



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2024년09월20일
(11) 등록번호 10-270777
(24) 등록일자 2024년09월12일

- (51) 국제특허분류(Int. Cl.)
HO4N 19/11 (2014.01) HO4N 19/105 (2014.01)
HO4N 19/124 (2014.01) HO4N 19/176 (2014.01)
HO4N 19/186 (2014.01) HO4N 19/593 (2014.01)
HO4N 19/70 (2014.01)
- (52) CPC특허분류
HO4N 19/11 (2015.01)
HO4N 19/105 (2015.01)
- (21) 출원번호 10-2021-7033193
- (22) 출원일자(국제) 2020년04월30일
심사청구일자 2023년04월04일
- (85) 번역문제출일자 2021년10월14일
- (65) 공개번호 10-2022-0002292
- (43) 공개일자 2022년01월06일
- (86) 국제출원번호 PCT/US2020/030684
- (87) 국제공개번호 WO 2020/223467
국제공개일자 2020년11월05일
- (30) 우선권주장
PCT/CN2019/085398 2019년05월01일 중국(CN)
- (56) 선행기술조사문헌
Karczewicz (Qualcomm) M et al: "CE8-related: Quantized residual BDPCM", JVET-N0413, version5, 2019-3-26.
Henry (ORANGE) F et al: "CE8: BDPCM with horizontal/vertical predictor and independently decodable areas (test 8.3.1b)", JVET-M0057, 2019.1.10.

- (73) 특허권자
바이트댄스 아이엔씨
미국, 90066 캘리포니아 로스엔젤레스 스위트룸 137호 6층 제퍼슨 블러바드 웨스트 12655
- (72) 발명자
주, 웨이지아
미국 90066 캘리포니아 로스엔젤레스 스위트룸 137호 6층 제퍼슨 블러바드 웨스트 12655
장, 리
미국 90066 캘리포니아 로스엔젤레스 스위트룸 137호 6층 제퍼슨 블러바드 웨스트 12655
(뒷면에 계속)
- (74) 대리인
특허법인 무한

전체 청구항 수 : 총 12 항

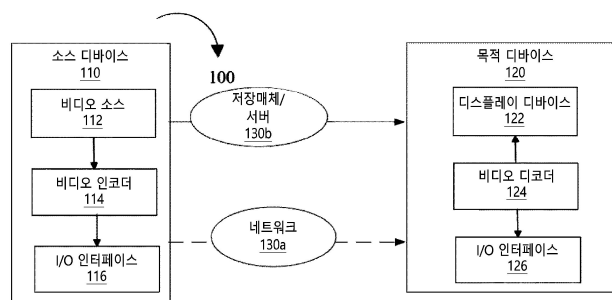
심사관 : 김영태

(54) 발명의 명칭 양자화된 잔차 차동 펄스 코드 변조 코딩을 사용하는 인트라 코딩된 비디오

(57) 요약

비디오 코딩 및 디코딩 방법이 개시된다. 일 예의 방법은, 차동 코딩 모드를 사용하여, 현재 비디오 블록과 연관되어 저장되는 제1 인트라 코딩 모드를 결정함으로써 비디오의 상기 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환을 수행하는 단계를 포함하고, 상기 현재 비디오 블록과 연관된 상기 제1 인트라 (뒷면에 계속)

대표도



코딩 모드는 상기 차동 코딩 모드에 의해 사용되는 제2 예측 모드에 따라 결정되며, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 인트라 예측의 양자화된 잔차와 상기 양자화된 잔차의 예측 간의 차이는 차동 펄스 코딩 변조(Differential Pulse Coding Modulation; DPCM) 표현을 사용하여 상기 현재 비디오 블록의 비트스트림 표현으로 나타난다.

(52) CPC특허분류

H04N 19/124 (2015.01)

H04N 19/176 (2015.01)

H04N 19/186 (2015.01)

H04N 19/593 (2015.01)

H04N 19/70 (2015.01)

(72) 발명자

슈, 지즈영

미국 90066 캘리포니아 로스엔젤레스 스위트룸 13
7호 6층 제퍼슨 블러바드 웨스트 12655

추앙, 호시아오 치앙

미국 90066 캘리포니아 로스엔젤레스 스위트룸 13
7호 6층 제퍼슨 블러바드 웨스트 12655

명세서

청구범위

청구항 1

비디오 프로세싱 방법에 있어서,

비디오의 제1 비디오 블록에 차동 코딩 모드가 적용되는지 결정하는 단계 - 상기 차동 코딩 모드에서 상기 제1 비디오 블록의 제1 인트라 예측 모드에서 유도된 양자화된 잔차(quantized residual)와 상기 양자화된 잔차의 예측 간의 차이는 상기 비디오의 비트스트림에 포함되고, 상기 차동 코딩 모드는 상기 양자화된 잔차에 대한 블록 차동 펄스-코드 변조임 -;

상기 제1 비디오 블록에 대한 상기 차동 코딩 모드에서 사용된 예측 방향에 대해 정렬(aligned)되어, 상기 제1 비디오 블록의 상기 제1 인트라 예측 모드를 저장하는 단계;

상기 제1 비디오 블록의 상기 저장된 제1 인트라 예측 모드에 기초하여, 상기 비디오의 제2 비디오 블록의 모드 후보 리스트를 구성하는 단계 - 상기 제1 비디오 블록은 상기 제2 비디오 블록의 공간적 인접(spatial neighboring) 블록이고, 상기 제2 비디오 블록은 인트라 블록이고, 상기 모드 후보 리스트는 MPM(most probable modes)리스트임 -;

상기 모드 후보 리스트에 기초하여, 상기 제2 비디오 블록의 제2 인트라 예측 모드를 결정하는 단계; 및

상기 제2 인트라 예측 모드에 기초하여, 상기 제2 비디오 블록과 상기 비디오의 상기 비트스트림 간의 변환을 수행하는 단계

를 포함하고,

상기 제1 비디오 블록이 사용 가능하지 않은 경우, 상기 제1 비디오 블록에 대응하는 상기 모드 후보 리스트 내의 후보는 플래나(planar) 인트라 예측 모드로 설정되고,

상기 차동 코딩 모드가 상기 제1 비디오 블록에 적용되는 것에 응답하여, 크로스-컴포넌트 선형 모델 코딩 모드가 상기 제1 비디오 블록에 적용되지 않고, 및

상기 제1 비디오 블록이 루마 블록이고, 제3 신택스 요소(element)가 상기 차동 코딩 모드가 크로마 컴포넌트에 적용되는지 여부를 나타내는데 더 사용되는,

비디오 프로세싱 방법.

청구항 2

제1항에 있어서,

상기 제1 인트라 예측 모드는, 상기 제1 비디오 블록의 상기 차동 코딩 모드에서 사용되는 상기 예측 방향이 수직 방향인 것에 응답하여, 수직 인트라 예측 모드로 추론되는(inferred),

비디오 프로세싱 방법.

청구항 3

제1항에 있어서,

상기 제1 인트라 예측 모드는, 상기 제1 비디오 블록의 상기 차동 코딩 모드에서 사용되는 상기 예측 방향이 수평 방향인 것에 응답하여, 수평 인트라 예측 모드로 추론되는,

비디오 프로세싱 방법.

청구항 4

제1항에 있어서,

상기 제1 비디오 블록의 상기 차동 코딩 모드의 상기 예측 방향은, 상기 예측 방향이 수평 또는 수직 방향인지를 나타내는 상기 비트스트림에 포함된 제1 선택스 요소(element)에 기초하여 결정되는,

비디오 프로세싱 방법.

