인트라 화상이 될 수 있다. 인트라 화상 및 독립 디코더 리프레시 화상(independent decoder refresh picture)과 같은 그 도출물(derivation)은 디코더 상태를 재설정하기 위해 사용될 수 있으므로, 코딩된 비디오 비트스트림 및 비디오 세션의 첫 번째 화상 또는 스틸 이미지로 사용될 수 있다. 인트라 블록의 샘플은 변환에 노출될 수 있으며, 변환 계수는 엔트로피 코딩 전에 양자화될 수 있다. 인트라 예측은, 변환 전 도메인(pre-transform domain)에서 샘플 값을 최소화하는 기술이 될 수 있다. 일부 경우에, 변환 후 DC 값이 더 작을수록, 그리고 AC 계수가 더 작을수록, 엔트로피 코딩 후 블록을 표현하기 위해 주어진 양자화 단계 크기에서 필요로 하는 비트의 수는 더 적다.

[0008] 예를 들어 MPEG-2 생성 코딩 기술로부터 알려진 것과 같은 전통적인 인트라 코딩은 인트라 예측을 사용하지 않는다. 그러나, 일부 새로운 비디오 압축 기술은, 예를 들어, 공간적으로 이웃하며, 디코딩 순서로 선행하는 데이터 블록의 인코딩/디코딩 중 획득한 메타 데이터 및/또는 주변 샘플 데이터로부터 시도하는 기술을 포함한다. 이러한 기술은 이하에서 "인트라 예측" 기술이라고 한다. 적어도 일부 경우에, 인트라 예측은, 참조 화상으로부터가 아닌, 재구성 중인 현재 화상으로부터의 참조 데이터만을 사용함을 주목해야 한다.

[0009] 다양한 형태의 인트라 예측이 존재할 수 있다. 이러한 기술 중 2 이상이, 주어진 비디오 코딩 기술에서 사용될 수 있는 경우, 사용 중인 기술은 인트라 예측 모드에서 코딩될 수 있다. 특정 경우에, 모드는 하위 모드 및/또는 파라미터를 가질 수 있으며, 이들은 개별적으로 코딩되거나, 모드 코드워드(codeword)에 포함될 수 있다. 주어진 모드/하위 모드/파라미터 조합에 어떤 코드워드를 사용할 것인지는, 인트라 예측을 통한 코딩 효율 이득에 영향을 미칠 수 있으며, 코드워드를 비트스트림으로 변환하기 위해 사용되는 엔트로피 코딩 기술도 마찬가지이다.

[0010] 특정 모드의 인트라 예측이 H.264로 도입되었고, H.265에서 개선되었으며, 공동 탐사 모델(joint exploration model, JEM), 다목적 비디오 코딩(versatile video coding, VVC) 및 벤치마크 세트(BMS)와 같은 최신 코딩 기술에서 더욱 개선되었다. 이미 사용 가능한 샘플에 속하는 이웃 샘플 값을 사용하여 예측자 블록(predictor block)이 형성될 수 있다. 이웃 샘플의 샘플 값은, 방향에 따라 예측자 블록으로 복사된다. 사용되는 방향에 대한 참조은, 비트스트림에 코딩될 수 있거나, 자체적으로 예측될 수 있다.

[0011] 도 1을 참조하면, 우측 하단에 표시된 것은 H.265의 35 가지 가능한 예측자 방향(predictor direction)에서 알려진 9 개의 예측자 방향의 서브세트이다. 화살표가 수렴하는 지점(101)은 예측되는 샘플을 나타낸다. 화살표는 샘플이 예측되는 방향을 나타낸다. 예를 들어, 화살표(102)는, 샘플(101)이, 수평으로부터 45 도 각도로, 우측 상단을 향하는 샘플 또는 샘플들로부터 예측됨을 나타낸다. 이와 유사하게, 화살표(103)는, 샘플(101)이, 수평으로부터 22.5도 각도로, 샘플(101)의 좌측 아래를 향하는 샘플 또는 샘플들로부터 예측됨을 나타낸다.

- [0012] 도 1을 계속 참조하면, 도면의 좌측 상단에는 4×4 샘플들의 사각형 블록(104)이 도시되어 있다(굵은 점선으로 표시됨). 정사각형 블록(104)은, 각각 "S", Y 차원에서의 위치(예를 들어, 행 인덱스) 및 X 차원에서의 위치(예를 들어, 열 인덱스)로 라벨링된 16 개의 샘플들을 포함한다. 예를 들어, 샘플 S21은 Y 차원에서 (위에서부터) 2 번째 샘플이고, X 차원에서 (왼쪽에서부터) 1 번째 샘플이다. 이와 유사하게, 샘플 S44는 Y 및 X 차원 모두에서 블록(104)의 4 번째 샘플이다. 블록의 크기가 4×4 샘플이므로, S44는 우측 하단에 있다. 추가로 도시된 것들은, 유사한 번호 부여 방식을 따르는 참조 샘플들이다. 참조 샘플은, 블록(104)을 참조으로 R, Y 위치(예를 들어, 행 인덱스) 및 X 위치(열 인덱스)로 라벨링된다. H.264 및 H.265 모두에서, 예측 샘플은 재구성 중인 블록에 이웃하며; 따라서 음의 값을 사용할 필요가 없다.
- [0013] 인트라 화상 예측은, 시그널링된 예측 방향에 의해 적절한 이웃 샘플로부터 참조 샘플 값을 복사하여 작동할 수 있다. 예를 들어, 코딩된 비디오 비트스트림이, 이 블록에 대해, 화살표(102)와 일치하는 예측 방향을 표시하는 시그널링을 포함, 즉, 샘플이, 수평으로부터 45 도 각도로, 우측 상단을 향하는 예측 샘플 또는 샘플들로부터 예측된 것으로 가정한다. 이 경우, 샘플 S41, S32, S23 및 S14는 동일한 R05로부터 예측된다. 다음으로, 샘플 S44는 R08로부터 예측된다.
- [0014] 특정 경우에, 참조 샘플을 계산하기 위해, 예를 들어 보간을 통해, 여러 참조 샘플들의 값을 결합할 수 있다; 특히 방향들이 45 도로 균등하게 나누어질 수 없는 경우.
- [0015] 비디오 코딩 기술이 발전함에 따라, 가능한 방향의 개수가 증가했다. H.264(2003 년)에서는 9 가지의 다른 방향을 나타낼 수 있었다. 그것은 H.265(2013 년)에 33 가지로 증가했으며, 개시 시점에, JEM/VVC/BMS는 최대 65 가지의 방향을 지원할 수 있다. 가능성이 가장 높은 방향을 식별하기 위해 실험이 수행되었으며, 엔트로피 코딩의 특정 기술을 사용하여, 가능성이 보다 적은 방향에 대한 특정 패널티를 수용하여 이를 가능성이 있는 방향들을 적은 개수의 비트로 표현한다. 또한, 방향 자체는 때때로, 이미 디코딩된 이웃 블록에서 사용되는 이웃 방향으로부터 예측될 수 있다.
- [0016] 도 2는, 시간에 따른 예측 방향의 개수가 증가하는 것을 설명하기 위해 JEM에 따른 65 가지 인트라 예측 방향을 나타내는 개략도(201)를 도시한다.
- [0017] 방향을 나타내는 코딩된 비디오 비트스트림에서 인트라 예측 방향 비트의 매핑은, 비디오 코딩 기술 별로 다를 수 있고; 예를 들어, 코드워드에 대한, 인트라 예측 모드에 대한 예측 방향의 단순한 직접 매핑부터, 대부분의 가능한 모드 및 유사 기술을 포함하는 복잡한 적응적 스킴까지의 범위를 가질 수 있다. 본 발명의 기술 분야에서 통상의 지식을 가진 자는 이를 기술들에 대해 용이하게 익숙하다. 그러나, 모든 경우에, 비디오 콘텐츠에서 통계적으로 다른 특정 방향보다 발생할 가능성이 적은 특정 방향이 존재할 수 있다. 비디오 압축의 목표는 중복성을 줄이는 것이므로, 잘 작동하는 비디오 코딩 기술에서, 가능성이 낮은 방향은 가능성이 높은 방향보다 더 많은 개수의 비트로 표시된다.
- 발명의 내용**
- [0018] 본 개시의 측면들은 비디오 코딩을 위한 방법 및 장치를 제공한다.
- 일부 예들에서, 디코더에서 비디오 코딩을 위한 방법은, 코딩된 비디오 비트스트림으로부터 현재 블록의 예측 정보를 디코딩하는 단계 - 여기서 예측 정보는 인트라 블록 카피(intra block copy) 모드를 표시함 -; 다수의 후보 해상도 세트로부터, 현재 블록에 대한 블록 벡터 차이의 해상도를 선택하는 단계; 블록 벡터 차이의 선택된 해상도 및 현재 블록의 블록 벡터 예측자에 따라 현재 블록의 블록 벡터를 결정하는 단계; 및 블록 벡터에 따라 현재 블록의 적어도 하나의 샘플을 재구성하는 단계를 포함한다.
- 일부 예들에서, 장치는 비디오 디코딩을 위한 프로세싱 회로를 포함한다. 프로세싱 회로는, 코딩된 비디오 비트스트림으로부터 현재 블록의 예측 정보를 디코딩한다. 예측 정보는 인트라 블록 카피 모드를 표시한다. 프로세싱 회로는, 다수의 후보 해상도 세트로부터, 현재 블록에 대한 블록 벡터 차이의 해상도를 선택하고, 블록 벡터 차이의 선택된 해상도 및 현재 블록의 블록 벡터 예측자에 따라 현재 블록의 블록 벡터를 결정한다. 다음으로, 프로세싱 회로는, 블록 벡터에 따라 현재 블록의 적어도 하나의 샘플을 재구성한다.
- [0019] 삭제

[0020]	삭제
[0021]	삭제
[0022]	삭제
[0023]	삭제
[0024]	삭제

[0025] 본 개시의 측면들은 또한, 비디오 디코딩을 위해 컴퓨터에 의해 실행될 때 컴퓨터로 하여금 비디오 디코딩을 위한 방법을 수행하도록 하는 명령을 저장하는 비 일시적인 컴퓨터로 판독 가능한 매체를 제공한다.

도면의 간단한 설명

[0026] 개시된 대상(subject matter)에 대한 추가적인 특징들, 특성들 및 다양한 이점들은, 다음의 상세한 설명 및 첨부 도면으로부터 더욱 명백해질 것이다.

도 1은 H.265에 따른 인트라 예측 모드의 서브세트의 개략도이다.

도 2는 JEM에 따른 인트라 예측 방향을 설명하기 위한 도면이다.

도 3은 일 실시 예에 따른 통신 시스템의 단순화된 블록도에 해당하는 개략도이다.

도 4는 일 실시 예에 따른 통신 시스템의 단순화된 블록도에 해당하는 개략도이다.

도 5는 일 실시 예에 따른 디코더의 단순화된 블록도에 해당하는 개략도이다.

도 6은 일 실시 예에 따른 인코더의 단순화된 블록도에 해당하는 개략도이다.

도 7은 다른 실시 예에 따른 인코더의 블록도를 나타낸다.

도 8은 다른 실시 예에 따른 디코더의 블록도를 나타낸다.

도 9는 본 개시의 일 실시 예에 따른 인트라 블록 카페의 일 예를 나타낸다.

도 10은 본 개시의 일 실시 예에 따른 프로세스를 개략적으로 설명하는 흐름도를 나타낸다.

도 11은 일 실시 예에 따른 컴퓨터 시스템의 개략도이다.

발명을 실시하기 위한 구체적인 내용

[0027] 도 3은 본 개시의 일 실시 예에 따른 통신 시스템(300)의 단순화된 블록도를 도시한다. 통신 시스템(300)은, 예를 들어, 네트워크(350)를 통해 서로 통신할 수 있는 복수의 단말 디바이스를 포함한다. 예를 들어, 통신 시스템(300)은, 네트워크(350)를 통해 상호 연결된 제1 쌍의 단말 디바이스(310 및 320)를 포함한다. 도 3에서, 제1 쌍의 단말 디바이스(310 및 320)는 데이터의 단방향 전송을 수행한다. 예를 들어, 단말 디바이스(310)는, 네트워크(350)를 통해 다른 단말 디바이스(320)에 전송하기 위해 비디오 데이터(예를 들어, 단말 디바이스(310)에 의해 캡처되는 비디오 화상의 스트림)를 코딩할 수 있다. 인코딩된 비디오 데이터는, 하나 이상의 코딩된 비디오 비트스트림의 형태로 전송될 수 있다. 단말 디바이스(320)는 네트워크(350)로부터 코딩된 비디오 데이터를 수신하고, 코딩된 비디오 데이터를 디코딩하여 비디오 화상을 복원하고, 복원된 비디오 데이터에 따라 비디오 화상을 표시할 수 있다. 단방향 데이터 전송은, 미디어 서빙 애플리케이션 (media serving application) 등에 일반적일 수 있다.

[0028] 다른 예에서, 통신 시스템(300)은, 예를 들어, 화상 회의 중 발생할 수 있는 코딩된 비디오 데이터의 양방향 전송을 수행하는 제2 쌍의 단말 디바이스(330, 340)를 포함한다. 데이터의 양방향 전송을 위해, 일 예로, 단말 디

바이스(330, 340)의 각 단말 디바이스는, 네트워크(350)를 통해 단말 디바이스(330, 340)의 다른 단말 디바이스에 전송하기 위해 비디오 데이터(예를 들어, 단말 디바이스에 의해 캡처되는 비디오 화상의 스트림)를 코딩할 수 있다. 단말 디바이스(330, 340)의 각 단말 디바이스는 또한, 단말 디바이스(330, 340)의 다른 단말 디바이스에 의해 전송되는 코딩된 비디오 데이터를 수신할 수도 있으며, 코딩된 비디오 데이터를 디코딩하여 비디오 화상을 복원할 수 있고, 복원된 비디오 데이터에 따라 액세스 가능한 표시 디바이스에 비디오 화상을 표시할 수 있다.

[0029] 도 3의 예에서, 단말 디바이스(310, 320, 330, 340)는, 서버, 개인용 컴퓨터 및 스마트 폰으로 예시될 수 있지만, 본 개시의 원리는 이에 제한되지 않을 수 있다. 본 개시의 실시 예들은 랩톱 컴퓨터, 태블릿 컴퓨터, 미디어 플레이어 및/또는 전용 화상 회의 장비를 갖는 애플리케이션을 인지한다. 네트워크(350)는, 예를 들어 유선(wired) 및/또는 무선 통신 네트워크를 포함하여 단말 디바이스(310, 320, 330, 340) 사이에서, 코딩된 비디오 데이터를 전달하는 임의의 수의 네트워크를 나타낸다. 통신 네트워크(350)는 회선 교환 방식 및/또는 패킷 교환 방식 채널에서 데이터를 교환할 수 있다. 대표적인 네트워크는 통신 네트워크, 근거리 네트워크, 광역 네트워크 및/또는 인터넷을 포함한다. 본 논의를 위해, 네트워크(350)의 아키텍처 및 토폴로지는 여기 아래에서 설명되지 않는 한 본 개시의 동작에 중요하지 않을 수 있다.