청구항 5

제1항에 있어서,

상기 제1 비디오 블록은, 상기 제2 비디오 블록의 인접 블록의 레프트(left) 또는 위(above)인,

비디오 프로세싱 방법.

청구항 6

제1항에 있어서,

상기 제2 인트라 예측 모드는 상기 비트스트림의 제2 선택스 요소에 기초하여 결정되고, 상기 제2 선택스 요소는 상기 제2 비디오 블록의 인트라 예측에 사용되는 상기 모드 후보 리스트 내 후보의 인덱스를 나타내는,

비디오 프로세싱 방법.

청구항 7

제1항에 있어서,

상기 차이는 차동 펄스 코딩 변조 표현(representation)을 이용하여 표현되는,

비디오 프로세싱 방법.

청구항 8

제1항에 있어서,

상기 변환은 상기 제2 비디오 블록을 상기 비트스트림으로 인코딩하는 것을 포함하는,

비디오 프로세싱 방법.

청구항 9

제1항에 있어서,

상기 변환은 상기 비트스트림으로부터 상기 제2 비디오 블록을 디코딩하는 것을 포함하는,

비디오 프로세싱 방법.

청구항 10

비디오 프로세싱 장치에 있어서,

프로세서; 및

명령어를 저장하는 비-일시적(non-transitory) 메모리

를 포함하고, 상기 명령어는 상기 프로세서에 의해 실행되면 상기 프로세서로 하여금,

비디오의 제1 비디오 블록에 차동 코딩 모드가 적용되는지 결정하는 동작 - 상기 차동 코딩 모드에서 상기 제1 비디오 블록의 제1 인트라 예측 모드에서 유도된 양자화된 잔차(quantized residual)와 상기 양자화된 잔차의 예측 간의 차이는 상기 비디오의 비트스트림에 포함되고, 상기 차동 코딩 모드는 상기 양자화된 잔차에 대한 블록 차동 펄스-코드 변조임 -;

상기 제1 비디오 블록에 대한 상기 차동 코딩 모드에서 사용된 예측 방향에 대해 정렬(aligned)되어, 상기 제1 비디오 블록의 상기 제1 인트라 예측 모드를 저장하는 동작;

상기 제1 비디오 블록의 상기 저장된 제1 인트라 예측 모드에 기초하여, 상기 비디오의 제2 비디오 블록의 모드 후보 리스트를 구성하는 동작 - 상기 제1 비디오 블록은 상기 제2 비디오 블록의 공간적 인접(spatial neighboring) 블록이고, 상기 제2 비디오 블록은 인트라 블록이고, 상기 모드 후보 리스트는 MPM(most probable modes)리스트임 -;

상기 모드 후보 리스트에 기초하여, 현재 비디오 블록의 제2 인트라 예측 모드를 결정하는 동작; 및

상기 제2 인트라 예측 모드에 기초하여, 상기 제2 비디오 블록과 상기 비디오의 상기 비트스트림 간의 변환을 수행하는 동작

을 수행하도록 하고,

상기 제1 비디오 블록이 사용 가능하지 않은 경우, 상기 제1 비디오 블록에 대응하는 상기 모드 후보 리스트 내의 후보는 플래나(planar) 인트라 예측 모드로 설정되고,

상기 차동 코딩 모드가 상기 제1 비디오 블록에 적용되는 것에 응답하여, 크로스-컴포넌트 선형 모델 코딩 모드가 상기 제1 비디오 블록에 적용되지 않고, 및

상기 제1 비디오 블록이 루마 블록이고, 제3 선택 요소(element)가 상기 차동 코딩 모드가 크로마 컴포넌트에 적용되는지 여부를 나타내는데 더 사용되는,

비디오 프로세싱 장치.

청구항 11

명령어를 저장하는 비-일시적 컴퓨터 판독 가능 기록 매체에 있어서,

상기 명령어는 프로세서로 하여금,

비디오의 제1 비디오 블록에 차동 코딩 모드가 적용되는지 결정하는 동작 - 상기 차동 코딩 모드에서 상기 제1 비디오 블록의 제1 인트라 예측 모드에서 유도된 양자화된 잔차(quantized residual)와 상기 양자화된 잔차의 예측 간의 차이는 상기 비디오의 비트스트림에 포함되고, 상기 차동 코딩 모드는 상기 양자화된 잔차에 대한 블록 차동 펄스-코드 변조임 -;

상기 제1 비디오 블록에 대한 상기 차동 코딩 모드에서 사용된 예측 방향에 대해 정렬(aligned)되어, 상기 제1 비디오 블록의 상기 제1 인트라 예측 모드를 저장하는 동작;

상기 제1 비디오 블록의 상기 저장된 제1 인트라 예측 모드에 기초하여, 상기 비디오의 제2 비디오 블록의 모드 후보 리스트를 구성하는 동작 - 상기 제1 비디오 블록은 상기 제2 비디오 블록의 공간적 인접(spatial neighboring) 블록이고, 상기 제2 비디오 블록은 인트라 블록이고, 상기 모드 후보 리스트는 MPM(most probable modes)리스트임 -;

상기 모드 후보 리스트에 기초하여, 현재 비디오 블록의 제2 인트라 예측 모드를 결정하는 동작; 및

상기 제2 인트라 예측 모드에 기초하여, 상기 제2 비디오 블록과 상기 비디오의 상기 비트스트림 간의 변환을 수행하는 동작

을 수행하도록 하고,

상기 제1 비디오 블록이 사용 가능하지 않은 경우, 상기 제1 비디오 블록에 대응하는 상기 모드 후보 리스트 내의 후보는 플래나(planar) 인트라 예측 모드로 설정되고,

상기 차동 코딩 모드가 상기 제1 비디오 블록에 적용되는 것에 응답하여, 크로스-컴포넌트 선형 모델 코딩 모드가 상기 제1 비디오 블록에 적용되지 않고, 및

상기 제1 비디오 블록이 루마 블록이고, 제3 신택스 요소(element)가 상기 차동 코딩 모드가 크로마 컴포넌트에 적용되는지 여부를 나타내는데 더 사용되는,

비-일시적 컴퓨터 판독 가능 기록 매체.

청구항 12

비디오 프로세싱 장치에 의해 수행되는 방법에 의해 생성되는 비디오의 비트스트림을 저장하는 비-일시적 컴퓨터 판독 가능 기록 매체에 있어서,

상기 방법은,

비디오의 제1 비디오 블록에 차동 코딩 모드가 적용되는지 결정하는 단계 - 상기 차동 코딩 모드에서 상기 제1 비디오 블록의 제1 인트라 예측 모드에서 유도된 양자화된 잔차(quantized residual)와 상기 양자화된 잔차의 예측 간의 차이는 상기 비디오의 비트스트림에 포함되고, 상기 차동 코딩 모드는 상기 양자화된 잔차에 대한 블록 차동 펄스-코드 변조임 -;

상기 제1 비디오 블록에 대한 상기 차동 코딩 모드에서 사용된 예측 방향에 대해 정렬(aligned)되어, 상기 제1 비디오 블록의 상기 제1 인트라 예측 모드를 저장하는 단계;

상기 제1 비디오 블록의 상기 저장된 제1 인트라 예측 모드에 기초하여, 상기 비디오의 제2 비디오 블록의 모드 후보 리스트를 구성하는 단계 - 상기 제1 비디오 블록은 상기 제2 비디오 블록의 공간적 인접(spatial neighboring) 블록이고, 상기 제2 비디오 블록은 인트라 블록이고, 상기 모드 후보 리스트는 MPM(most probable modes)리스트임 -;

상기 모드 후보 리스트에 기초하여, 현재 비디오 블록의 제2 인트라 예측 모드를 결정하는 단계; 및

상기 제2 인트라 예측 모드에 기초하여, 상기 비트스트림을 생성하는 단계

를 포함하고,

상기 제1 비디오 블록이 사용 가능하지 않은 경우, 상기 제1 비디오 블록에 대응하는 상기 모드 후보 리스트 내의 후보는 플래나(planar) 인트라 예측 모드로 설정되고,

상기 차동 코딩 모드가 상기 제1 비디오 블록에 적용되는 것에 응답하여, 크로스-컴포넌트 선형 모델 코딩 모드가 상기 제1 비디오 블록에 적용되지 않고, 및

상기 제1 비디오 블록이 루마 블록이고, 제3 신택스 요소(element)가 상기 차동 코딩 모드가 크로마 컴포넌트에 적용되는지 여부를 나타내는데 더 사용되는,

비-일시적 컴퓨터 판독 가능 기록 매체.