[0030] 도 4는 개시된 대상에 대한 애플리케이션의 일 예로서, 스트리밍 환경에서의 비디오 인코더 및 비디오 디코더의 배치를 도시한다. 개시된 대상은, 예를 들어, 화상 회의, 디지털 TV, CD, DVD, 메모리 스틱 등을 포함하는 디지털 미디어에 압축된 비디오의 저장을 포함하는 다른 비디오 지원 애플리케이션(video enabled application)에도 동일하게 적용될 수 있다.

[0031] 스트리밍 시스템은, 비디오 소스(401), 예를 들어 디지털 카메라를 포함할 수 있는, 캡처 서브시스템(413)을 포함할 수 있으며, 이는 예를 들어 압축되지 않은 비디오 화상의 스트림(402)을 생성한다. 일 예에서, 비디오 화상 스트림(402)은 디지털 카메라에 의해 촬영된 샘플을 포함한다. 인코딩된 비디오 데이터(404)(또는 코딩된 비디오 비트스트림)와 비교할 때, 더 높은 데이터 볼륨을 강조하기 위해 짙은 선으로 표시된 비디오 화상 스트림(402)은, 비디오 소스(401)에 결합된 비디오 인코더(403)를 포함하는 전자 디바이스(420)에 의해 처리될 수 있다. 비디오 인코더(403)는 하드웨어, 소프트웨어, 또는 이들의 조합을 포함하여, 아래에서 더욱 상세하게 설명되는 바와 같이, 개시된 대상의 측면들을 가능하도록 하거나 구현할 수 있다. 비디오 화상의 스트림(402)과 비교할 때, 더 낮은 데이터 볼륨을 강조하기 위해 가는 선으로 표시된, 인코딩된 비디오 데이터(404)(또는 인코딩된 비디오 비트스트림(404))는, 향후 사용을 위해 스트리밍 서버(405)에 저장될 수 있다. 도 4의 클라이언트 서브시스템(406, 408)과 같은, 하나 이상의 스트리밍 클라이언트 서브시스템은, 인코딩된 비디오 데이터(404)의 카피(407, 409)를 검색하기 위해 스트리밍 서버(405)에 액세스할 수 있다. 클라이언트 서브시스템(406)은, 예를 들어, 전자 디바이스(430)에 비디오 디코더(410)를 포함할 수 있다. 비디오 디코더(410)는, 인코딩된 비디오 데이터의 인입(incoming) 카피(407)를 디코딩하고, 디스플레이(412)(예를 들어, 디스플레이 스크린) 또는 다른 렌더링 디바이스(도시되지 않음) 상에서 렌더링될 수 있는 비디오 화상(411)의 인출(outgoing) 스트림을 생성한다. 일부 스트리밍 시스템에서, 인코딩된 비디오 데이터(404, 407, 409)(예를 들어, 비디오 비트스트림)는 특정 비디오 코딩/압축 표준에 따라 인코딩될 수 있다. 이러한 표준의 예로는 ITU-T 권고 H.265가 있다. 일 예로, 개발 중인 비디오 코딩 표준은 비공식적으로 다목적 비디오 코딩(Versatile Video Coding) 또는 VVC로 알려져 있다. 개시된 대상은 VVC의 맥락에서 사용될 수 있다.

[0032] 전자 디바이스(420, 430)는 다른 구성 요소(도시하지 않음)를 포함할 수 있음에 유의해야 한다. 예를 들어, 전자 디바이스(420)는 비디오 디코더(도시되지 않음)를 포함할 수 있고, 전자 디바이스(430) 또한 비디오 인코더(도시되지 않음)를 포함할 수 있다.

[0033] 도 5는 본 개시의 일 실시 예에 따른 비디오 디코더(510)의 블록도를 도시한다. 비디오 디코더(510)는 전자 디바이스(530)에 포함될 수 있다. 전자 디바이스(530)는 수신기(531)(예를 들어, 수신 회로)를 포함할 수 있다. 비디오 디코더(510)는, 도 4 예의 비디오 디코더(410) 대신에 사용될 수 있다.

[0034] 수신기(531)는 비디오 디코더(510)에 의해 디코딩될 하나 이상의 코딩된 비디오 시퀀스를 수신할 수 있고; 동일하거나 다른 실시 예에서, 한 번에 하나의 코딩된 비디오 시퀀스이고, 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(501)로부터 수신될 수 있다. 수신기(531)는 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림과 함께, 인코딩된 비디오 데이터를 수신할 수 있으며, 이들은 각각의 사용 엔티티(도시되지 않음)에 전달될 수 있다. 수신기(531)는 코딩된 비디오 시퀀스

를 다른 데이터로부터 분리할 수 있다. 네트워크 지터(network jitter)를 방지하기 위해, 버퍼 메모리(515)가 수신기(531)와 엔트로피 디코더/파서(520)(이하 "파서(520)") 사이에 결합될 수 있다. 특정 애플리케이션에서, 버퍼 메모리(515)는 비디오 디코더(510)의 일부이다. 다른 경우에는, 비디오 디코더(510)(도시되지 않음)의 외부에 있을 수 있다. 또 다른 경우에는, 예를 들어 네트워크 지터를 방지하기 위해, 비디오 디코더(510) 외부에 버퍼 메모리(도시되지 않음)가 있을 수 있으며, 예를 들어 플레이아웃 타이밍(playout timing)을 처리하기 위해, 추가로 비디오 디코더(510) 내부에 다른 버퍼 메모리(515)가 있을 수 있다. 수신기(531)가 충분한 대역폭 및 제어 가능성을 갖는 저장/전달 디바이스로부터, 또는 동시성 네트워크로부터 데이터를 수신하는 경우, 버퍼 메모리(515)는 필요하지 않거나, 작을 수 있다. 인터넷과 같은 최선 노력(best effort) 패킷 네트워크에서 사용하기 위해, 버퍼 메모리(515)가 필요할 수 있고, 비교적 클 수 있고, 적응형 크기가 유리할 수 있으며, 운영 체제 또는 비디오 디코더(510) 외부의 유사한 요소(도시되지 않음)에서 적어도 부분적으로 구현될 수 있다.

[0035] 비디오 디코더(510)는 코딩된 비디오 시퀀스로부터 심볼(521)을 재구성하기 위해 파서(520)를 포함할 수 있다. 이러한 심볼의 범주에는, 도 5에 도시된 바와 같이, 비디오 디코더(510)의 동작을 관리하기 위해 사용되는 정보, 전자 디바이스(530)의 필수 부분은 아니지만 전자 디바이스(530)에 결합 가능한 렌더링 디바이스(512)(예를 들어, 디스플레이 스크린)와 같은 렌더링 디바이스를 제어하기 위한 잠재적인 정보가 포함된다. 렌더링 디바이스(들)에 대한 제어 정보는, 보충 강화 정보(Supplementary Enhancement Information)(SEI 메시지) 또는 비디오 사용성 정보(Video Usability Information, VUI) 파라미터 세트 프래그먼트(fragments)(도시되지 않음)의 형태일 수 있다. 파서(520)는, 수신되는 코딩된 비디오 시퀀스를 파싱/엔트로피 디코딩할 수 있다. 코딩된 비디오 시퀀스의 코딩은 비디오 코딩 기술 또는 표준에 따를 수 있으며, 가변 길이 코딩, 허프만(Huffman) 코딩, 맥락 민감도(context sensitivity)가 있거나 없는 산술 코딩 등을 포함하는 다양한 원칙들을 따를 수 있다. 파서(520)는, 그룹에 대응하는 적어도 하나의 파라미터에 기초하여, 비디오 디코더 내의 팩셀의 서브 그룹 중 적어도 하나에 대한 서브 그룹 파라미터의 세트를, 코딩된 비디오 시퀀스로부터 추출할 수 있다. 서브 그룹에는 화상 그룹(Group of Pictures, GOPs), 화상, 타일, 슬라이스, 매크로블록, 코딩 유닛(Coding Units, CUs), 블록, 변환 유닛(Transform Units, TUs), 예측 유닛(Prediction Units, PUs) 등이 포함될 수 있다. 파서(520)는 또한, 변환 계수, 양자화기 파라미터 값, 모션 벡터 등과 같은, 코딩된 비디오 시퀀스 정보로부터 추출할 수도 있다.

[0036] 파서(520)는 버퍼 메모리(515)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩/파싱 동작을 수행하여, 심볼(521)을 생성할 수 있다.

[0037] 심볼(521)의 재구성은, 코딩된 비디오 화상 또는 그 일부분(예를 들어, 인터 및 인트라 화상, 인터 및 인트라 블록)의 유형 및 기타 요인에 따라, 여러 다른 유닛을 포함할 수 있다. 파서(520)에 의해, 코딩된 비디오 시퀀스로부터 파싱된 서브 그룹 제어 정보에 의해, 어떤 유닛이 관련될지, 그리고 어떻게 할지가 제어될 수 있다. 명확성을 위해, 파서(520)와 아래의 여러 유닛 사이의 이러한 서브 그룹 제어 정보의 흐름은 설명되지 않았다.

[0038] 이미 언급된 기능 블록을 넘어서, 비디오 디코더(510)는 아래에서 설명되는 바와 같이 개념적으로 다수의 기능 유닛으로 세분될 수 있다. 상업적 제약 하에서 동작하는 실제 일 구현에서, 이들 유닛들 중 다수는 서로 밀접하게 상호 작용하며, 적어도 부분적으로는 서로 통합될 수 있다. 그러나, 개시된 대상을 설명하기 위해, 아래와 같이 기능 단위로 개념적으로 세분화하는 것이 적절하다.

[0039] 제1 유닛은 스케일러/역 변환 유닛(551)이다. 스케일러/역 변환 유닛(551)은 파서(520)로부터, 양자화된 변환 계수뿐 아니라, 사용할 변환, 블록 크기, 양자화 인자, 양자화 스케일링 행렬 등을 포함하는 제어 정보를 심볼(들)(521)로서 수신한다. 스케일러/역 변환 유닛(551)은, 집적기(aggregator)(555)에 입력될 수 있는, 샘플 값들을 포함하는 블록들을 출력할 수 있다.

[0040] 일부 경우에, 스케일러/역 변환(551)의 출력 샘플은 인트라 코딩된 블록에 속할 수 있으며; 즉: 이전에 재구성된 화상의 예측 정보를 사용하지는 않지만 현재 화상에 대해 이전에 재구성된 부분의 예측 정보를 사용할 수 있는 블록이다. 이러한 예측 정보는 인트라 화상 예측 유닛(552)에 의해 제공될 수 있다. 일부 경우에, 인트라 화상 예측 유닛(552)은, 현재 화상 버퍼(558)로부터 페치(fetch)되는 이미 재구성된 주변 정보를 이용하여, 재구성 중인 블록과 동일한 크기 및 모양의 블록을 생성한다. 현재 화상 버퍼(558)는, 예를 들어, 부분적으로 재구성된 현재 화상 및/또는 완전히 재구성된 현재 화상을 버퍼링한다. 집적기(555)는, 일부 경우에, 인트라 예측 유닛(552)이 생성한 예측 정보를, 샘플 단위로, 스케일러/역 변환 유닛(551)에 의해 제공되는 출력 샘플 정보에 추가한다.

[0041] 다른 경우에, 스케일러/역 변환 유닛(551)의 출력 샘플은, 인터 코딩되고 잠재적으로 모션 보상된 블록에 속할

수 있다. 이러한 경우, 모션 보상 예측 유닛(553)은, 예측에 사용되는 샘플을 폐치하기 위해 참조 화상 메모리(557)에 액세스할 수 있다. 블록과 관련된 심볼(521)에 따라 폐치된 샘플을 모션 보상한 후, 이를 샘플들은 접적기(555)에 의해 스케일러/역 변환 유닛(551)의 출력에 추가되어(이 경우 잔차 샘플 또는 잔차 신호라고 함) 출력 샘플 정보를 생성할 수 있다. 모션 보상 예측 유닛(553)이 예측 샘플을 폐치하는 참조 화상 메모리(557) 내의 주소는 모션 벡터에 의해 제어될 수 있으며, 모션 보상 예측 유닛(553)에 대해, 예를 들어 X, Y 및 참조 화상 구성 요소를 가질 수 있는 심볼(521)의 형태로 이용 가능하다. 모션 보상은 또한, 서브 샘플 일치 모션 벡터(sub-sample exact motion vector)가 사용 중인 경우, 참조 화상 메모리(557)로부터 폐치된 샘플 값들에 대한 보간, 모션 벡터 예측 메커니즘 등을 포함할 수 있다.

[0042] 접적기(555)의 출력 샘플은 루프 필터 유닛(556)의 다양한 루프 필터링 기술의 대상일 수 있다. 비디오 압축 기술은, 코딩된 비디오 시퀀스(코딩된 비디오 비트스트림이라고도 함)에 포함된 파라미터에 의해 제어되고, 파서(520)로부터 심볼(521)로서 루프 필터 유닛(556)에 이용 가능하게 되는 인 루프 필터(in-loop filter) 기술을 포함할 수 있지만, 코딩된 화상 또는 코딩된 비디오 시퀀스의 (디코딩 순서에서) 이전 부분을 디코딩하는 동안 획득된 메타 정보에 응답할 수 있을뿐만 아니라, 이전에 재구성되고 루프 필터링된 샘플 값에 응답할 수도 있다.

[0043] 루프 필터 유닛(556)의 출력은 렌더링 디바이스(512)로 출력될 수 있을뿐만 아니라, 향후 인터 화상 예측에 사용하기 위해 참조 화상 메모리(557)에 저장될 수도 있는 샘플 스트림일 수 있다.

[0044] 일단 완전히 재구성된, 특정 코딩된 화상은 향후 예측을 위한 참조 화상으로 사용될 수 있다. 예를 들어, 일단 현재 화상에 대응하는 코딩된 화상이 완전히 재구성되고 코딩된 화상이 (예를 들어, 파서(520)에 의해) 참조 화상으로 식별되면, 현재 화상 버퍼(558)는 참조 화상 메모리(557)의 일부가 될 수 있고, 새로운 현재 화상 버퍼는 이어지는 코딩된 화상의 재구성을 시작하기 전에 재할당될 수 있다.

[0045] 비디오 디코더(510)는 ITU-T Rec. H.265와 같은 표준에서 미리 정해진 비디오 압축 기술에 따라 디코딩 동작을 수행할 수 있다. 코딩된 비디오 시퀀스는, 코딩된 비디오 시퀀스가 비디오 압축 기술 또는 표준의 구문론(syntax)과, 비디오 압축 기술 또는 표준에서 문서로서의 프로필(profile) 모두를 준수한다는 점에서, 사용되는 비디오 압축 기술 또는 표준에 의해 명세되는 구문론을 따를 수 있다. 특히, 프로필은, 비디오 압축 기술 또는 표준에서 사용할 수 있는 모든 도구에서 해당 프로필에서 사용할 수 있는 유일한 도구로 특정 도구를 선택할 수 있다. 또한, 준수를 위해 필요한 것은, 코딩된 비디오 시퀀스의 복잡도가 비디오 압축 기술 또는 표준의 레벨에 정의된 범위 내에 있어야 한다는 것이다. 일부 경우에, 레벨은 최대 화상 크기, 최대 프레임 레이트, 최대 재구성 샘플 레이트(예를 들어, 초당 평균 샘플로 측정됨), 최대 참조 화상 크기 등을 제한한다. 레벨 별로 설정된 제한은, 일부 경우에, 가상 참조 디코더(Hypothetical Reference Decoder, HRD) 사양 및 코딩된 비디오 시퀀스에서 시그널링되는 HRD 버퍼 관리를 위한 메타 데이터를 통해 추가로 제한될 수 있다.

[0046] 일 실시 예에서, 수신기(531)는, 인코딩된 비디오와 함께 추가(중복) 데이터를 수신할 수 있다. 추가 데이터는 코딩된 비디오 시퀀스(들)의 일부로서 포함될 수 있다. 추가 데이터는, 데이터를 적절하게 디코딩하고 및/또는 원본 비디오 데이터를 더욱 정확하게 재구성하기 위해 비디오 디코더(510)에 의해 사용될 수 있다. 추가 데이터는, 예를 들어 시간적, 공간적 또는 신호 잡음비(signal noise ratio, SNR) 향상 레이어, 중복 슬라이스, 중복 화상, 순방향 오류 정정 코드 등의 형태일 수 있다.

[0047] 도 6은 본 개시의 일 실시 예에 따른 비디오 인코더(603)의 블록도를 도시한다. 비디오 인코더(603)는 전자 디바이스(620)에 포함된다. 전자 디바이스(620)는 전송기(640)(예를 들어, 전송 회로)를 포함한다. 비디오 인코더(603)는 도 4 예의 비디오 인코더(403) 대신에 사용될 수 있다.

[0048] 비디오 인코더(603)는, 비디오 인코더(603)에 의해 코딩될 비디오 이미지(들)를 캡처할 수 있는 비디오 소스(601)(도 6 예에서 전자 디바이스(620)의 일부가 아님)로부터 비디오 샘플을 수신할 수 있다. 다른 예에서, 비디오 소스(601)는 전자 디바이스(620)의 일부이다.

[0049] 비디오 소스(601)는, 임의의 적합한 비트 심도(예를 들어, 8 비트, 10 비트, 12 비트, ...), 임의의 색 공간(예를 들어, BT.601 Y CrCB, RGB, ...) 및 임의의 적합한 샘플링 구조(예를 들어, Y CrCb 4:2:0, Y CrCb 4:4:4)일 수 있는 디지털 비디오 샘플 스트림의 형태로 비디오 인코더(603)에 의해 코딩될 소스 비디오 시퀀스를 제공할 수 있다. 미디어 서빙 시스템에서, 비디오 소스(601)는 이전에 준비된 비디오를 저장하는 저장 디바이스일 수 있다. 화상 회의 시스템에서, 비디오 소스(601)는, 로컬 이미지 정보를 비디오 시퀀스로 캡처하는 카메라일 수 있다. 비디오 데이터는, 순차적으로 볼 때, 모션을 부여하는 복수의 개별 화상으로 제공될 수 있다. 화상 자체

는 픽셀의 공간적 어레이로 구성될 수 있으며, 각 픽셀은, 사용중인 샘플링 구조, 색 공간 등에 따라 하나 이상의 샘플을 포함할 수 있다. 본 발명의 속하는 기술 분야에서 통상의 지식을 가진 자는 픽셀과 샘플 사이의 관계를 쉽게 이해할 수 있다. 아래 설명은 샘플에 중점을 두고 있다.

[0050] 일 실시 예에 따르면, 비디오 인코더(603)는 소스 비디오 시퀀스의 화상을, 실시간으로 또는 애플리케이션에 의해 요구되는 임의의 다른 시간 제약 조건 하에 코딩된 비디오 시퀀스(643)로 코딩 및 압축할 수 있다. 적절한 코딩 속도를 적용하는 것은 컨트롤러(650)의 기능 중 하나이다. 일부 실시 예들에서, 제어기(650)는 후술되는 바와 같이 다른 기능 유닛들을 제어하고, 다른 기능 유닛들에 기능적으로 결합된다. 명확성을 위해 커플 링은 표시되지 않았다. 컨트롤러(650)에 의해 설정되는 파라미터는, 레이트 제어 관련 파라미터(화상 스kip, 양자화기, 레이트 왜곡 최적화 기술의 람다 값 등), 화상 크기, 화상 그룹(GOP) 레이아웃, 최대 모션 벡터 검색 범위 등을 포함할 수 있다. 제어기(650)는, 특정 시스템 설계에 최적화된 비디오 인코더(603)에 속하는 다른 적절한 기능을 갖도록 구성될 수 있다.

[0051] 일부 실시 예들에서, 비디오 인코더(603)는 코딩 루프에서 동작하도록 구성된다. 지난치게 단순화된 설명으로서, 일 예에서, 코딩 루프는 소스 코더(630)(예를 들어, 코딩될 입력 화상, 참조 화상(들)에 기초하여, 심볼 스트림과 같은 심볼 생성을 담당함) 및 비디오 인코더(603)에 내장 된 (로컬) 디코더(633)를 포함할 수 있다. 디코더(633)는 심볼을 재구성하여 (원격) 디코더가 생성하는 것과 유사한 방식으로 샘플 데이터를 생성한다 (심볼과 코딩된 비디오 비트스트림 사이의 모든 압축은 개시된 대상에서 고려되는 비디오 압축 기술에서 무손실이기 때문임). 재구성된 샘플 스트림(샘플 데이터)은 참조 화상 메모리(634)에 입력된다. 심볼 스트림의 디코딩은 디코더 위치(로컬 또는 원격)와 무관한 비트 일치(bit-exact) 결과로 이어지므로, 참조 화상 메모리(634)의 콘텐츠는 로컬 인코더와 원격 인코더 사이에서도 비트 일치한다. 즉, 인코더의 예측 부분은, 디코딩 중에 예측을 사용할 때 디코더가 "보는" 것과 정확히 동일한 샘플 값을 참조 화상 샘플로 "인식"한다. 참조 화상 동기성(reference picture synchronicity)의 이러한 기본 원칙(예를 들면, 채널 오류 등으로 동기성을 유지할 수 없는 경우, 드리프트를 야기함)은 일부 관련 기술에서도 또한 사용된다.

[0052] "로컬" 디코더(633)의 동작은 비디오 디코더(510)와 같은 "원격" 디코더의 동작과 동일할 수 있으며, 이는 도 5와 관련하여 위에서 상세하게 설명되었다. 도 5를 또한 간략하게 참조하면, 그러나, 심볼이 이용 가능하고 엔트로피 코더(645) 및 파서(520)에 의해, 심볼을 코딩된 비디오 시퀀스로 인코딩/디코딩하는 것은 무손실일 수 있으므로, 베퍼 메모리(515) 및 파서(520)를 포함하는, 비디오 디코더(510)의 엔트로피 디코딩 부분은, 로컬 디코더(633)에서 완전히 구현되지 않을 수 있다.

[0053] 이 시점에서 관찰할 수 있는 것은, 디코더에 존재하는 파싱/엔트로피 디코딩을 제외한 모든 디코더 기술이 반드시 상응하는 인코더에 실질적으로 동일한 기능적 형태로 존재해야 한다는 것이다. 이러한 이유로, 개시된 대상은 디코더 동작에 초점을 맞춘다. 인코더 기술에 대한 설명은, 포괄적으로 설명된 디코더 기술의 반대이므로 줄일 수 있다. 특정 영역에서만 더 자세한 설명이 필요하며 아래에서 제공된다.

[0054] 동작 중, 일부 예들에서, 소스 코더(630)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(632)은, 입력 화상에 대한 예측 참조(들)로서 선택될 수 있는 입력 화상의 픽셀 블록과 참조 화상(들)의 픽셀 블록 간의 차이를 코딩한다.

[0055] 로컬 비디오 디코더(633)는, 소스 코더(630)에 의해 생성된 심볼에 기초하여, 참조 화상으로 지정될 수 있는 화상의 코딩된 비디오 데이터를 디코딩할 수 있다. 코딩 엔진(632)의 동작은 유리하게 손실 프로세스일 수 있다. 코딩된 비디오 데이터가 비디오 디코더(도 6에 도시되지 않음)에서 디코딩될 수 있을 때, 재구성된 비디오 시퀀스는 일반적으로 약간의 오류가 있는 소스 비디오 시퀀스의 복제본일 수 있다. 로컬 비디오 디코더(633)는 참조 화상에 대해 비디오 디코더에 의해 수행될 수 있는 디코딩 프로세스를 복제하고, 재구성된 참조 화상이 참조 화상 캐시(634)에 저장되도록 할 수 있다. 이러한 방식으로, 비디오 인코더(603)는 원단(far-end) 비디오 디코더(전송 오류 없음)에 의해 획득될 재구성된 참조 화상으로서 공통 콘텐츠를 갖는 재구성된 참조 화상의 사본을 로컬로(locally) 저장할 수 있다.

[0056] 예측자(635)는 코딩 엔진(632)에 대한 예측 검색을 수행할 수 있다. 즉, 코딩될 새로운 화상에 대해, 예측자(635)는 참조 화상 메모리(634)에서 (후보 참조 픽셀 블록으로서의) 샘플 데이터 또는 참조 화상 모션 벡터, 블록 모양 등과 같은 특정 메타 데이터를 검색할 수 있으며, 이는 새로운 화상에 대한 적절한 예측 참조 역할을 할 수 있다. 예측자(635)는 적절한 예측 참조를 찾기 위해 샘플 블록 별 픽셀 블록(sample block-by-pixel block) 단위로 동작할 수 있다. 일부 경우에, 예측자(635)에 의해 획득되는 검색 결과에 의해 결정되는 바와 같

이, 입력 화상은, 참조 화상 메모리(634)에 저장된 다수의 참조 화상으로부터 추출되는 예측 참조를 가질 수 있다.

[0057] 제어기(650)는, 예를 들어, 비디오 데이터를 인코딩하기 위해 사용되는 파라미터 및 서브 그룹 파라미터에 대한 설정을 포함하는 소스 코더(630)의 코딩 동작을 관리할 수 있다.

[0058] 전술한 모든 기능 유닛의 출력은 엔트로피 코더(645)의 엔트로피 코딩의 대상일 수 있다. 엔트로피 코더(645)는 예를 들어 허프만 코딩, 가변 길이 코딩, 산술 코딩 등과 같은, 해당 기술 분야에서 통상의 지식을 가진 자에게 알려진 기술에 따라 심볼을 무손실 압축함으로써, 다양한 기능 유닛에 의해 생성된 심볼을 코딩된 비디오 시퀀스로 변환한다.

[0059] 전송기(640)는 엔트로피 코더(645)에 의해 생성된, 코딩된 비디오 시퀀스(들)를 버퍼링하여, 인코딩된 비디오 데이터를 저장할 수 있는 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는 통신 채널(660)을 통한 전송을 준비할 수 있다. 전송기(640)는 비디오 코더(603)로부터의 코딩된 비디오 데이터를 전송될 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림(소스는 도시되지 않음)과 병합할 수 있다.

[0060] 제어부(650)는 비디오 인코더(603)의 동작을 관리할 수 있다. 코딩 중, 제어기(650)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.

[0061] 인트라 화상(Intra Picture)(I 화상(I picture))은, 예측 소스로서 시퀀스의 다른 어떤 화상도 사용하지 않고 코딩 및 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱은, 예를 들어 독립적인 디코더 리프레시(Independent Decoder Refresh)("IDR") 화상을 포함하여 다양한 유형의 인트라 화상을 허용한다. 본 발명의 속하는 기술 분야에서 통상의 지식을 가진 자는, I 화상의 이러한 변형 및 각각의 응용 및 특징을 알 수 있다.

[0062] 예측 화상(P 화상(P picture))은, 각 블록의 샘플 값을 예측하기 위해 최대 하나의 모션 벡터 및 참조 인덱스를 사용하는 인터 예측 또는 인트라 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다.

[0063] 양방향 예측 화상(B 화상(B picture))은, 각 블록의 샘플 값을 예측하기 위해 최대 2 개의 모션 벡터 및 참조 인덱스를 사용하는 인터 예측 또는 인트라 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다. 이와 유사하게, 다중 예측 화상은, 단일 블록의 재구성을 위해 3 개 이상의 참조 화상 및 연관된 메타 데이터를 사용할 수 있다.

[0064] 소스 화상은 일반적으로, 복수의 샘플 블록(예를 들어, 각각 4x4, 8x8, 4x8 또는 16x16 샘플 블록)으로 공간적으로 세분화되고, 블록 단위로 코딩될 수 있다. 블록은, 블록의 각각의 화상에 적용되는 코딩 할당에 의해 결정되는 바와 같이 다른 (이미 코딩된) 블록을 참조하여 예측 적으로 코딩될 수 있다. 예를 들어, I 화상의 블록은 비 예측적으로(non-predictively) 코딩될 수 있거나, 이들은 동일한 화상에 대해 이미 코딩된 블록을 참조하여 예측 적으로(predictively) 코딩될 수 있다(공간적 예측 또는 인트라 예측). P 화상의 픽셀 블록은, 이전에 코딩된 하나의 참조 화상을 참조하여, 공간적 예측을 통해 또는 시간적 예측을 통해 예측 적으로 코딩될 수 있다. B 화상의 블록은, 이전에 코딩된 하나 또는 2 개의 참조 화상을 참조하여, 공간적 예측을 통해 또는 시간적 예측을 통해 예측 적으로 코딩될 수 있다.

[0065] 비디오 인코더(603)는 ITU-T Rec. H.265와 같은 미리 정해진 비디오 코딩 기술 또는 표준에 따라 코딩 동작을 수행할 수 있다. 해당 동작에서, 비디오 인코더(603)는, 입력 비디오 시퀀스에서 시간적 및 공간적 중복성을 이용하는 예측 코딩 동작을 포함하는 다양한 압축 동작을 수행할 수 있다. 따라서, 코딩된 비디오 데이터는, 사용되는 비디오 코딩 기술 또는 표준에 의해 지정되는 구문론을 따를 수 있다.

[0066] 일 실시 예에서, 전송기(640)는 인코딩된 비디오와 함께 추가 데이터를 전송할 수 있다. 소스 코더(630)는 코딩된 비디오 시퀀스의 일부로서 이러한 데이터를 포함할 수 있다. 추가 데이터는, 시간적/공간적/SNR 항상 계층화, 중복 화상 및 슬라이스, SEI(Supplementary Enhancement Information) 메시지, VUI(Visual Usability Information) 파라미터 세트 프로그먼트 등과 같은 다른 형태의 중복 데이터를 포함할 수 있다.