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

- 청구항 16
삭제
- 청구항 17
삭제
- 청구항 18
삭제
- 청구항 19
삭제
- 청구항 20
삭제
- 청구항 21
삭제
- 청구항 22
삭제
- 청구항 23
삭제
- 청구항 24
삭제
- 청구항 25
삭제
- 청구항 26
삭제
- 청구항 27
삭제
- 청구항 28
삭제
- 청구항 29
삭제
- 청구항 30
삭제
- 청구항 31
삭제

청구항 32

삭제

발명의 설명

기술 분야

[0001] 이 출원은 2019년 5월 1일에 출원된 국제 특허 출원 번호 PCT/CN2019/085398의 우선권과 이익을 주장한, 국제 특허 출원 번호 PCT/US2020/030684의 국내단계 진입이다. 상기 출원의 전체 개시는 본 출원의 개시의 일부로 참조에 의해 통합된다.

[0002] 이 특허 문서는 비디오 코딩 기술, 장치 및 시스템과 관련이 있다.

배경 기술

[0003] 비디오 압축의 발전에도 불구하고 디지털 비디오는 여전히 인터넷 및 기타 디지털 통신 네트워크에서 가장 큰 대역폭 사용을 차지한다. 비디오를 수신하고 표시할 수 있는 연결된 사용자 디바이스의 수가 증가함에 따라 디지털 비디오 사용에 대한 대역폭 수요가 계속 증가할 것으로 예상된다.

발명의 내용

해결하려는 과제

[0004] 본 문서는 비디오 또는 이미지의 디코딩 또는 인코딩 중에 보조(secondary) 변환이 사용되는 다양한 구현에 맞 기술을 설명한다.

과제의 해결 수단

[0005] 일 예에서, 비디오 프로세싱 방법은, 차동 코딩 모드를 사용하여, 현재 비디오 블록과 연관되어 저장되는 제1 인트라 코딩 모드를 결정함으로써 비디오의 상기 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현(representation) 간의 변환을 수행하는 단계를 포함하고, 상기 현재 비디오 블록과 연관된 상기 제1 인트라 코딩 모드는 상기 차동 코딩 모드에 의해 사용되는 제2 예측 모드에 따라 결정되며, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 인트라 예측의 양자화된 잔차와 상기 양자화된 잔차의 예측 간의 차이는 차동 펄스 코딩 변조(Differential Pulse Coding Modulation; DPCM) 표현을 사용하여 상기 현재 비디오 블록의 비트스트림 표현으로 나타난다.

[0006] 다른 일 예에서, 비디오 프로세싱 방법은, 규칙에 따라, 비디오의 현재 비디오 블록과 상기 현재 비디오 블록의 비트 스트림 표현 간의 변환 중에 차동 코딩 모드에 의해 사용되는 인트라 코딩 모드를 결정하는 단계; 및 상기 결정에 기초하여, 상기 차동 코딩 모드를 사용하여 상기 현재 비디오 블록과 상기 현재 비디오 블록의 상기 비트 스트림 표현 사이의 변환을 수행하는 단계를 포함하고, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 인트라 예측의 양자화된 잔차와 상기 양자화된 잔차의 예측 간의 차이는 차동 펄스 코딩 변조(DPCM) 표현을 이용하여 현재 비디오 블록에 대한 비트스트림 표현으로 나타나며, 상기 양자화된 잔차의 상기 예측은 상기 인트라 코딩 모드에 따라 수행된다.

[0007] 또 다른 예에서, 비디오 처리 방법이 개시된다. 상기 방법은, 차동 코딩 모드 및 선택적으로 공존(coexistence) 규칙에 기반한 인트라 예측 모드를 이용하여 현재 비디오 블록과 상기 현재 비디오 블록의 비트 스트림 표현(representation) 간의 변환을 수행하는 단계를 포함하고, 상기 인트라 예측 모드는 상기 현재 비디오 블록의 샘플들의 예측들을 생성하는데 이용되고, 상기 차동 코딩 모드는 차동 펄스 코딩 변조 표현을 이용하여 상기 픽셀들의 상기 예측으로부터 양자화된 잔차 블록을 표현하는데 사용된다.

[0008] 또 다른 예에서, 또 다른 비디오 프로세싱 방법이 개시된다. 상기 방법은, 현재 비디오 블록의 픽셀 예측으로부터 정량화된 잔차 블록이 차동 펄스 코딩 변조 표현을 사용하여 표현되는 차동 코딩 모드를 사용하여 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 변환을 수행하는 것을 포함하고, 상기 예측의 제1 방향(direction) 또는 상기 차동 코딩 모드의 제2 방향은 상기 비트스트림 표현으로부터 추론된다.

[0009] 또 다른 예에서, 또 다른 비디오 프로세싱 방법이 개시된다. 상기 방법은, 적용 가능성 규칙에 기초하여, 현재

비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환에 차동 코딩 모드가 적용 가능함을 결정하는 단계; 상기 차동 코딩 모드를 사용하여 현재 비디오 블록과 비트스트림 표현 간의 변환을 수행하는 단계를 포함한다. 여기서, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 픽셀의 인트라 예측으로부터 양자화된 잔차 블록은 수평 또는 수직 방향과 다른 잔차 예측 방향에서 수행되는 차동 펄스 코딩 변조 표현을 사용하여 표현된다.

[0010] 또 다른 예에서, 또 다른 비디오 프로세싱 방법이 개시된다. 상기 방법은, 차동 코딩 모드가 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환에 적용 가능한지 결정하는 단계; 및 차동 코딩 모드가 적용 가능함을 결정하는 단계; 상기 차동 코딩 모드의 구현 규칙을 사용하여 상기 현재 비디오 블록과 상기 비트스트림 표현 간의 변환을 수행하는 단계를 포함하고, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 픽셀의 인트라 예측으로부터 양자화된 잔차 블록은 수평 또는 수직 방향과 다른 잔차 예측 방향에서 수행되는 차동 펄스 코딩 변조 표현을 사용하여 표현된다.

[0011] 또 다른 예에서, 또 다른 비디오 프로세싱 방법이 개시된다. 상기 방법은, 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환 중에 사용되는 차동 코딩 모드가 상기 현재 비디오 블록과 연관된 인트라 코딩 모드와 동일한지 결정하는 단계 및 상기 차동 코딩 모드의 구현 규칙을 사용하여 현재 비디오 블록과 비트스트림 표현 간의 변환을 수행하는 단계를 포함한다.

[0012] 또 다른 예에서, 비디오 프로세싱 장치가 개시된다. 상기 장치는 상기 개시된 방법을 수행하기 위해 구성된 프로세서를 포함한다.