[0067] 비디오는 시간적 순서로 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 화상들 간의(시간적 또는 기타) 상관 관계를 사용한다. 일 예에서, 현재 화상이라고 지칭되는 인코딩/디코딩 중인 특정 화상은 블록으로 파티션 분할된다. 현재 화상의 블록이, 비디오에서 이전에 코딩되고 여전히 버퍼링된 참조 화상의 참조 블록과 유사한 경우, 현재 화상의 블록은 모션 벡터라고 지칭되는 벡터로 코딩될 수 있다. 모션 벡터는 참조 화상의 참

조 블록을 가리키며, 여러 참조 화상이 사용 중인 경우 참조 화상을 식별하는 3 차원을 가질 수 있다.

[0068] 일부 실시 예에서, 이중 예측(bi-prediction) 기술이 인터 화상 예측에 사용될 수 있다. 이중 예측 기술에 따르면, 비디오의 현재 화상에 대해 디코딩 순서가 모두 선행하는(그러나 표시 순서에서 각각 과거 및 미래에 있을 수 있는) 제1 및 제2 참조 화상과 같은 2 개의 참조 화상이 사용된다. 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 블록 및 제2 참조 블록의 조합에 의해 예측될 수 있다.

[0069] 또한, 인터 화상 예측에 병합 모드 기술을 사용하여 코딩 효율을 높일 수 있다.

[0070] 본 개시의 일부 실시 예들에 따르면, 인터 화상 예측 및 인트라 화상 예측과 같은 예측은 블록 단위로 수행된다. 예를 들어, HEVC 표준에 따라, 비디오 화상 시퀀스의 일 화상은 압축을 위해 코딩 트리 유닛(coding tree unit, CTU)으로 패티션 분할되고, 화상의 CTU는, 64x64 픽셀, 32x32 픽셀 또는 16x16 픽셀과 같이 동일한 크기를 갖는다. 일반적으로, CTU에는 3 개의 코딩 트리 블록(CTB)이 포함되며, 이는 1 개의 루마(luma) CTB 및 2 개의 크로마(chroma) CTB이다. 각 CTU는 재귀적으로 쿼드 트리(quadtreen)를 하나 또는 여러 코딩 유닛(coding unit, CU)으로 분할할 수 있다. 예를 들어, 64x64 픽셀의 CTU는 64x64 픽셀의 CU 1 개 또는 32x32 픽셀의 CU 4 개 또는 16x16 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) 및 2 개의 크로마 PB를 포함한다. 일 실시 예에서, 코딩에서의 예측 동작(인코딩/디코딩)은 예측 블록 단위로 수행된다. 예측 블록의 일 예로서 루마 예측 블록을 사용하여, 예측 블록은, 8x8 픽셀, 16x16 픽셀, 8x16 픽셀, 16x8 픽셀 등과 같은 픽셀에 대한 값들(예를 들어, 루마 값들)의 행렬을 포함한다.

[0071] 도 7은 본 개시의 다른 실시 예에 따른 비디오 인코더(703)를 설명하기 위한 도면이다. 비디오 인코더(703)는, 비디오 화상 시퀀스에서 현재 비디오 화상 내의 샘플 값의 프로세싱 블록(예를 들어, 예측 블록)을 수신하고, 프로세싱 블록을, 코딩된 비디오 시퀀스의 일부인 코딩된 화상으로 인코딩하도록 구성된다. 일 예에서, 비디오 인코더(703)는 도 4 예의 비디오 인코더(403) 대신에 사용된다. 4 예.

[0072] HEVC 예에서, 비디오 인코더(703)는 8x8 샘플의 예측 블록 등과 같은, 프로세싱 블록에 대한 샘플 값들의 행렬을 수신한다. 비디오 인코더(703)는, 프로세싱 블록이, 예를 들어 레이트 왜곡 최적화를 사용하여, 인트라 모드, 인터 모드 또는 이중 예측 모드를 사용하여 최상으로 코딩되는지(best coded) 여부를 결정한다. 프로세싱 블록이 인트라 모드에서 코딩되는 경우, 비디오 인코더(703)는, 프로세싱 블록을 코딩된 화상으로 인코딩하기 위해 인트라 예측 기술을 사용할 수 있고; 프로세싱 블록이 인터 모드 또는 이중 예측 모드에서 코딩되는 경우, 비디오 인코더(703)는, 프로세싱 블록을 코딩된 화상으로 인코딩하기 위해, 인터 예측 또는 이중 예측 기술을 각각 사용할 수 있다. 특정 비디오 코딩 기술에서, 병합 모드는, 예측자 외부의 코딩된 모션 벡터 구성 요소라는 이점 없이, 모션 벡터가 하나 이상의 모션 벡터 예측자로부터 도출되는 인터 화상 예측 하위 모드일 수 있다. 다른 특정 비디오 코딩 기술에서, 주제 블록(subject block)에 적용 가능한 모션 벡터 구성 요소가 존재 할 수 있다. 일 예에서, 비디오 인코더(703)는 프로세싱 블록의 모드를 결정하기 위한 모드 결정 모듈(도시되지 않음)과 같은 다른 구성 요소를 포함한다.

[0073] 도 7 예에서, 비디오 인코더(703)는, 도 7에 도시된 바와 같이 함께 결합된, 인터 인코더(730), 인트라 인코더(722), 잔차 계산기(723), 스위치(726), 잔차 인코더(724), 일반 제어기(721) 및 엔트로피 인코더(725)를 포함한다.

[0074] 인터 인코더(730)는 현재 블록(예를 들어, 프로세싱 블록)의 샘플을 수신하고, 블록을 참조 화상의 하나 이상의 참조 블록(예를 들어, 이전 화상 및 나중 화상의 블록)과 비교하고, 인터 예측 정보(예를 들어, 인터 인코딩 기술에 따른 중복 정보에 대한 설명, 모션 벡터, 병합 모드 정보)를 생성하고, 임의의 적합한 기술을 사용하여 인터 예측 정보를 기반으로 인터 예측 결과(예를 들어, 예측 블록)를 계산한다. 일부 예에서, 참조 화상은 인코딩된 비디오 정보에 기초하여 디코딩되는, 디코딩된 참조 화상이다.

[0075] 인트라 인코더(722)는 현재 블록(예를 들어, 프로세싱 블록)의 샘플을 수신하고, 일부 경우에 블록을, 동일한 화상에서 이미 코딩된 블록과 비교하고, 변환 후 양자화된 계수를 생성하고, 일부 경우에는 또한 인트라 예측 정보(예를 들어, 하나 이상의 인트라 인코딩 기술에 따른 인트라 예측 방향 정보)를 계산한다. 일 예에서, 인트라 인코더(722)는 또한, 동일한 화상의 인트라 예측 정보 및 참조 블록에 기초하여 인트라 예측 결과(예를 들어, 예측 블록)를 계산한다.

- [0076] 일반 제어기(721)는 일반 제어 데이터를 결정하고, 일반 제어 데이터에 기초하여 비디오 인코더(703)의 다른 구성 요소를 제어하도록 구성된다. 일 예에서, 일반 제어기(721)는 블록의 모드를 결정하고, 모드에 따라 스위치(726)에 제어 신호를 제공한다. 예를 들어, 모드가 인트라인 경우, 일반 제어기(721)는 스위치(726)를 제어하여, 잔차 계산기(723)에 의해 사용할 인트라 모드 결과를 선택하고, 엔트로피 인코더(725)를 제어하여 인트라 예측 정보를 선택하고 비트스트림에 인트라 예측 정보를 포함시키고; 모드가 인터 모드인 경우, 일반 제어기(721)는 스위치(726)를 제어하여, 잔차 계산기(723)에 의해 사용할 인터 예측 결과를 선택하고, 엔트로피 인코더(725)를 제어하여 인터 예측 정보를 선택하고 비트스트림에 인터 예측 정보를 포함시킨다.
- [0077] 잔차 계산기(723)는, 수신된 블록과 인트라 인코더(722) 또는 인터 인코더(730)로부터 선택된 예측 결과 사이의 차이(잔차 데이터)를 계산하도록 구성된다. 잔차 인코더(724)는 잔차 데이터에 기초하여 동작하고, 잔차 데이터를 인코딩하여 변환 계수를 생성하도록 구성된다. 일 예에서, 잔차 인코더(724)는 주파수 도메인에서 잔차 데이터를 변환하고, 변환 계수를 생성하도록 구성된다. 다음으로, 변환 계수는, 양자화된 변환 계수를 획득하기 위해 양자화 프로세싱이 된다. 다양한 실시 예들에서, 비디오 인코더(703)는 또한 잔차 디코더(728)를 포함한다. 잔차 디코더(728)는 역 변환을 수행하고, 디코딩된 잔차 데이터를 생성하도록 구성된다. 디코딩된 잔차 데이터는 인트라 인코더(722) 및 인터 인코더(730)에 의해 적절하게 사용될 수 있다. 예를 들어, 인터 인코더(730)는, 디코딩된 잔차 데이터 및 인터 예측 정보에 기초하여 디코딩된 블록을 생성할 수 있고, 인트라 인코더(722)는, 디코딩된 잔차 데이터 및 인트라 예측 정보에 기초하여 디코딩된 블록을 생성할 수 있다. 디코딩된 블록은 적절하게 처리되어 디코딩된 화상을 생성하고, 디코딩된 화상은 메모리 회로(도시되지 않음)에 버퍼링되어 일부 예들에서 참조 화상으로서 사용될 수 있다.
- [0078] 엔트로피 인코더(725)는 인코딩된 블록을 포함하도록 비트스트림을 포맷(format)하도록 구성된다. 엔트로피 인코더(725)는 HEVC 표준과 같은 적절한 표준에 따라 다양한 정보를 포함하도록 구성된다. 일 예에서, 엔트로피 인코더(725)는, 일반 제어 데이터, 선택된 예측 정보(예를 들어, 인트라 예측 정보 또는 인터 예측 정보), 잔차 정보 및 기타 적절한 정보를 비트스트림에 포함하도록 구성된다. 개시된 대상에 따르면, 인터 모드 또는 이중 예측 모드의 병합 하위 모드에서 블록을 코딩하는 경우, 잔차 정보가 존재하지 않음에 유의해야 한다.
- [0079] 도 8은 본 개시의 다른 실시 예에 따른 비디오 디코더(810)를 설명하기 위한 도면이다. 비디오 디코더(810)는, 코딩된 비디오 시퀀스의 일부인 코딩된 화상을 수신하고, 코딩된 화상을 디코딩하여 재구성된 화상을 생성하도록 구성된다. 일 예에서, 비디오 디코더(810)는 도 4의 비디오 디코더(410) 대신에 사용된다.
- [0080] 도 8에서, 비디오 디코더(810)는, 도 8에 도시된 바와 같이 함께 결합된, 엔트로피 디코더(871), 인터 디코더(880), 잔차 디코더(873), 재구성 모듈(874) 및 인트라 디코더(872)를 포함한다.
- [0081] 엔트로피 디코더(871)는, 코딩된 화상로부터, 코딩된 화상이 구성되는 구문론 요소를 나타내는 특정 심볼을 재구성하도록 구성될 수 있다. 이러한 심볼은, 예를 들어, (예를 들어, 인트라, 인터, b 예측(b-predicted) 모드, 병합 하위 모드 또는 다른 하위 모드에서 후자의 2 가지와 같은) 블록이 코딩되는 모드, 인트라 디코더(872) 또는 인터 디코더(880)에 의해 각각 예측에 사용되는 특정 샘플 또는 메타 데이터를 식별할 수 있는 (예를 들어, 인트라 예측 정보 또는 인터 예측 정보와 같은) 예측 정보, 예를 들어, 양자화된 변환 계수 형태의 잔차 정보 등을 포함할 수 있다. 일 예에서, 예측 모드가 인터 또는 이중 예측 모드인 경우, 인터 예측 정보는 인터 디코더(880)에 제공되고; 예측 유형이 인트라 예측 유형인 경우, 인트라 예측 정보는 인트라 디코더(872)에 제공된다. 잔차 정보는 역 양자화의 대상일 수 있고 잔차 디코더(873)에 제공된다.
- [0082] 인터 디코더(880)는 인터 예측 정보를 수신하고, 인터 예측 정보에 기초하여 인터 예측 결과를 생성하도록 구성된다.
- [0083] 인트라 디코더(872)는 인트라 예측 정보를 수신하고, 인트라 예측 정보에 기초하여 예측 결과를 생성하도록 구성된다.
- [0084] 잔차 디코더(873)는 역 양자화를 수행하여 탈 양자화된(de-quantized) 변환 계수를 추출하고, 탈 양자화된 변환 계수를 처리하여 주파수 도메인에서 공간적 도메인으로 잔차를 변환하도록 구성된다. 잔차 디코더(873)는 또한 특정 제어 정보(양자화기 파라미터(Quantizer Parameter, QP) 포함)를 필요로 할 수 있으며, 해당 정보는 엔트로피 디코더(871)에 의해 제공될 수 있다(표시되지 않은 데이터 경로는 낮은 볼륨 제어 정보(low volume control information) 뿐일 수 있음).
- [0085] 재구성 모듈(874)은, 공간적 도메인에서, 잔차 디코더(873)에 의해 출력되는 잔차 및 예측 결과(경우에 따라 인터 또는 인트라 예측 모듈에 의해 출력됨)를 결합하여, 재구성된 화상의 일부일 수 있는 재구성된 블록을 형성

하고, 이는 결국 재구성된 비디오의 일부일 수 있다. 디 블로킹(deblocking) 동작 등과 같은 다른 적절한 동작이 시각적 품질을 개선하기 위해 수행될 수 있다는 점을 유의해야 한다.

[0086] 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 임의의 적절한 기술을 사용하여 구현될 수 있다는 점에 유의해야 한다. 일 실시 예에서, 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 하나 이상의 집적 회로를 사용하여 구현될 수 있다. 다른 실시 예에서, 비디오 인코더(403, 603, 603) 및 비디오 디코더(410, 510, 810)는 소프트웨어 명령을 실행하는 하나 이상의 프로세서를 사용하여 구현될 수 있다.

[0087] 본 개시의 측면들은 인트라 화상 블록 보상을 위한 기술들을 제공한다.

[0088] 다른 화상으로부터의 블록 기반 보상을 모션 보상이라고 한다. 이와 유사하게, 블록 보상은 동일한 화상 내에서 이전에 재구성된 영역에서도 수행될 수 있다. 동일한 화상 내의 재구성된 영역으로부터의 블록 기반 보상을, 인트라 화상 블록 보상 또는 인트라 블록 카피라고 한다. 동일한 화상에서 현재 블록과 참조 블록 사이의 오프셋을 나타내는 변위 벡터를 블록 벡터(또는 줄여서 BV)라고 한다. 임의의 값(x 또는 y 방향에서 양수 또는 음수)이 될 수 있는, 모션 보상의 모션 벡터와 달리, 블록 벡터에는 참조 블록이 사용 가능하고 이미 재구성되었는지 확인하기 위한 몇 가지 제약 조건이 있다. 또한, 일부 예들에서, 병렬 처리 고려를 위해, 타일 경계(tile boundary) 또는 과면 사다리 모양 경계(wavefront ladder shape boundary)인 일부 참조 영역이 제외된다.

[0089] 블록 벡터의 코딩은 명시적 또는 암시적일 수 있다. 명시적 모드에서, 블록 벡터와 그 예측자 사이의 차이가 시그널링되고; 암시적 모드에서, 블록 벡터는, 병합 모드의 모션 벡터와 유사한 방식으로 예측자(블록 벡터 예측자라고 함)로부터 복원된다. 일부 구현들에서, 블록 벡터의 해상도는 정수 위치로 제한되고; 다른 시스템에서는, 블록 벡터가 소수 위치(fractional position)를 가리킬 수 있다.

[0090] 일부 예들에서, 블록 레벨에서의 인트라 블록 카피의 사용은, 참조 인덱스 접근법을 사용하여 시그널링될 수 있다. 디코딩 중인 현재 화상은 참조 화상으로 취급된다. 일 예에서, 이러한 참조 화상은 참조 화상 목록의 마지막 위치에 배치된다. 이러한 특수한 참조 화상은 또한 디코딩된 화상 버퍼(decoded picture buffer, DPB)와 같은 버퍼의 다른 시간적 참조 화상과 함께 관리된다.

[0091] 또한, 플립된(flipped) 인트라 블록 카피(참조 블록은 현재 블록을 예측하기 위해 사용되기 전에 수평 또는 수직으로 플립됨) 또는 라인 기반 인트라 블록 카피(MxN 코딩 블록 내의 각 보상 유닛이 Mx1 또는 1xN 라인임)와 같은 인트라 블록 카피에 대한 몇 가지 변형도 존재한다.

[0092] 도 9는 본 개시의 일 실시 예에 따른 인트라 블록 카피의 일 예를 도시한다. 현재 화상(900)이 디코딩 중이다. 현재 화상(900)은 재구성된 영역(910)(회색 영역) 및 디코딩될 영역(920)(흰색 영역)을 포함한다. 현재 블록(930)은 디코더에 의해 재구성 중이다. 현재 블록(930)은, 재구성된 영역(910)에 존재하는 참조 블록(940)으로부터 재구성될 수 있다. 참조 블록(940)과 현재 블록(930) 사이의 위치 오프셋을 블록 벡터(950)(또는 BV(950))라고 한다.

[0093] 전통적으로, 모션 벡터 해상도는, 예를 들어 H.264/AVC 및 HEVC 메인 프로필의 1/4-pel(픽셀) 정확도 또는 1/8-pel 정확도 등에서 고정 값이다. HEVC SCC에서, 모션 벡터의 해상도는 1-integer-pel 또는 1/4-pel에서 선택될 수 있다. 각 슬라이스에서 스위칭이 발생한다. 즉, 슬라이스에 있는 모든 모션 벡터의 해상도는 동일할 것이다.

[0094] 일부 이후 개발에서, 모션 벡터의 해상도는 1/4-pel, 1-integer-pel 또는 4-integer-pel에 있을 수 있다. 4-integer-pel의 예에서, 각 단위는 4 개의 정수 픽셀을 나타낸다. 따라서, 심볼 "0"에서 "1" 사이에는 4 개의 정수 픽셀 거리가 존재한다. 또한, 적응성(adaptivity)은 블록 레벨에서 발생하며 – 모션 벡터는 블록 별로 다른 해상도를 선택할 수 있다.

[0095] 본 개시의 측면들은, 이미지 및 비디오 압축에서 적응형 모션 벡터 해상도 및 블록 벡터 해상도를 위한 방법을 제공한다.

[0096] 블록 벡터는 일반적으로 정수 해상도로 시그널링된다. 인트라 블록 카피가 전체 프레임 범위로 확장되어, 현재 디코딩된 화상의 모든 재구성된 영역이 참조로서 사용될 수 있을 때, 멀리 떨어진 참조에 대한 코딩 블록 벡터 차이의 비용이 높아진다. 적응형 블록 벡터 차이 해상도는, 블록 벡터 차이의 코딩을 개선하기 위해 사용될 수 있다.

[0097] 일부 예에서, 다중 블록 벡터 해상도는 블록 벡터의 코딩을 위해 사용되며, 블록 벡터 해상도는 블록 레벨에서 스위칭 가능하다. 또한, 3 개 이상의 가능한 해상도가 사용될 때 1-bin 보다 많이 포함할 수 있는 시그널링 플래그가 사용되어, 다중 블록 벡터 해상도 중 어느 것이 현재 블록의 블록 벡터 차이에 대해 사용되는지 알려준

다. 가능한 해상도는 1/8-pel, 1/4-pel, 1/2-pel, 1-integer-pel, 2-integer-pel, 4-integer-pel, 8-integer-pel 등을 포함하지만 이에 한정되지 않는다. X-integer-pel은 심볼의 최소 단위가 X 개의 정수 위치들을 나타냄을 의미한다. 예를 들어, 2-integer-pel은 심볼의 각 최소 단위가 2 개의 정수 위치들을 나타냄을 의미하고, 4-integer-pel은 심볼의 각 최소 단위가 4 개의 정수 위치를 나타냄을 의미하고, 8-integer-pel은 심볼의 각 최소 단위가 8 개의 정수 위치를 나타냄을 의미한다.

[0098] 일 실시 예에서, 해상도 세트가 사용될 수 있고, 해상도 세트 중 어느 것이 사용되는지를 표시하기 위해 플래그가 시그널링될 수 있다. 일례로, 블록 벡터 차이 해상도 세트는 1-integer-pel 및 2-integer-pel과 같은 2 가지 해상도를 포함한다. 그런 다음, 2 개의 가능한 해상도 중에서 하나의 해상도를 선택하기 위해 1-bin(1 바이너리) 플래그가 시그널링된다. 다른 예에서, 블록 벡터 해상도 세트는 1-integer-pel 및 4-integer-pel을 포함한다. 그런 다음, 가능한 2 가지 해상도 중 하나를 선택하도록 1-bin 플래그가 신그널링된다.

[0099] 다른 예에서, 블록 벡터 차이 해상도 세트는 1-integer-pel, 2-integer pel 및 4-integer-pel 을 포함한다. 1-bin 플래그는 1-integer-pel 해상도가 사용되는지 여부를 표시하기 위해 시그널링된다. 1-integer-pel이 사용되지 않으면, 4-integer-pel 해상도를 선택할지 여부를 표시하기 위해 다른 1-bin 플래그가 시그널링된다. 3 가지 가능성으로부터의 해상도 선택을 표시하기 위해, 1-bin 또는 2-bin을 사용하는 유사한 방식으로, 상이한 이진화 실시 예들이 유도될 수 있다.

[0100] 다른 예에서, 블록 벡터 차이 해상도 세트는 1-integer-pel, 4-integer pel 및 8-integer-pel 을 포함한다. 1-bin 플래그는 1-integer-pel 해상도가 사용되는지 여부를 표시하기 위해 시그널링된다. 1-integer-pel이 사용되지 않으면, 4-integer-pel 해상도를 선택할지 여부를 표시하기 위해 다른 1-bin 플래그가 시그널링된다. 3 가지 가능성으로부터의 해상도 선택을 표시하기 위해, 1-bin 또는 2-bin을 사용하는 유사한 방식으로, 상이한 이진화 실시 예들이 유도될 수 있다.

[0101] 다른 예에서, 블록 벡터 차이의 x 및 y 성분들은 서로 다른 해상도를 사용할 수 있다. 예를 들어, 블록 벡터 차이의 x 성분이 1-integer-pel 해상도에 있고, y 성분은 4-integer-pel 해상도에 있다. 각 성분에 대한 선택은, 명시적 시그널링(성분 당 1 개의 플래그) 또는 추론을 통해 이루어질 수 있다. 예를 들어, 현재 블록의 블록 벡터 예측자의 크기는, 해상도를 추론하기 위해 사용될 수 있다. 일례로, 블록 벡터 예측자의 성분의 크기가 문턱값보다 큰 경우, 해당 성분에 대한 블록 벡터 차이 해상도는 더 큰 단계 해상도, 예를 들어 4-integer-pel 해상도를 사용할 것이며; 그렇지 않으면, 해당 성분은 더 작은 단계 해상도, 예를 들어 1-integer-pel 해상도를 사용할 것이다.

[0102] 다른 예에서, 블록 벡터 차이의 x 및 y 성분들은 서로 다른 해상도를 사용할 수 있다. 디폴트로, 성분들 모두, 1-integer-pel 해상도와 같은 고정 해상도를 사용할 것이다. 성분 별 조건은, 해상도를 결정하기 위해 성분들에 대해 각각 설정된다. 예를 들어, 일부 성분 별 조건이 충족되면, 각 성분은 추론을 통해 다른 해상도로 스위칭 할 수 있다. 예측자의 성분은, 또한 대응하는 해상도로 양자화되거나, 원본 해상도를 변경하지 않고 그대로 유지한다. 예를 들어, 하나의 성분 별 조건은, 현재 블록의 블록 벡터 예측자의 크기와 관련이 있다. 각 성분에서 현재 블록의 블록 벡터 예측자의 크기를 평가하여, 성분에 대한 해상도를 추론할 수 있다. 예를 들어, 블록 벡터 예측자의 성분의 크기가 문턱값보다 큰 경우, 해당 성분에 대한 블록 벡터 차이 해상도는 더 큰 단계 해상도, 예를 들어 4-integer-pel 해상도를 사용하며; 그렇지 않으면 해당 성분은 디폴트 단계 해상도, 예를 들어 1-integer-pel 해상도를 사용할 것이다. 특정 예에서, 디폴트 해상도는 1-integer-pel이다. 블록 벡터 예측자는 (-21, -3)으로 설정되고, 4-integer-pel 해상도를 사용하는 문턱값은 각 성분에 대해 20이다. 디코딩된 블록 벡터 차이 심볼은 (2, 2)이다. 규칙에 따라, x 성분은 4-integer-pel 해상도를 사용하고, y 성분은 1-integer-pel 해상도를 사용할 것이다. 그러면 예측자는 (-20, -3)(-21은 -20으로 반올림 됨)이되고, (x 성분에 대한 4-integer-pel 및 y 성분에 대한 1-integer-pel로 인해) 블록 벡터 차이는 (8, 2)가 될 것이다. 본 예에서, 최종적으로 디코딩되는 블록 벡터는 (-12, -1)이다.

[0103] 위의 예에서의 유사한 전개를 사용하여 유사한 예를 유도할 수 있다. 일 실시 예에서, 가능한 해상도의 세트가 형성되고, 블록 별 해상도의 선택을 표시하기 위해 시그널링 플래그(들)가 사용된다. 또한, 위의 예에서, 시그널링 플래그의 빈(bin)은 컨텍스트 코딩되거나, 바이패스 코딩될 수 있다. 컨텍스트 코딩을 사용하는 경우, 현재 블록의 공간적 이웃의 블록 벡터 해상도를 컨텍스트 모델링에 사용할 수 있다. 또는, 마지막 코딩된 블록 벡터의 해상도를 사용할 수 있다.

[0104] 본 개시의 다른 측면에 따르면, 블록 벡터 예측자는 통상적으로, 현재 블록에 대한 이웃 블록의 하나의 이전에 코딩된 블록 벡터로부터 유도된다. 예측자로 사용될 때, 이전에 디코딩된 블록 벡터는, 현재 블록의 해상도와

다른 해상도를 가질 수 있다. 본 개시는 이러한 문제를 해결하기 위한 방법을 제공한다.

[0105] 일 실시 예에서, 블록 벡터 예측자의 원본 해상도는 변경되지 않은 상태로 유지된다. 디코딩된 블록 벡터 차이는, 그 타깃 해상도에서, 최종 디코딩된 블록 벡터를 획득하기 위해 블록 벡터 예측자에 추가될 것이다. 따라서, 최종 디코딩된 블록 벡터는 2 가지 해상도(블록 벡터 예측자의 원본 해상도 및 디코딩된 블록 벡터 차이의 타깃 해상도) 중 더 높은 해상도가 될 것이다. 예를 들어, 블록 벡터 예측자는 1-integer-pe1 해상도를 갖는 벡터 (-11, 0)이다. 디코딩된 블록 벡터 차이는 4-integer-pe1 해상도를 갖는 벡터 (-4, 0)이다. 디코딩된 블록 벡터는 1-integer-pe1 해상도를 갖는 (-15, 0)일 것이다.

[0106] 다른 실시 예에서, 블록 벡터 예측자의 원본 해상도는, 현재 블록의 타깃 해상도로 반올림된다. 디코딩된 블록 벡터 차이는, 그 타깃 해상도에서, 블록 벡터 예측자에 추가될 것이다. 따라서, 최종 디코딩된 블록 벡터는 둘 중 더 낮은 정밀도 해상도가 될 것이다. 예를 들어, 블록 벡터 예측자는 1-integer-pe1 해상도를 갖는 벡터 (-11, 0)이다. 디코딩된 블록 벡터 차이는 4-integer-pe1 해상도를 갖는 벡터 (-4, 0)이다. 블록 벡터 예측자는, 디코딩된 블록 벡터 차이에 추가되기 전에, 먼저 4-integer-pe1 해상도를 갖는 벡터 (-12, 0)로 반올림될 것이다. 디코딩된 블록 벡터는 4-integer-pe1 해상도를 갖는 벡터 (-16, 0)일 것이다.

[0107] 다양한 반올림 기법들이 사용될 수 있다. 일 예에서, 벡터는, 차이에 따라 가장 가까운 정수(정수 심볼 값에 대응함)로 반올림될 수 있다. 예를 들어, 1-integer-pe1 해상도를 갖는 (-11, 0)은 4-integer-pe1 해상도를 갖는 (-12, 0)으로 반올림되고, 1-integer-pe1 해상도를 갖는 (-13, 0)은 4-integer-pe1 해상도를 갖는 (-12, 0)으로 반올림된다.

[0108] 다른 예에서, 올림 연산이, 현재 값보다 작지 않은 가장 가까운 정수(정수 심볼 값에 대응함)로 반올림하기 위해 벡터에 적용된다. 예를 들어, 1-integer-pe1 해상도를 갖는 (-11, 0)은 4-integer-pe1 해상도를 갖는 (-8, 0)으로 반올림되고, 1-integer-pe1 해상도를 갖는 (-13, 0)은 4-integer-pe1 해상도를 갖는 (-12, 0)으로 반올림된다.

[0109] 다른 예에서, 내림 연산이, 현재 값보다 크지 않은 가장 가까운 정수(정수 심볼 값에 대응함)로 반올림하기 위해 벡터에 적용된다. 예를 들어, 1-integer-pe1 해상도를 갖는 (-11, 0)은 4-integer-pe1 해상도를 갖는 (-12, 0)으로 반올림되고, 1-integer-pe1 해상도를 갖는 (-13, 0)은 4-integer-pe1 해상도를 갖는 (-16, 0)으로 반올림된다.