[0013] 다른 예시적인 측면에서, 컴퓨터 판독 가능 프로그램 매체(computer readable program medium)가 개시된다. 상기 매체에는 위에서 설명한 방법의 프로세서 구현을 위한 코드가 저장되어 있다.

[0014] 이들 및 기타 측면은 본 문서에 설명되어 있다.

도면의 간단한 설명

- [0015] 도 1은 인트라 블록 카피를 도시한 도면이다.
- 도 2는 팔레트 모드에서 코딩된 블록의 예를 도시한다.
- 도 3은 팔레트 엔트리들을 시그널링하기 위한 팔레트 예측자의 사용 예를 도시한다.
- 도 4는 수평 및 수직 횡단(traverse) 스캔의 예를 도시한다.
- 도 5는 팔레트 인덱스의 코딩의 예를 도시한다.
- 도 6은 아핀 선형 가중 인트라 예측(affine linear weighted intra prediction)(ALWIP) 프로세스의 일 예를 도시한다.
- 도 7은 아핀 선형 가중 인트라 예측(affine linear weighted intra prediction)(ALWIP) 프로세스의 일 예를 도시한다.
- 도 8은 아핀 선형 가중 인트라 예측(affine linear weighted intra prediction)(ALWIP) 프로세스의 일 예를 도시한다.
- 도 9는 아핀 선형 가중 인트라 예측(affine linear weighted intra prediction)(ALWIP) 프로세스의 일 예를 도시한다.
- 도 10은 본 문서에 설명된 기술을 구현하는 데 사용되는 하드웨어 플랫폼의 예의 블록도이다.
- 도 11은 비디오 프로세싱 방법의 일 예에 따른 흐름도이다.
- 도 12는 4개의 병합 후보의 예를 도시한다.
- 도 13은 비디오 코딩에 사용되는 병합 후보 쌍(pair)들의 예를 도시한다.
- 도 14는 67개의 인트라 예측 모드의 예를 도시한다.
- 도 15는 본 개시의 기술을 활용할 수 있는 비디오 코딩 시스템의 일 예를 도시한 블록도이다.
- 도 16은 비디오 인코더의 일 예를 도시한 블록도이다.

도 17은 비디오 디코더의 일 예를 도시한 블록도이다.

도 18은 여기에 개시된 다양한 기술들이 구현될 수 있는 예시적인 비디오 프로세싱 시스템을 도시한 블록도이다.

도 19 및 도 20은 예시적인 비디오 프로세싱 방법의 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0016] 본 문서에서 사용된 섹션 제목은 이해를 쉽게 하기 위한 것으로서, 섹션에 개시된 실시예는 해당 섹션으로만 제한되지 않는다. 더욱이, 특정 실시예는 다목적(Versatile) 비디오 코딩 또는 기타 특정 비디오 코덱에 대한 참조로 설명되지만, 개시된 기술은 다른 비디오 코딩 기술에도 적용 가능하다. 더욱이, 일부 실시예는 비디오 코딩 단계를 자세히 설명하는 반면, 디코더에 의해 코딩을 취소하는 대응하는 단계가 구현될 것으로 이해된다. 또한 비디오 프로세싱이라는 용어에는 비디오 코딩 또는 압축, 비디오 디코딩 또는 압축해제 및 비디오 트랜스 코딩을 포함하며, 비디오 픽셀은 하나의 압축된 형식에서 다른 압축된 형식으로 또는 다른 압축된 비트레이트로 표현된다.

[0017] 요약

[0018] 이 특허 문서는 비디오 코딩 기술과 관련이 있다. 구체적으로, 이것은 비디오 코딩에서 DPCM 코딩과 관련이 있다. HEVC와 같은 기존 비디오 코딩 표준 또는 마무리될 표준(다목적 비디오 코딩(Versatile Video Coding))에 적용될 수 있다. 미래의 비디오 코딩 표준 또는 비디오 코덱에도 적용될 수 있다.

[0019] 초기 논의

[0020] 비디오 코딩 표준은 주로 잘 알려진 ITU-T 및 ISO/IEC 표준의 개발을 통해 발전해 왔다. ITU-T는 H.261 및 H.263을 제작했으며, ISO/IEC는 MPEG-1 및 MPEG-4 비주얼을 제작했으며, 두 조직은 H.262/MPEG-2 비디오 및 H.264/MPEG-4 고급 비디오 코딩(AVC) 및 H.265/HEVC[1] 표준을 공동 제작했다. H.262 이후, 비디오 코딩 표준은 시간 예측과 변환 코딩이 사용되는 하이브리드 비디오 코딩 구조에 기초한다. HEVC를 넘어선 미래의 비디오 코딩 기술을 탐구하기 위해, 2015년에 VCEG와 MPEG가 공동으로 조인트 비디오 연구 팀(Joint Video Exploration Team)(JVET)을 설립했다. 그 이후로, JVET에 의해 많은 새로운 방법이 채택되었고, 공동 탐사 모델 (JEM) [3,4]라는 레퍼런스(reference) 소프트웨어에 적용되었다. 2018년 4월, VCEG(Q6/16)와 ISO/IEC JTC1 SC29/WG11(MPEG) 사이의 공동 비디오 전문가 팀(Joint Video Expert Team)(JVET)은 HEVC에 비해 50% 비트 전송률 감소를 목표로 하는 VVC 표준 작업을 위해 만들어졌다.

[0021] The latest version of VVC draft, i.e., Versatile Video Coding (Draft 4) could be found at: VVC 드래프트의 최신 버전, 즉 다목적 비디오 코딩(Versatile Video Coding)(드래프트(Draft) 4)은 다음에서 찾을 수 있다.

[0022] phenix.it-sudparis.eu/jvet/doc_end_user/current_document.php?id=5755

[0023] The latest reference software of VVC, named VTM, could be found at:

[0024] vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/tags/VTM-4.0

[0025] 2.1 인트라 블록 카피

[0026] 일명, 현재 픽처 레퍼런싱(picture referencng)라고 하는 IBC(인트라 블록 카피)는 HEVC 스크린 콘텐츠 코딩 확장(HEVC-SCC) 및 현재 VVC 테스트 모델(VTM-4.0)에 채택되었다. IBC는 모션 보상의 개념을 프레임 간 코딩에서 프레임 내 코딩으로 확장한다. 도 1에 도시된 바와 같이, 현재 블록은 IBC가 적용될 때 동일한 픽처 내의 레퍼런스 블록(reference block)에 의해 예측된다. 레퍼런스 블록의 샘플은 현재 블록이 코딩 되거나 디코딩 되기 전에 이미 재구성되어 있어야 한다. IBC는 대부분의 카메라 캡처 시퀀스에 대해 그다지 효율적이지 않지만 화면 콘텐츠에 대해 상당한 코딩 이득을 보여준다. 그 이유는 화면 콘텐츠 픽처에서 아이콘, 텍스트 문자 등 반복되는 패턴이 많기 때문이다. IBC는 이러한 반복 패턴 사이의 중복성을 효과적으로 제거할 수 있다. HEVC-SCC에서 CU(인tra 코딩된 코딩 유닛)는 현재 픽처를 레퍼런스 픽처로 선택하는 경우 IBC를 적용할 수 있다. MV는 블록 벡터(BV)로 이름이 바뀌고, BV는 항상 정수(integer) 픽셀 정밀도(precision)를 갖는다. 메인 프로파일 HEVC와 호환되도록, 현재 픽처는 디코딩 픽처 버퍼(Decoded Picture Buffer; DPB)에서 "장기(long-term)" 레퍼런스 픽처로 표시된다. 유사하게, 다중 시점/3D 비디오 코딩 표준에서 시점내(inter-view) 레퍼런스 픽처

는 "장기" 레퍼런스 픽처로도 표시된다는 점에 유의해야 한다.

- [0027] BV를 따라 그의 레퍼런스 블록을 찾고, 레퍼런스 블록을 카피하여 예측을 생성할 수 있다. 잔차는 원래 시그널링에서 레퍼런스 픽셀을 감산하여 얻을 수 있다. 그 다음, 변환 및 양자화는 다른 코딩 모드에서와 같이 적용될 수 있다.
- [0028] 도 1은 인트라 블록 카피를 도시한다.
- [0029] 그러나, 레퍼런스 블록이 픽처 외부에 있거나 현재 블록과 중첩되거나 재구성(reconstructed) 영역 외부 또는 일부 제약 조건(constrain)에 의해 제한된 유효 영역 외부에 있는 경우에는, 일부 또는 전체 픽셀 값이 정의되지 않는다. 기본적으로 이러한 문제를 처리하는 두 가지 솔루션이 있다. 하나는 예를 들어, 비트스트림 적합성(conformance)에서 그러한 상황을 허용하지 않는 것이다. 다른 하나는 정의되지 않은 픽셀 값에 패딩(padding)을 적용하는 것이다. 다음 서브 섹션에서는 솔루션에 대해 자세히 설명한다.
- [0030] 2.2 HEVC 화면 콘텐츠 코딩 확장에서의 IBC
- [0031] HEVC의 화면 콘텐츠 코딩 확장에서 블록이 현재 픽처를 레퍼런스로 사용할 때 다음 사양 텍스트에 표시된 대로 전체 레퍼런스 블록이 사용 가능한 재구성 영역 내에 있음을 보장해야 한다.
- [0032] 변수 offsetX 및 offsetY는 다음과 같이 유도된다:
- [0033]
$$\text{offsetX} = (\text{ChromaArrayType} = 0) ? 0 : (\text{mvCLX}[0] \& 0x7 ? 2 : 0) \quad (8-106)$$
- [0034]
$$\text{offsetY} = (\text{ChromaArrayType} = 0) ? 0 : (\text{mvCLX}[1] \& 0x7 ? 2 : 0) \quad (8-107)$$
- [0035] 레퍼런스 픽처가 현재 픽처일 때, 루마 모션 벡터 mvLX가 다음 제약 조건을 따라야 하는 것은 비트 스트림 적합성의 요구 사항이다:
- [0036] - 6.4.1 절에 규정된 z-스캔 차수 블록 가용성을 위한 유도 프로세스가 (xCurr, yCurr) 이 (xCb, yCb)와 동일하고 인접 루마 위치 (xNbY, yNbY)가 (xPb + (mvLX[0] >> 2) - offsetX, yPb + (mvLX[1] >> 2) - offsetY)와 입력으로서 동일하게 세팅되어 개시되는 경우, 출력은 참이어야 한다.
- [0037] - 6.4.1 절에 규정된 z-스캔 차수 블록 가용성을 위한 유도 프로세스가 (xCurr, yCurr) 이 (xCb, yCb)와 동일하고 인접 루마 위치 (xNbY, yNbY)가 (xPb + (mvLX[0] >> 2) + nPbW - 1 + offsetX, yPb + (mvLX[1] >> 2) + nPbH - 1 + offsetY)와 입력으로서 동일하게 세팅되어 개시되는 경우, 출력은 참이어야 한다.
- [0038] - 다음의 조건 중 하나 또는 둘 모두는 참이어야 한다:
- [0039] - The value of (mvLX[0] >> 2) + nPbW + xB1 + offsetX는 0보다 작거나 같다.
- [0040] -The value of (mvLX[1] >> 2) + nPbH + yB1 + offsetY는 0보다 작거나 같다.
- [0041] -다음의 조건들은 참이어야 한다:
- [0042]
$$\frac{(xPb + (mvLX[0] \gg 2) + nPbSw - 1 + offsetX)}{CtbSizeY} - xCurr / CtbSizeY \leq \frac{yCurr}{CtbSizeY} - \frac{(yPb + (mvLX[1] \gg 2) + nPbSh - 1 + offsetY)}{CtbSizeY} \quad (8-108)$$
- [0043] 따라서, 레퍼런스 블록(reference block)이 현재 블록과 겹치거나 레퍼런스 블록이 픽처 외부에 있는 경우는 발생하지 않는다. 레퍼런스 또는 예측 블록(prediction block)을 패딩(padding)할 필요가 없다.
- [0044] 2.3 VVC 테스트 모델의 IBC
- [0045] 현재 VVC 테스트 모델, 즉 VTM-4.0 설계에서 전체 레퍼런스 블록은 현재 CTU(코딩 트리 유닛)와 함께 있어야 하며 현재 블록과 겹치지 않아야 한다. 따라서 레퍼런스 또는 예측 블록을 패딩(padding)할 필요가 없다. IBC 플래그는 현재 CU의 예측 모드로 코딩된다. 따라서, 각 CU에 대해 MODE_INTRA, MODE_INTER 및 MODE_IBC의 세 가지 예측 모드가 있다.
- [0046] 2.3.1 IBC 병합 모드
- [0047] IBC 병합 모드(merge mode)에서 IBC 병합 후보 목록의 항목을 가리키는 인덱스는 비트스트림에서 파싱된다. IBC 병합 목록의 구성은 다음 단계 순서에 따라 요약할 수 있다:
- [0048] * 1단계: 공간 후보 유도