[0110] 다른 예에서, 벡터는 0 방향으로 반올림된다. 예를 들어, 1-integer-pe1 해상도를 갖는 (-11, 0)은 4-integer-pe1 해상도를 갖는 (-8, 0)으로 반올림되고, 1-integer-pe1 해상도를 갖는 (-11, -5)은 4-integer-pe1 해상도를 갖는 (8, -4)로 반올림된다.

[0111] 본 개시의 측면들은 또한 다중 해상도로 경계 제약 조건을 처리하는 기술을 제공한다. 일 예에서, 블록 벡터는, 인트라 화상 블록 보상을 사용할 수 있는 참조 영역을 가리키도록 제한된다. 제약은, 허용된 참조 영역의 화상/슬라이스/타일 및 파면 경계를 포함할 수 있다. 다중 해상도를 사용하는 경우, 특히 1-integer-pe1 보다 큰 해상도를 사용하는 경우, 경계 제약 조건을 올바르게 처리해야한다.

[0112] 일 실시 예에서, 블록 벡터(블록 벡터 예측자 + 차이)가 참조 영역 경계 외부에 있는 위치를 가리키는 경우, 클리핑 동작이 수행되어, 블록 벡터의 하나 또는 두 개의 성분이 경계의 가장자리로 돌아가도록 수정하여, 수정된 블록 벡터는 유효한 것으로 된다. 디코딩된 블록 벡터의 해상도를 고려하지 않고 클리핑 동작이 완료될 수 있다. 따라서, 수정 후, 블록 벡터의 해상도는 수정 전의 해상도와 다를 수 있다.

[0113] 다른 실시 예에서, 블록 벡터(블록 벡터 예측자 + 차이)가 참조 영역 경계 외부에 있는 위치를 가리키는 경우, 클리핑 동작이 수행되어, 벡터의 하나 또는 두 개의 성분이 경계의 가장자리를 다시 향하도록 수정하여, 수정된 블록 벡터는 유효한 것으로 된다. 클리핑 동작은 디코딩된 블록 벡터의 해상도를 고려하여 수행될 수 있다. 따라서, 수정 후에, 블록 벡터의 해상도는 이전과 동일하게 유지된다.

[0114] 다른 실시 예에서, 블록 벡터(블록 벡터 예측자 + 차이)가 참조 영역 경계 외부에 있는 위치를 가리키는 경우, 경계 외부의 픽셀은, 경계에서 픽셀을 수평 또는 수직으로 확장하여 표현될 수 있다.

[0115] 다른 실시 예에서, 블록 벡터(블록 벡터 예측자 + 차이)가 참조 영역 경계 외부에 있는 위치를 가리키는 경우, 블록 벡터 차이의 해상도는 가능한 가장 높은 정밀도로 변경될 것이다. 예를 들어, 블록 벡터 예측자는 벡터 (0, 0)이고, 블록 벡터 차이는 4-integer-pe1 해상도의 심볼 (-5, 0)이다. 가능한 가장 높은 정밀도는 1-integer-pe1이다. 디코딩된 블록 벡터가 4-integer-pe1 해상도이면, 블록 벡터는 (-20, 0)이 될 것이다. (-

20,0)이 좌측 화상 경계 외부의 위치를 가리키면, 디코딩된 블록 벡터 차이는 1-integer-pe1 해상도로 변경되어, 디코딩 블록 벡터가 (-5, 0)이 될 것이고, 이 경우 유효한 벡터이다. 이러한 변경은, 블록 벡터를 경계 내부로 이동하는 데 도움이 될 수 있다.

[0116] 본 개시의 다른 측면에 따르면, 인트라 화상 블록 보상을 위한 블록 벡터에서 다중 해상도를 처리하기 위한 기술은, 인터 화상 블록 보상을 위한 모션 벡터에 유사하게 적용될 수 있다. 일부 실시 예들에서, 모션 벡터의 코딩을 위해 다중 모션 벡터 해상도가 사용되고, 모션 벡터 해상도는 블록 레벨에서 스위칭 가능하다. 또한, 3 개 이상의 가능한 해상도가 사용될 때 1-bin 보다 많이 포함할 수 있는 시그널링 플래그가 사용되어, 다중 모션 벡터 해상도 중 어느 것이 현재 블록의 모션 벡터 차이에 대해 사용되는지 알려준다. 가능한 해상도는 1/8-pe1, 1/4-pe1, 1/2-pe1, 1-integer-pe1, 2-integer-pe1, 4-integer-pe1, 8-integer-pe1 등을 포함하지만 이에 한정되지 않는다. X-integer-pe1은 심볼의 최소 단위가 X 개의 정수 위치들을 나타냄을 의미한다. 예를 들어, 2-integer-pe1은 심볼의 각 최소 단위가 2 개의 정수 위치들을 나타냄을 의미하고, 4-integer-pe1은 심볼의 각 최소 단위가 4 개의 정수 위치를 나타냄을 의미하고, 8-integer-pe1은 심볼의 각 최소 단위가 8 개의 정수 위치를 나타냄을 의미한다.

[0117] 일 예에서, 모션 벡터 차이의 x 및 y 성분은 서로 다른 해상도를 사용할 수 있다. 예를 들어, 모션 벡터 차이의 x 성분이 1-integer-pe1 해상도에 있고, y 성분은 4-integer-pe1 해상도에 있다. 각 성분에 대한 선택은, 명시적 시그널링(성분 당 1 개의 플래그) 또는 추론을 통해 이루어질 수 있다. 예를 들어, 현재 블록의 모션 벡터 예측자의 크기는, 해상도를 추론하기 위해 사용될 수 있다. 일례로, 모션 벡터 예측자의 성분의 크기가 문턱값보다 큰 경우, 해당 성분에 대한 모션 벡터 차이 해상도는 더 큰 단계 해상도, 예를 들어 4-integer-pe1 해상도를 사용할 것이며; 그렇지 않으면, 해당 성분은 더 작은 단계 해상도, 예를 들어 1-integer-pe1 해상도를 사용할 것이다.

[0118] 다른 예에서, 모션 벡터 차이의 x 및 y 성분들은 다른 해상도를 사용할 수 있다. 디폴트로, 성분들 모두, 1/4-pe1 해상도와 같은 고정 해상도를 사용할 것이다. 성분 별 조건은, 해상도를 결정하기 위해 성분들에 대해 각각 설정된다. 예를 들어, 일부 성분 별 조건이 충족되면, 각 성분은 추론을 통해 다른 해상도로 스위칭할 수 있다. 예측자의 성분은, 또한 대응하는 해상도로 양자화되거나, 원본 해상도를 변경하지 않고 그대로 유지한다. 예를 들어, 하나의 성분 별 조건은, 현재 블록의 블록 벡터 예측자의 크기와 관련이 있다. 각 성분에서 현재 블록의 모션 벡터 예측자의 크기를 평가하여, 성분에 대한 해상도를 추론할 수 있다. 예를 들어, 모션 벡터 예측자의 성분의 크기가 문턱값보다 큰 경우, 해당 성분에 대한 모션 벡터 차이 해상도는 더 큰 단계 해상도, 예를 들어 4-integer-pe1 해상도를 사용하며; 그렇지 않으면 해당 성분은 디폴트 단계 해상도, 예를 들어 1/4-pe1 해상도를 사용할 것이다. 특정 예에서, 디폴트 해상도는 1/4-pe1이다. 모션 벡터 예측자는 (-20.75, -3.75)로 설정되고, 1-integer-pe1 해상도를 사용하는 문턱값은 각 성분에 대해 5이다. 디코딩된 모션 벡터 차이 심볼은 (2, 2)이다. 규칙에 따라, x 성분은 1-integer-pe1 해상도를 사용하고, y 성분은 1/4-pe1 해상도를 사용할 것이다. 그러면 모션 벡터 예측자는 (-20, -3.75)(예를 들어, -20.75는 -20으로 0 방향으로 반올림 됨)이되고, (x 성분에 대한 1-integer-pe1 및 y 성분에 대한 1/4-pe1로 인해) 블록 벡터 차이는 (2, 0.5)가 될 것이다. 본 예에서, 최종적으로 디코딩되는 블록 벡터는 (-18, -3.25)이다.

[0119] 도 10은 본 개시의 일 실시 예에 따른 프로세스(1000)를 개략적으로 설명하는 흐름도를 도시한다. 프로세스(1000)는 재구성 중인 블록에 대한 예측 블록을 생성하기 위해, 인트라 모드에서 코딩된 블록의 재구성에 사용될 수 있다. 다양한 실시 예에서, 프로세스(1000)는 단말 디바이스(310, 320, 330, 340)의 프로세싱 회로, 비디오 인코더(403)의 기능들을 수행하는 프로세싱 회로, 비디오 디코더(410)의 기능들을 수행하는 프로세싱 회로, 비디오 디코더(510)의 기능들을 수행하는 프로세싱 회로, 인트라 예측 모듈(552)의 기능들을 수행하는 프로세싱 회로, 비디오 인코더(603)의 기능들을 수행하는 프로세싱 회로, 예측자(635)의 기능들을 수행하는 프로세싱 회로, 인트라 인코더(722)의 기능들을 수행하는 프로세싱 회로, 인트라 디코더(872)의 기능들을 수행하는 프로세싱 회로 등과 같은 프로세싱 회로에 의해 실행된다. 일부 실시 예들에서, 프로세스(1000)는 소프트웨어 명령으로 구현되므로, 프로세싱 회로가 소프트웨어 명령을 실행할 때, 프로세싱 회로는 프로세스(1000)를 수행한다. 프로세스는 (S1001)에서 시작하여 (S1010)으로 진행된다.

[0120] (S1010)에서, 코딩된 비디오 비트스트림으로부터 현재 블록의 예측 정보가 디코딩된다. 예측 정보는 인트라 블록 카피(intra block copy) 모드를 표시한다.

[0121] (S1020)에서, 다수의 후보 해상도 세트로부터, 현재 블록의 블록 벡터 차이에 대한 해상도가 선택된다. 일 예에서, 해상도 플래그가 수신되고, 해상도는 플래그에 기초하여 선택된다. 다른 예에서, 해상도는 추론에 기초하여

결정된다.

[0122] (S1030)에서, 현재 블록의 블록 벡터는, 블록 벡터 차이의 선택된 해상도 및 현재 블록의 블록 벡터 예측자에 따라 결정된다.

[0123] (S1040)에서, 결정된 블록 벡터에 따라, 현재 블록의 샘플들이 구성된다. 이후, 프로세스는 (S1099)로 진행하여 종료한다.

[0124] 전술한 기술들은, 컴퓨터로 관독 가능한 명령을 사용하여 컴퓨터 소프트웨어로 구현되고, 하나 이상의 컴퓨터로 관독 가능한 매체에 물리적으로 저장될 수 있다. 예를 들어, 도 11은 개시된 대상의 특정 실시 예들을 구현하기에 적합한 컴퓨터 시스템(1100)을 도시한다.

[0125] 컴퓨터 소프트웨어는, 하나 이상의 컴퓨터 중앙 처리 장치(CPU), 그래픽 처리 장치(GPU) 등에 의해, 직접, 해석, 마이크로 코드 실행 등을 통해 실행될 수 있는 명령을 포함하는 코드를 생성하기 위해 어셈블리, 컴파일, 링크 또는 이와 유사한 메커니즘의 대상일 수 있는, 임의의 적절한 기계어 코드 또는 컴퓨터 언어를 사용하여 코딩될 수 있다.

[0126] 명령은, 예를 들어, 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트폰, 게임 디바이스, 사물 인터넷 디바이스 등을 포함하는, 다양한 유형의 컴퓨터 또는 그 구성 요소 상에서 실행될 수 있다.

[0127] 컴퓨터 시스템(1100)에 대해 도 11에 도시된 구성 요소들은 본질적으로 예시적이며, 본 개시의 실시 예들을 구현하는 컴퓨터 소프트웨어의 사용 또는 기능의 범위에 대한 어떠한 제한도 제안하려는 의도가 아니다. 구성 요소들의 구성은, 컴퓨터 시스템(1100)의 예시적인 실시 예에 예시된 구성 요소의 임의의 하나 또는 조합과 관련된 임의의 종속성 또는 요구 사항을 갖는 것으로 해석되어서는 안된다.

[0128] 컴퓨터 시스템(1100)은 특정 휴면 인터페이스 입력 디바이스를 포함할 수 있다. 이러한 휴면 인터페이스 입력 장치는, 촉각 입력(예를 들어, 키 입력, 스와이프, 데이터 글러브 이동(data glove movement)), 오디오 입력(예를 들어, 음성, 박수), 시각적 입력(예를 들어, 제스처), 후각 입력(도시되지 않음)을 통해 한 명 이상의 인간 사용자의 입력에 응답할 수 있다. 휴면 인터페이스 장치는 또한, 오디오(예를 들어, 음성, 음악, 주변 소리), 이미지(예를 들어, 스캔된 이미지, 정지 이미지 카메라로부터 획득한 사진 이미지), 비디오(예를 들어, 2 차원 비디오, 입체 비디오를 포함하는 3 차원 비디오)와 같이 사람의 의식적 입력과 직접 관련이 없는 특정 미디어를 캡처하기 위해서도 사용될 수 있다.

[0129] 입력 휴면 인터페이스 디바이스는: 키보드(1101), 마우스(1102), 트랙 패드(1103), 터치 스크린(1110), 데이터 글러브(도시되지 않음), 조이스틱(1105), 마이크(1106), 스캐너(1107), 카메라(1108) 중 하나 이상을 포함할 수 있다(설명된 각 항목 중 하나만 도시됨).

[0130] 컴퓨터 시스템(1100)은 또한 특정 휴면 인터페이스 출력 디바이스를 포함할 수 있다. 이러한 휴면 인터페이스 출력 장치는, 예를 들어 촉각 출력, 소리, 빛 및 냄새/맛을 통해 한 명 이상의 인간 사용자의 감각을 자극할 수 있다. 이러한 휴면 인터페이스 출력 장치는, 촉각 출력 디바이스(예를 들어, 터치 스크린(1110), 데이터 글러브(도시되지 않음) 또는 조이스틱(1105)에 의한 촉각 피드백, 그러나 입력 디바이스로 사용되지 않는 촉각 피드백 디바이스도 있을 수 있음), 오디오 출력 디바이스(예를 들어, 스피커(1109), 헤드폰(도시되지 않음)), 시각적 출력 디바이스(예를 들어, 각각 터치 스크린 입력 기능이 있거나 없고 각각 촉각 피드백 기능이 있거나 없는 CRT 스크린, LCD 스크린, 플라즈마 스크린, OLED 스크린을 포함하는 스크린 (1110) - 이들 중 일부는 스테레오 그래픽 출력과 같은 수단을 통해 2 차원 시각적 출력 또는 3 차원 이상의 출력이 가능할 수 있음; 가상 현실 안경(도시되지 않음), 홀로그램 디스플레이 및 연기 탱크(smoke tank)(도시되지 않음)) 및 프린터(도시되지 않음)가 포함될 수 있다.

[0131] 컴퓨터 시스템(1100)은 사람이 액세스할 수 있는 저장 디바이스 및 관련 매체, 예를 들어, CD/DVD 또는 유사한 미디어(1121)를 갖는 CD/DVD ROM/RW(1120)를 포함하는 광학 매체, 셤 드라이브(1122), 이동식 하드 드라이브 또는 솔리드 스테이트 드라이브(1123), 테이프 및 플로피 디스크(도시되지 않음)와 같은 레거시 자기 매체, 보안 동글(도시되지 않음)과 같은 특수 ROM/ASIC/PLD 기반 디바이스 등을 포함할 수도 있다.

[0132] 본 발명의 속하는 기술 분야에서 통상의 지식을 가진 자는 또한, 현재 개시된 대상과 관련하여 사용되는 "컴퓨터로 관독 가능한 매체"라는 용어가 전송 매체, 반송파 또는 기타 일시적인 신호를 포함하지 않는다는 것을 이해해야 한다.

[0133] 컴퓨터 시스템(1100)은 또한 하나 이상의 통신 네트워크에 대한 인터페이스를 포함할 수 있다. 네트워크는 예를

들어 무선, 유선, 광학일 수 있다. 네트워크는 또한 로컬, 광역, 대도시, 차량 및 산업, 실시간, 자연 허용(delay-tolerant) 등이 될 수 있다. 네트워크의 예로는, 이더넷과 같은 근거리 네트워크, 무선 LAN, GSM, 3G, 4G, 5G, LTE 등을 포함하는 셀룰러 네트워크, 케이블 TV, 위성 TV 및 지상파 방송 TV를 포함하는 TV 유선 또는 무선 광역 디지털 네트워크, CANBus를 포함하는 차량 및 산업용 등을 들 수 있다. 특정 네트워크는 일반적으로 특정 범용 데이터 포트 또는 주변 장치 버스(1149)(예를 들어, 컴퓨터 시스템(1100)의 USB 포트)에 부착된 외부 네트워크 인터페이스 어댑터를 필요로 하며; 다른 것들은 일반적으로 아래에 설명된 바와 같이 시스템 버스에 부착하여 컴퓨터 시스템(1100)의 코어에 통합된다(예를 들어, PC 컴퓨터 시스템으로의 이더넷 인터페이스 또는 스마트 폰 컴퓨터 시스템으로의 셀룰러 네트워크 인터페이스). 임의의 이들 네트워크를 사용하여, 컴퓨터 시스템(1100)은 다른 엔티티와 통신할 수 있다. 이러한 통신은, 단방향, 수신 전용(예를 들어, 방송 TV), 단방향 송신 전용(예를 들어, 특정 CANbus 디바이스에 대한 CANbus) 또는 양방향, 예를 들어, 로컬 또는 광역 디지털 네트워크를 사용하는 다른 컴퓨터 시스템일 수 있다. 특정 프로토콜 및 프로토콜 스택은, 위에서 설명한 바와 같이 이들 네트워크 및 네트워크 인터페이스 각각에서 사용될 수 있다.

[0134] 전술한 휴면 인터페이스 디바이스, 휴면 액세스 가능 저장 디바이스 및 네트워크 인터페이스는 컴퓨터 시스템(1100)의 코어(1140)에 부착될 수 있다.

[0135] 코어(1140)는 하나 이상의 중앙 처리 장치(CPU)(1141), 그래픽 처리 장치(GPU)(1142), FPGA(Field Programmable Gate Areas)(1143) 형태의 특수 프로그램 가능 처리 장치, 특정 작업에 대한 하드웨어 가속기(1144) 등을 포함할 수 있다. 읽기 전용 메모리(ROM)(1145), 랜덤 액세스 메모리(1146), 내부의 사용자 액세스 할 수 없는 하드 드라이브, SSD 등과 같은 내부 대용량 저장소(1147)와 함께, 이들 디바이스는 시스템 버스(1148)를 통해 연결될 수 있다. 일부 컴퓨터 시스템에서, 시스템 버스(1148)는, 추가적인 CPU, GPU 등에 의한 확장을 가능하게 하기 위해 하나 이상의 물리적 플러그의 형태로 액세스 가능할 수 있다. 주변 디바이스는 코어의 시스템 버스(1148)에 직접 부착되거나 주변 버스(1149)를 통해 부착될 수 있다. 주변 버스의 아키텍처에는 PCI, USB 등이 포함된다.

[0136] CPU(1141), GPU(1142), FPGA(1143) 및 가속기(1144)는, 조합하여, 앞서 언급한 컴퓨터 코드를 구성할 수 있는 특정 명령을 실행할 수 있다. 해당 컴퓨터 코드는 ROM(1145) 또는 RAM(1146)에 저장될 수 있다. 전환 데이터(transitional data)는 RAM(1146)에도 저장될 수 있는 반면, 영구 데이터는 예를 들어 내부 대용량 저장소(1147)에 저장될 수 있다. 하나 이상의 CPU(1141), GPU(1142), 대용량 저장소(1147), ROM(1145), RAM(1146) 등과 밀접하게 연관될 수 있는, 캐시 메모리를 사용하여 모든 메모리 디바이스에 대한 빠른 저장 및 검색을 활성화할 수 있다.

[0137] 컴퓨터로 판독 가능한 매체는, 다양한 컴퓨터로 구현된 동작을 수행하기 위한 컴퓨터 코드를 가질 수 있다. 매체 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.

[0138] 일 예로서 제한 없이, 아키텍처(1100)를 갖는 컴퓨터 시스템과, 특히 코어(1140)는, 하나 이상의 유형의 컴퓨터로 판독 가능한 매체에 구현된 소프트웨어 실행하는 프로세서(들)(CPU, GPU, FPGA, 가속기 등을 포함함)의 결과로서 기능을 제공할 수 있다. 이러한 컴퓨터로 판독 가능한 매체는, 위에서 소개한 사용자가 액세스 가능 대용량 저장소와 연관된 매체일 수 있으며, 코어 내부 대용량 저장소(1147) 또는 ROM(1145)과 같은, 비 일시적 특성인 코어(1140)의 특정 저장소일 수 있다. 본 개시의 다양한 실시 예들을 구현하는 소프트웨어는, 이러한 디바이스에 저장되고 코어(1140)에 의해 실행될 수 있다. 컴퓨터로 판독 가능한 매체는 특정 요구에 따라 하나 이상의 메모리 디바이스 또는 칩을 포함할 수 있다. 소프트웨어는, 코어(1140) 및 특히 그 안의 프로세서(CPU, GPU, FPGA 등 포함함)가, RAM(1146)에 저장된 데이터 구조를 정의하는 것을 포함하여, 여기에서 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하도록 할 수 있고, 소프트웨어에 의해 정의된 프로세스에 따라 이러한 데이터 구조를 수정한다. 추가로 또는 대안적으로, 컴퓨터 시스템은, 회로(예를 들어, 가속기(1144))에 하드 와이어되거나 아니면 구현된 로직에 대한 결과로 기능을 제공할 수 있고, 여기에서 설명되는 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하기 위해 소프트웨어 대신 또는 소프트웨어와 함께 동작할 수 있다. 소프트웨어에 대한 참조는 논리를 포함할 수 있으며, 적절하다면, 그 반대의 경우도 마찬가지이다. 컴퓨터로 판독 가능한 매체에 대한 참조는, 실행을 위한 소프트웨어를 저장하는 회로(예를 들어, 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.

[0139] 부록 A: 약어(Acronyms)

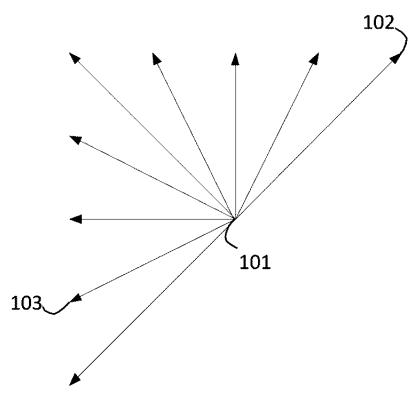
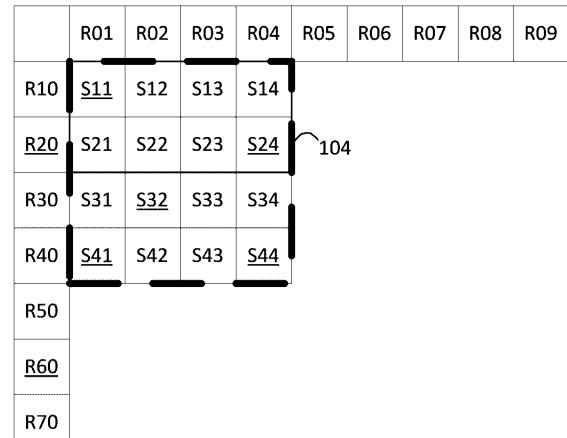
[0140]	JEM: 공동 탐사 모델
[0141]	VVC: 다목적 비디오 코딩
[0142]	BMS: 벤치마크 세트
[0143]	MV: 모션 벡터
[0144]	HEVC: 고효율 비디오 코딩
[0145]	SEI: 보충 강화 정보
[0146]	VUI: 비디오 사용성 정보
[0147]	GOPs: 화상 그룹
[0148]	TUs: 변환 유닛,
[0149]	PUs: 예측 유닛
[0150]	CTUs: 코딩 트리 유닛
[0151]	CTBs: 코딩 트리 블록
[0152]	PBs: 예측 블록
[0153]	HRD: 가상 참조 디코더
[0154]	SNR: 신호 잡음비
[0155]	CPUs: 중앙 처리 장치
[0156]	GPUs: 그래픽 처리 장치
[0157]	CRT: 음극선 관
[0158]	LCD: 액정 디스플레이
[0159]	OLED: 유기 발광 다이오드
[0160]	CD: 컴팩트 디스크
[0161]	DVD: 디지털 비디오 디스크
[0162]	ROM: 읽기 전용 메모리
[0163]	RAM: 랜덤 액세스 메모리
[0164]	ASIC: 응용 주문형 집적 회로
[0165]	PLD: 프로그래밍 가능한 논리 디바이스
[0166]	LAN: 근거리 네트워크
[0167]	GSM: 이동 통신을 위한 글로벌 시스템
[0168]	LTE: 루 텁 애블루션
[0169]	CANBus: 제어기 영역 네트워크 버스
[0170]	USB: 범용 직렬 버스
[0171]	PCI: 주변 구성 요소 상호 연결
[0172]	FPGA: 필드 프로그램 가능 게이트 영역
[0173]	SSD: 솔리드 스테이트 드라이브
[0174]	IC: 집적 회로
[0175]	CU: 코딩 유닛

[0176]

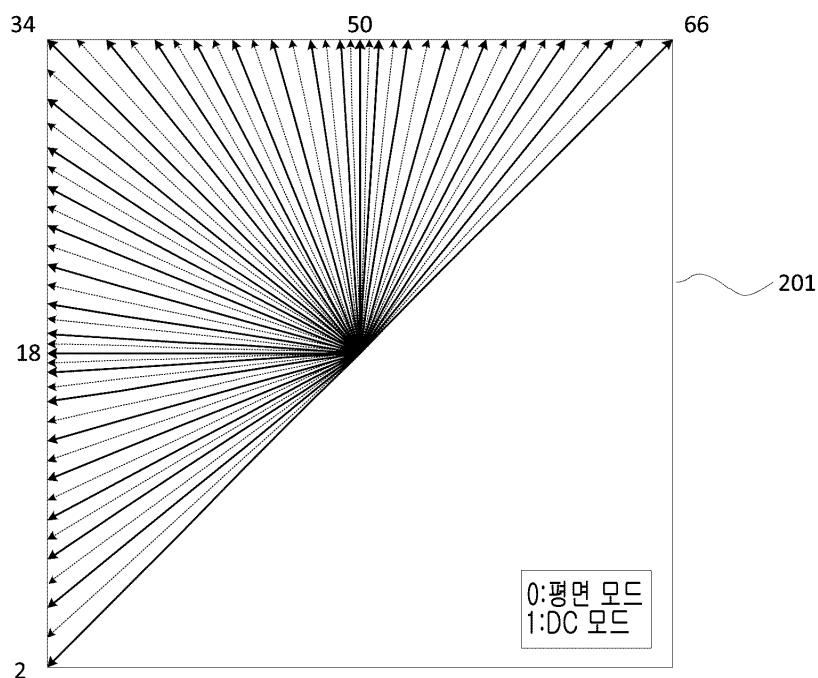
본 개시는 여러 예시적인 실시 예들을 설명했지만, 본 개시의 범위 내에 속하는 변경, 순열 및 다양한 대체 등 가물이 존재한다. 따라서, 본 발명의 속하는 기술 분야에서 통상의 지식을 가진 자는, 본 명세서에 명시적으로 도시되거나 설명되지 않았지만, 본 개시의 원리를 구현하고 따라서 그 사상 및 범위 내에 있는 수많은 시스템 및 방법을 고안할 수 있다는 것을 이해할 것이다.