- [0049] * 2단계: HMVP 후보 삽입
- [0050] * 3단계: 페어관련(pair-wise) 평균 후보 삽입
- [0051] 공간적 병합 후보의 유도에서, 도 12에 도시된 위치에 위치한 후보들 중에서 최대 4 개의 병합 후보가 선택된다. 유도 순서는 A1, B1, B0, A0 및 B2이다. 위치 B2는 위치 A1, B1, B0, A0의 PU를 사용할 수 없는 경우(예를 들어, 다른 슬라이스 또는 타일에 속하기 때문에) 또는 IBC 모드로 코딩 되지 않은 경우에만 고려된다. A1 위치의 후보가 추가된 후 나머지 후보의 삽입은 중복 확인(redundancy check)를 거쳐 동일한 모션 정보를 가진 후보를 목록에서 제외하여 코딩 효율성을 향상시킨다. 계산 복잡성을 줄이기 위해 언급된 중복성 검사(redundancy check)에서 가능한 모든 후보 페어를 고려하는 것은 아니다. 대신 도 13에서 화살표와 연결된 페어만 고려되고, 중복 검사에 사용되는 대응하는 후보가 동일한 모션 정보가 없는 경우에만 후보가 리스트(list)에 추가된다.
- [0052] 공간 후보를 삽입한 후 IBC 병합 목록 크기가 최대 IBC 병합 목록 크기보다 여전히 작은 경우 HMVP 테이블의 IBC 후보를 삽입할 수 있다.
- [0053] HMVP 후보를 삽입할 때 중복 확인이 수행된다.
- [0054] 마지막으로, 페어관련(pair-wise) 평균 후보가 IBC 병합 목록에 삽입된다.
- [0055] 병합 후보에 의해 식별된 레퍼런스 블록이 픽처 외부에 있거나 현재 블록과 중첩되거나 재구성된 영역 외부 또는 일부 제약 조건에 의해 제한된 유효 영역 외부에 있는 경우 병합 후보를 **유효하지 않은 병합 후보**(invalid merge candidate)라고 한다.
- [0056] 유효하지 않은 병합 후보가 IBC 병합 목록에 삽입될 수 있음에 주의한다.
- [0057] 2.3.2 IBCAMVP 모드
- [0058] IBC AMVP 모드에서 IBC AMVP 목록의 항목에 대한 AMVP 인덱스 포인트는 비트스트림에서 파싱(parsing)된다. IBC AMVP 목록의 구성은 다음 단계 순서에 따라 요약될 수 있다.
- [0059] * 1단계: 공간(spatial) 후보 유도
- [0060] 사용 가능한 후보를 찾을 때까지 A0, A1을 확인.
- [0061] 사용 가능한 후보를 찾을 때까지 B0, B1, B2를 확인.
- [0062] * 2 단계: HMVP 후보 삽입
- [0063] * 3 단계: 제로(zero) 후보 삽입
- [0064] 공간 후보를 삽입한 후, IBC AMVP 목록 크기가 최대 IBC AMVP 목록 크기보다 여전히 작은 경우 HMVP 테이블의 IBC 후보가 삽입될 수 있다.
- [0065] 마지막으로 IBC AMVP 목록에 제로(zero) 후보가 삽입된다.
- [0066] 2.4 팔레트 모드
- [0067] 팔레트 모드의 기본 아이디어는 CU의 샘플이 대표 컬러 값의 작은 세트로 표현된다는 것이다. 이 세트를 팔레트라고 한다. 이스케이프 기호(escape symbol) 뒤에 (양자화된) 컴포넌트 값이 오는 시그널링 함으로써 팔레트 외부에 있는 샘플을 나타내는 것도 가능하다. 이것은 도 2에 도시되어 있다.
- [0068] 2.5 HEVC 화면 콘텐츠 코딩 확장(HEVC-SCC)의 팔레트 모드
- [0069] HEVC-SCC의 팔레트 모드에서 예측 방식은 팔레트와 인덱스 맵을 코딩하는 데 사용된다.
- [0070] 2.5.1 팔레트 엔트리(entries)의 코딩
- [0071] 팔레트 엔트리의 코딩을 위해 팔레트 예측자(predictor)가 유지된다. 팔레트 예측자(palette predictor)와 팔레트의 최대 크기는 SPS 내에서 시그널링 된다. HEVC-SCC에서는 PPS에 palette_predictor_initializer_present_flag가 도입된다. 이 플래그가 1이면 팔레트 예측자를 초기화하기 위한 항목이 비트스트림에서 시그널링 된다. 팔레트 예측자는 각 CTU 행, 각 슬라이스 및 각 타일의 시작 부분에서 초기화된다. palette_predictor_initializer_present_flag의 값에 따라, 팔레트 예측자는 0으로 재설정되

거나 PPS에서 시그널링 된 팔레트 예측자 초기화 엔트리를 사용하여 초기화된다. HEVC-SCC에서 크기 0의 팔레트 예측자 초기화가 인에이블되어 PPS 레벨에서 팔레트 예측자 초기화를 명시적으로 디스에이블할 수 있다.

[0072] 팔레트 예측자의 각 엔트리에 대해 재사용 플래그가 시그널링되어 현재 팔레트의 일부인지 여부를 나타낸다. 이것은 도 3에 도시되어 있다. 재사용 플래그는 0의 실행 길이(run-length) 코딩을 사용하여 전송된다. 그 후, 새로운 팔레트 항목의 수는 차수 0의 지수 골롬(Golomb) 코드를 사용하여 시그널링 된다. 마지막으로, 새 팔레트 항목에 대한 컴포넌트 값이 시그널링 된다.

[0073] 2.5.2 팔레트 지수 코딩

[0074] 팔레트 인덱스는 도 4에 도시된 바와 같이, 수평 및 수직 횡단 스캔(traverse scan)을 사용하여 코딩 된다. 스캔 순서는 palette_transpose_flag를 사용하여 비트스트림에서 명시적으로 시그널링 된다. 나머지 서브 섹션에서는 스캔이 수평이라고 가정한다.

[0075] 팔레트 인덱스는 두 가지 주요 팔레트 샘플 모드를 사용하여 코딩된다: 'INDEX' 및 'COPY_ABOVE'. 앞서 설명한 것처럼, 이스케이프 기호도 'INDEX' 모드로 시그널링 되고 최대 팔레트 크기와 동일한 인덱스가 할당된다. 모드는 맨 위쪽 행 또는 이전 모드가 'COPY_ABOVE'인 경우를 제외하고 플래그를 사용하여 시그널링 된다. 'COPY_ABOVE' 모드에서는 위쪽 행에 있는 샘플의 팔레트 인덱스가 카피된다. 'INDEX' 모드에서 팔레트 인덱스는 명시적으로 시그널링 된다. 'INDEX' 및 'COPY_ABOVE' 모드 모두에 대해, 동일한 모드를 사용하여 코딩되는 후속 샘플의 수를 지정(specify)하는 실행 값(run value)이 시그널링 된다. 이스케이프 기호가 'INDEX' 또는 'COPY_ABOVE' 모드에서 실행의 일부인 경우, 이스케이프 컴포넌트 값(escape component value)이 각 이스케이프 기호를 위해 시그널링 된다. 팔레트 인덱스의 코딩은 도 5에 도시되어 있다.

[0076] 이러한 선택스(syntax) 순서는 다음과 같이 수행된다. 먼저 CU에 대한 인덱스 값의 수가 시그널링 된다. 그 다음에는 절단된(truncated) 이진 코딩을 사용하여 전체 CU에 대한 실제 인덱스 값의 시그널링이 이어진다. 인덱스 수와 인덱스 값 모두 바이패스 모드에서 코딩된다. 인덱스 관련 바이패스(bypass) 빈(bins)을 함께 그룹화한다. 그런 다음, 팔레트 샘플 모드(필요한 경우)와 실행이 보간(interleave) 방식으로 시그널링 된다. 마지막으로, 전체 CU에 대한 이스케이프 샘플에 해당하는 컴포넌트 이스케이프 값이 함께 그룹화되고 바이패스 모드에서 코딩된다.

[0077] 추가 선택스 요소인 last_run_type_flag는 인덱스 값을 시그널링 한 후 시그널링 된다. 이 선택스 요소는, 인덱스 수와 함께, 블록 내 마지막 실행에 대응하는 실행 값을 시그널링할 필요를 없애 준다.

[0078] HEVC-SCC에서 팔레트 모드는 4:2:2, 4:2:0 및 모노크롬 크로마 형식(format)에 대해서도 인에이블된다. 팔레트 항목 및 팔레트 인덱스의 시그널링은 모든 크로마 형식에 대해 거의 동일하다. 비-모노크롬(non-monochrome) 형식인 경우, 각 팔레트 항목은 3개의 컴포넌트로 구성된다. 모노크롬 형식에 대해, 각 팔레트 항목은 단일 컴포넌트로 구성된다. 서브 샘플링된 크로마 방향(directions)의 경우, 크로마 샘플은 2로 나눌 수 있는 루마 샘플 인덱스와 연관된다. CU에 대한 팔레트 인덱스를 재구성한 후, 샘플이 자신과 연관된 단일 컴포넌트만을 가지는 경우, 팔레트 항목의 제1 컴포넌트만 사용된다. 시그널링 상의 유일한 차이점은 이스케이프 컴포넌트 값에 대한 것이다. 각 이스케이프 샘플에 대해, 시그널링 되는 이스케이프 컴포넌트 값의 수는 해당 샘플과 연관된 컴포넌트의 수에 따라 다를 수 있다.