도면

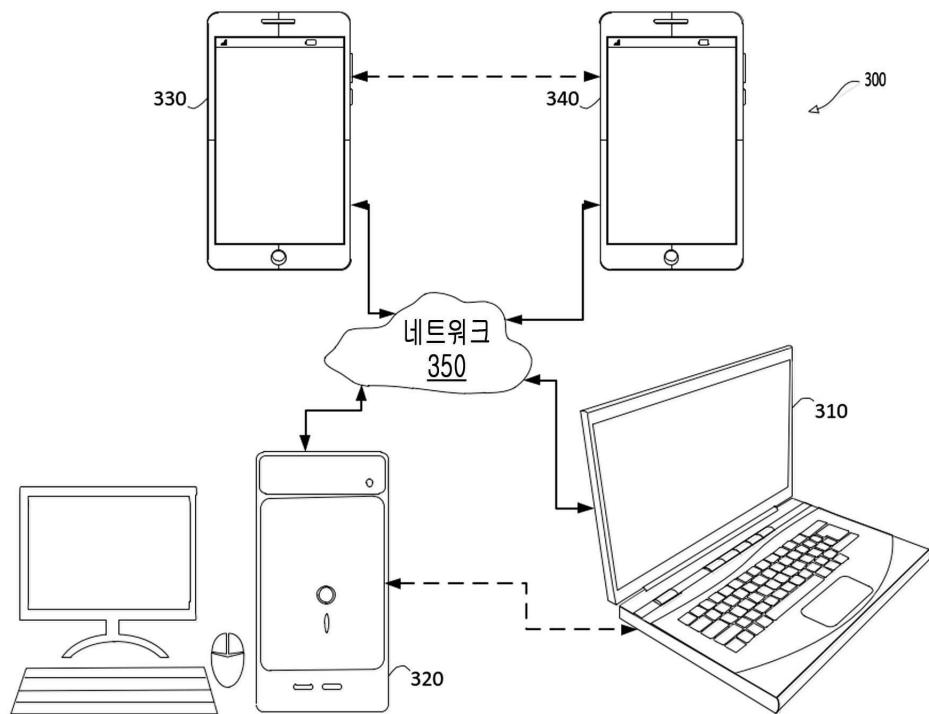
도면1



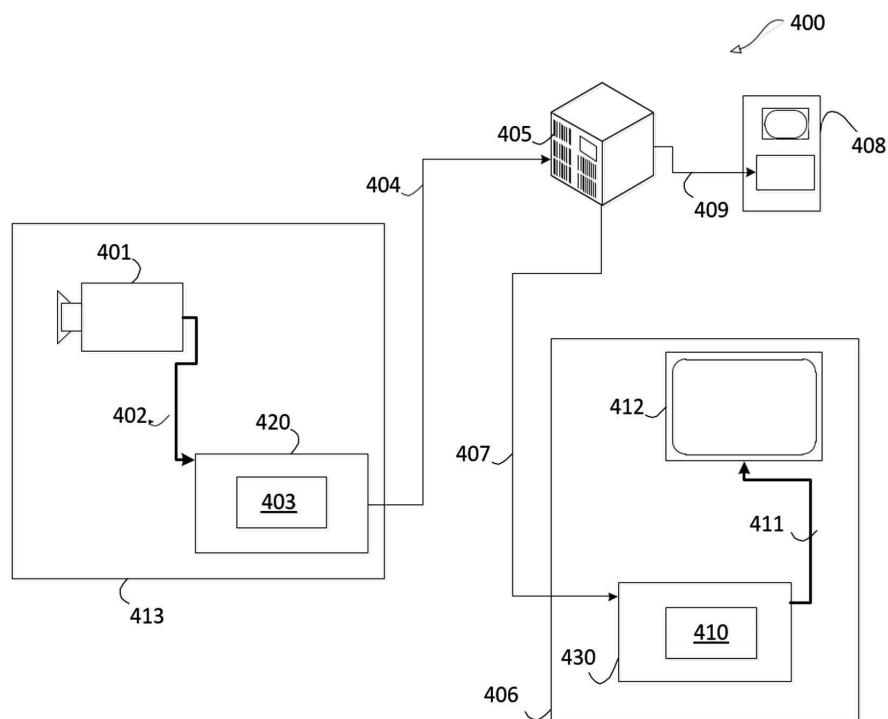
도면2



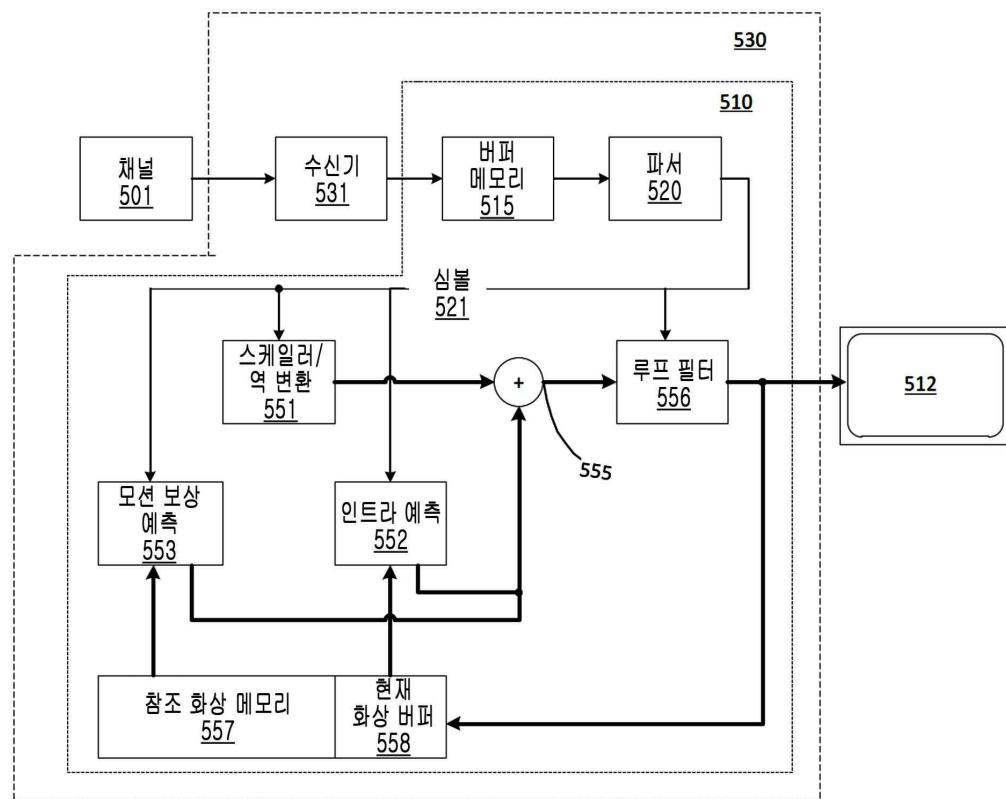
도면3



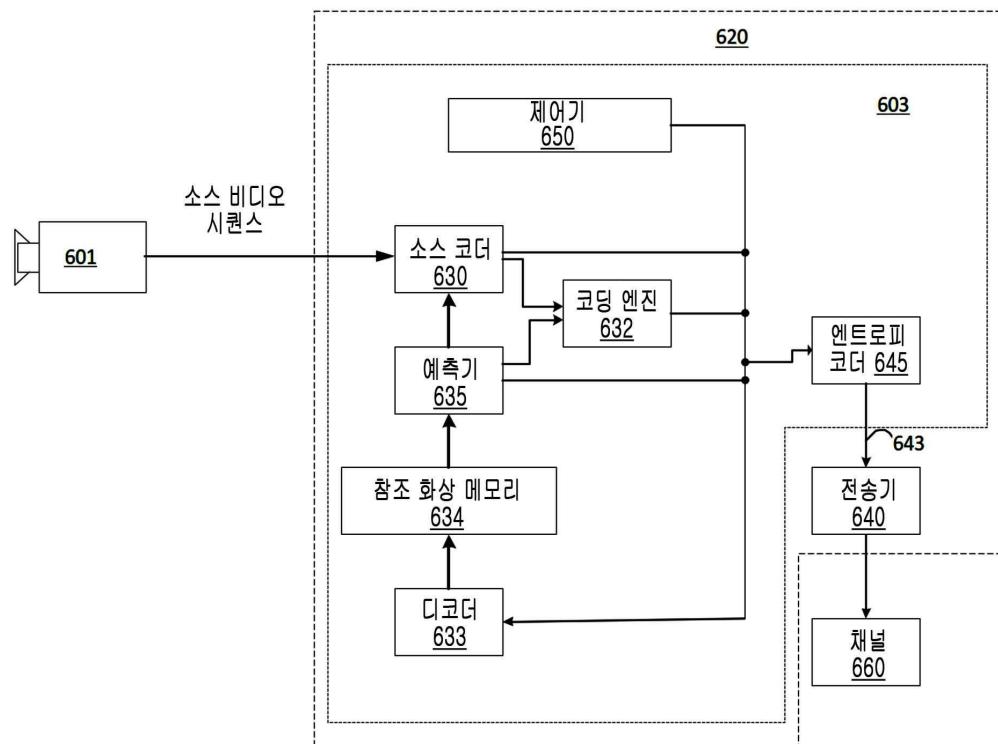
도면4



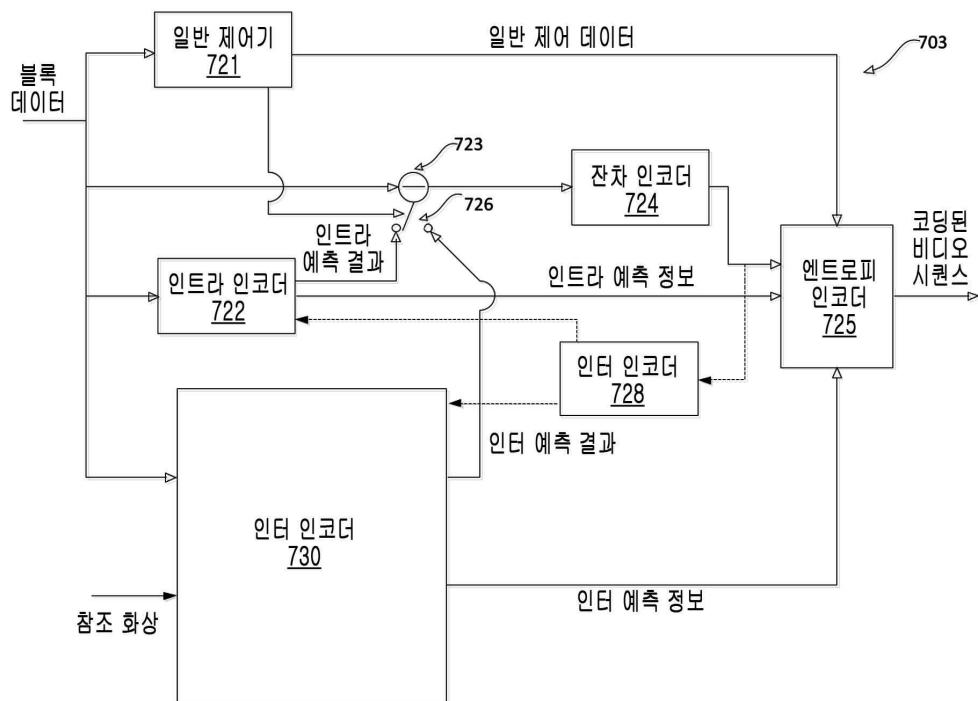
도면5



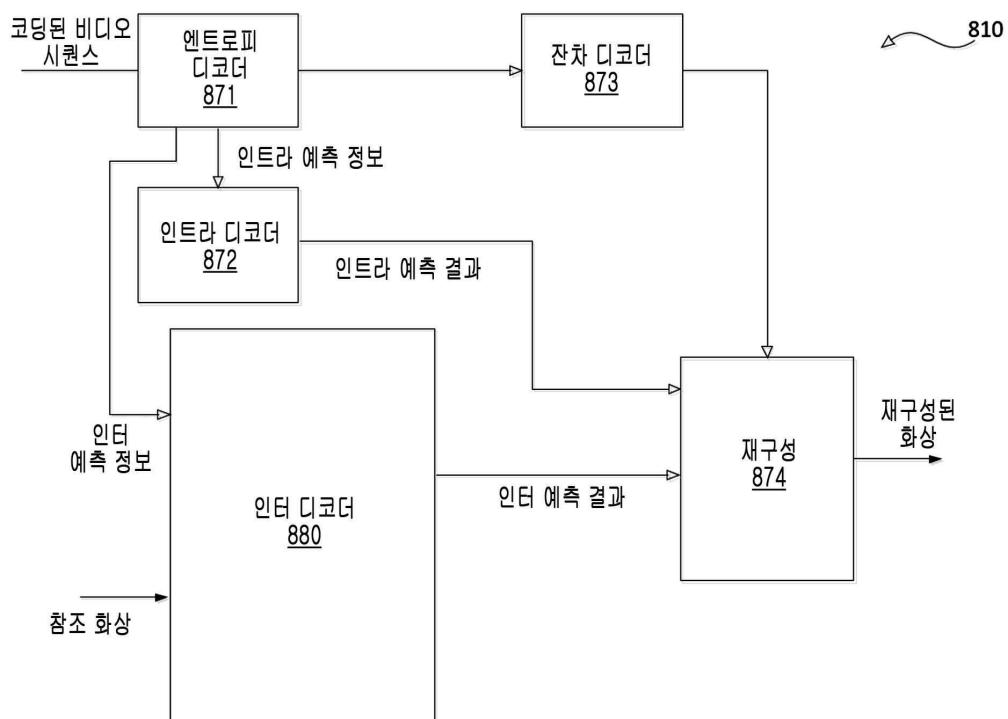
도면6



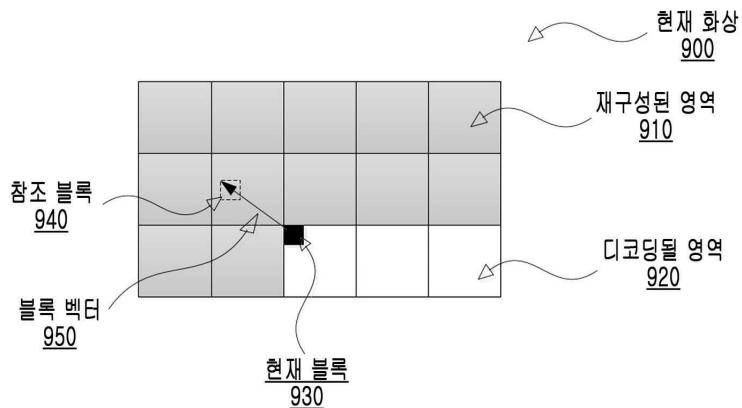
도면7



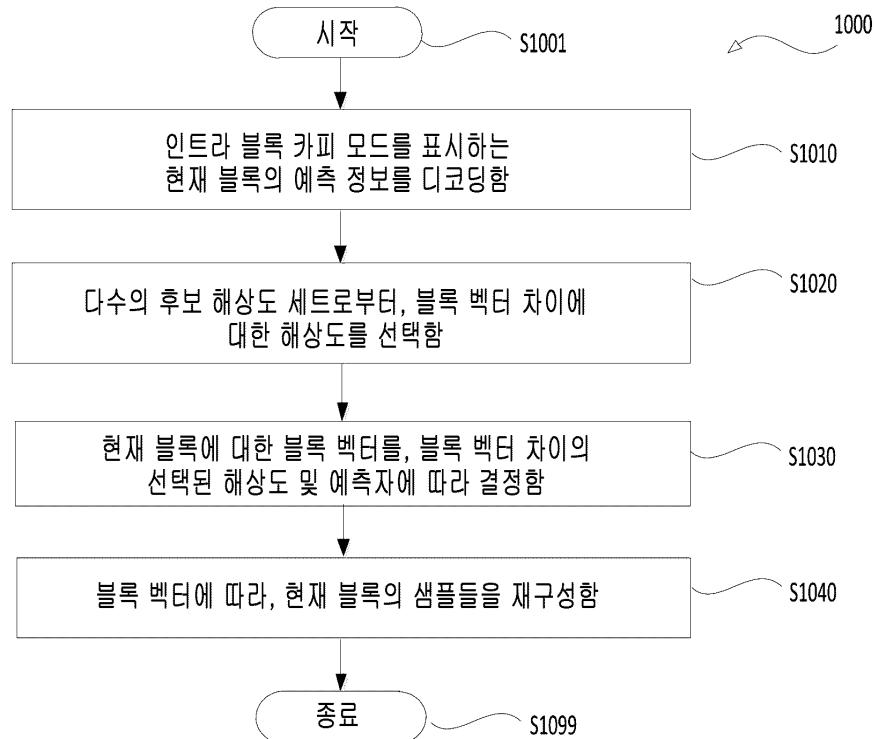
도면8



도면9



도면10



도면11