[0079] 2.6 변환 스킵 모드의 계수 코딩

[0080] JVET-M0464 및 JVET-N0280에서, 변환 스킵 레벨의 통계 및 시그널링 특성에 대해 잔차 코딩을 적용하기 위해 변환 스킵(Transform Skip; TS) 모드의 계수 코딩에 대한 몇 가지 수정이 제안된다.

[0081] 제안된 수정 사항은 다음과 같다.

[0082] **최종 중요 스캐닝 위치 없음:** 잔차 시그널링은 예측 후 공간 잔차를 반영하고 TS에 대해 변환에 의한 에너지 압축이 수행되지 않기 때문에, 변환 블록의 오른쪽 하단 모서리에서 후행(trailing) 0 또는 중요하지 않은 레벨에 대한 더 높은 확률은 더 이상 제공되지 않는다. 따라서, 이 경우 최종 중요 스캐닝 위치 시그널링은 생략된다. 대신, 프로세싱되는 제1 서브 블록은 변환 블록 내에서 가장 아래 오른쪽 서브 블록이다.

[0083] **서브 블록 CBF:** 최종 중요 스캐닝 위치 시그널링의 부재로 인해, TS에 대한 coded_sub_block_flag로 시그널링되는 서브 블록 CBF가 다음과 같이 수정되어야 한다:

[0084] * 양자화로 인해, 전술한 중요도가 낮은(insignificance) 시퀀스는 변환 블록 내부에서 여전히 국부적으로 발생

할 수 있다. 따라서, 앞서 설명한 바와 같이, 최종 중요 스캐닝 위치가 제거되고 모든 서브 블록에 대해 coded_sub_block_flag가 코딩된다.

- [0085] * DC 주파수 위치(왼쪽 상단 서브 블록)를 커버하는 서브블록에 대한 coded_sub_block_flag는 특별한 경우를 나타낸다. VVC Draft 3에서, 이 서브 블록에 대한 coded_sub_block_flag는 시그널링 되지 않고, 항상 1과 같도록 추론(infer)된다. 최종 중요 스캐닝 위치가 다른 서브 블록에 위치하는 경우, DC 서브 블록 외부에 적어도 하나의 중요(significant) 레벨이 있음을 의미한다. 결과적으로, DC 서브 블록은 이 서브 블록에 대한 coded_sub_block_flag가 1과 동일한 것으로 추론되더라도, 단지 0/중요도가 없는 레벨만을 포함할 수 있다. TS에 최종 스캐닝 위치 정보가 없으면, 각 서브블록에 대한 coded_sub_block_flag가 시그널링 된다. 이것은 또한 다른 모든 coded_sub_block_flag 선택스 요소가 이미 0과 같은 경우를 제외하고 DC 서브 블록에 대한 coded_sub_block_flag를 포함한다. 이 경우, DC coded_sub_block_flag는 1과 동일한 것으로 추론된다 (inferDcSbCbf=1). 이 DC 서브 블록에는 적어도 하나의 유효 레벨이 있어야 하므로, (0,0)에서 첫 번째 위치에 대한 sig_coeff_flag 선택스 요소는 시그널링 되지 않고 이 DC 서브 블록의 다른 모든 sig_coeff_flag 선택스 요소가 0과 동일한 경우 대신 1과 동일하도록 유도(derive)된다(inferSbDcSigCoeffFlag=1).
- [0086] * coded_sub_block_flag에 대한 컨텍스트 모델링이 변경된다. 컨텍스트 모델 인덱스는 왼쪽에 있는 coded_sub_block_flag와 현재 서브블록 위에 있는 coded_sub_block_flag의 합과 둘 다의 논리적 분리로 계산된다.
- [0087] **sig_coeff_flag 컨텍스트 모델링:** sig_coeff_flag 컨텍스트 모델링의 로컬 템플릿은 현재 스캐닝 위치의 왼쪽 인접(NB0)과 위쪽 인접(NB1)만 포함하도록 수정된다. 컨텍스트 모델 오프셋은 중요한 인접 위치 sig_coeff_flag[NB0] + sig_coeff_flag[NB1]의 수이다. 따라서, 현재 변환 블록 내의 대각선을 따른 다른 컨텍스트 세트의 선택이 제거된다. 그 결과로 세 개의 컨텍스트 모델과 sig_coeff_flag 플래그를 코딩하기 위한 단일 컨텍스트 모델 세트가 생성된다.
- [0088] **abs_level_gt1_flag 및 par_level_flag 컨텍스트 모델링:** abs_level_gt1_flag 및 par_level_flag에 대해 단일 컨텍스트 모델이 채용(employ)된다.
- [0089] **abs_remainder 코딩:** 변환 스킵 잔차 절대 레벨의 경험적 분포는 일반적으로 여전히 라플라시안 또는 기하학적 분포에 적합하지만, 변환 계수 절대 레벨보다 더 큰 비정상성이 존재한다. 특히, 연속 실현(consecutive realization)의 윈도우(window) 내 분산(variance)은 잔차 절대 레벨에 대해 더 높다. 이것은 abs_remainder 선택스 이진화 및 컨텍스트 모델링의 다음 수정을 야기한다:
- [0090] * 이진화(binanzation)에서 더 높은 컷오프 값, 즉 sig_coeff_flag, abs_level_gt1_flag, par_level_flag 및 abs_level_gt3_flag를 사용한 코딩으로부터 abs_remainder를 위한 라이스 코드(Rice code)로의 트랜지션 포인트와 각 빈(bin) 위치에 대한 전용 컨텍스트 모델을 사용하여, 더 높은 압축 효율을 얻을 수 있다. 컷오프를 늘리면 컷오프에 도달할 때까지 abs_level_gt5_flag, abs_level_gt7_flag 등을 도입하는 등 "X보다 큼" 플래그가 생성된다. 컷오프 자체는 5로 고정된다(numGtFlags=5)
- [0091] * 라이스 파라미터 유도를 위한 템플릿이 수정되고, 즉, 왼쪽에 있는 인접과 현재 스캐닝 위치 위에 있는 인접만이 sig_coeff_flag 컨텍스트 모델링을 위한 로컬 템플릿과 유사한 것으로 간주된다.
- [0092] **coeff_sign_flag 컨텍스트 모델링:** 기호(signs)의 시퀀스 내부의 비정상성과 예측 잔차가 종종 바이어스(biased)된다는 사실로 인해, 전역(global) 경험적 분포가 거의 균일하게 분포된 경우에도 컨텍스트 모델을 사용하여 기호를 코딩할 수 있다. 단일 전용 컨텍스트 모델이 기호 코딩에 사용되며 기호는 sig_coeff_flag 다음에 파싱되어 모든 컨텍스트 코딩된 빈(bins)을 함께 유지한다.
- [0093] 2.7 양자화 잔차 블록 차동 펄스 부호 변조(QR-BDPCM)
- [0094] JVET-M0413에서는 화면 콘텐츠를 효율적으로 코딩 하기 위해 양자화 잔차 블록 차동 펄스 코드 변조(QR-BDPCM)가 제안된다.
- [0095] QR-BDPCM에서 사용되는 예측 방향은 수직 및 수평 예측 모드일 수 있다. 인트라 예측은 인트라 예측과 유사한 예측 방향(수평 또는 수직 예측)으로 샘플 카피함으로써 전체 블록에 대해 수행된다. 잔차는 양자화되고 양자화된 잔차와 그 예측자(수평 또는 수직) 양자화 값 사이의 델타가 코딩된다. 다음으로 설명될 수 있다: 크기가 M(행) X N(열)인 블록의 경우, 위쪽 또는 왼쪽 블록 경계 샘플에서 필터링 되지 않은 샘플을 사용하여, 인트라 예측을 수평(예측된 블록을 가로질러 왼쪽 인접 픽셀 값을 라인 단위로 카피) 또는 수직으로(예측된 블록

의 각 라인에 상단 인접 라인 카피) 수행한 이후 $r_{i,j}$, $0 \leq i \leq M-1$, $0 \leq j \leq N-1$ 을 예측 잔차로 둔다. $Q(r_{i,j})$, $0 \leq i \leq M-1$, $0 \leq j \leq N-1$ 이 잔차 $r_{i,j}$ 의 양자화된 버전을 나타내고, 여기서 잔차는 원래 블록과 예측된 블록 값 간

$$\tilde{r}_{i,j}$$

의 차이이다. 그런 다음 블록 DPCM이 양자화된 잔차 샘플에 적용되어 요소 $\tilde{r}_{i,j}$ 을 갖는 수정된 $M \times N$ 어레이

$$\tilde{R}$$

이 생성된다. 수직 BDPCM이 시그널링 될 때:

[0096]

$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & i = 0, \quad 0 \leq j \leq (N - 1) \\ Q(r_{i,j}) - Q(r_{(i-1),j}), & 1 \leq i \leq (M - 1), \quad 0 \leq j \leq (N - 1) \end{cases} \quad (2-7-1)$$

[0097]

수평 예측의 경우 유사한 규칙이 적용되고 잔차 양자화된 샘플은 다음과 같이 획득된다.

[0098]

$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & 0 \leq i \leq (M - 1), \quad j = 0 \\ Q(r_{i,j}) - Q(r_{i,(j-1)}), & 0 \leq i \leq (M - 1), \quad 1 \leq j \leq (N - 1) \end{cases} \quad (2-7-2)$$

$$\tilde{r}_{i,j}$$

[0099]

잔차 양자화된 샘플은 디코더로 전송된다.

[0100]

디코더 측에서는, 위의 계산을 반대로 하여

$$Q(r_{i,j}), \quad 0 \leq i \leq M - 1, \quad 0 \leq j \leq N - 1$$

를 생성한다. 수직 예측 케이스의 경우,

[0101]

$$Q(r_{i,j}) = \sum_{k=0}^i \tilde{r}_{k,j}, \quad 0 \leq i \leq (M - 1), \quad 0 \leq j \leq (N - 1) \quad (2-7-3)$$

[0102]

수평 케이스의 경우,

[0103]

$$Q(r_{i,j}) = \sum_{k=0}^j \tilde{r}_{i,k}, \quad 0 \leq i \leq (M - 1), \quad 0 \leq j \leq (N - 1) \quad (2-7-4)$$

$$Q^{-1}(Q(r_{i,j}))$$

[0104]

역 양자화된 잔차, 는 재구성된 샘플 값을 생성하기 위해 인트라 블록 예측 값에

추가된다.

[0105] 이 방식의 주요 이점은, 계수가 과싱되면서 계수의 과싱 과정에서 예측자를 추가하여 역 DPCM이 즉석에서 이루어질 수도 있고, 이것이 과싱 이후에 수행될 수도 있다는 점이다.

[0106] QR-BDPCM의 드레프트 텍스트 변경 사항은 다음과 같다.

[0107] 7.3.6.5 코딩 유닛 선택스

	Descriptor
coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	
if(tile_group_type != 1 sps_IBC_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA)	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] == 0 && tile_group_type != 1)	
pred_mode_flag	ae(v)
if(((tile_group_type == 1 && cu_skip_flag[x0][y0] == 0) (tile_group_type != 1 && CuPredMode[x0][y0] != MODE_INTRA)) && sps_IBC_enabled_flag)	
pred_mode_IBC_flag	ae(v)
}	
if(CuPredMode[x0][y0] == MODE_INTRA) {	
if(pred_mode_flag == MODE_INTRA && (cldx == 0) && (cbWidth <= 32) && (cbHeight <= 32)) {	
bdpcm_flag[x0][y0]	ae(v)
if(bdpcm_flag[x0][y0]) {	
bdpcm_dir_flag[x0][y0]	ae(v)
}	
else {	
if(sps_pcm_enabled_flag && cbWidth >= MinIpcmCbSizeY && cbWidth <= MaxIpcmCbSizeY && cbHeight >= MinIpcmCbSizeY && cbHeight <= MaxIpcmCbSizeY)	
pcm_flag[x0][y0]	ae(v)
if(pcm_flag[x0][y0]) {	
while(!byte_aligned())	
pcm_alignment_zero_bit	f(1)
pcm_sample(cbWidth, cbHeight, treeType)	
} else {	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
if((y0 % CtbSizeY) > 0)	
intra_luma_ref_idx[x0][y0]	ae(v)
if(intra_luma_ref_idx[x0][y0] == 0 && (cbWidth <= MaxTbSizeY cbHeight <= MaxTbSizeY) && (cbWidth * cbHeight > MinTbSizeY * MinTbSizeY))	
intra_subpartitions_mode_flag[x0][y0]	ae(v)
if(intra_subpartitions_mode_flag[x0][y0] == 1 && cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY)	
intra_subpartitions_split_flag[x0][y0]	ae(v)
if(intra_luma_ref_idx[x0][y0] == 0 && intra_subpartitions_mode_flag[x0][y0] == 0)	
intra_luma_mpm_flag[x0][y0]	ae(v)
if(intra_luma_mpm_flag[x0][y0])	
intra_luma_mpm_idx[x0][y0]	ae(v)
else	
intra_luma_mpm_remainder[x0][y0]	ae(v)
}	
}	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA)	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
...	
}	

[0108]

[0109] **bdpcm_flag[x0][y0]** 이 1과 같다는 것은 위치 (x0, y0)에서 루마 코딩 블록을 포함하는 코딩 유닛 내에 specifies that a **bdpcm_dir_flag**가 있다는 것이다.

[0110] **bdpcm_dir_flag**[x0][y0]이 0과 같다는 것은 bdpcm 블록에서 사용되는 예측 방향이 수직이고, 다른 경우는 수직이라는 것이다.

[0111] 8.4.2 루마 인트라 예측 모드를 위한 유도 프로세스

[0112] 이 프로세스의 입력은 다음과 같다:

[0113] - 현재 픽처의 왼쪽 상단 루마 샘플에 대한 현재 코딩 블록의 왼쪽 상단 샘플을 지정하는 루마 위치(xCb, yCb),

[0114] - 루마 샘플에서 현재 코딩 블록의 너비를 지정하는 변수 cbWidth,

[0115] - 루마 샘플에서 현재 코딩 블록의 높이를 지정하는 변수 cbHeight.

[0116] 이 프로세스에서, 루마 인트라 예측 모드 IntraPredModeY[xCb][yCb]가 유도된다.

[0117] 표 8-1은 인트라 예측 모드 IntraPredModeY[xCb][yCb]의 값 및 연관된 이름을 지정한다.

[0118] 표 8-1 인트라 예측 모드 및 연관된 이름의 사양

인트라 예측 모드	연관된 이름
0	INTRA_PLANAR
1	INTRA_DC
2..66	INTRA_ANGULAR2..INTRA_ANGULAR66
81..83	INTRA_LT_CCLM, INTRA_L_CCLM, INTRA_T_CCLM

[0119]

[0120] 노트 : 인트라 예측 모드 INTRA_LT_CCLM, INTRA_L_CCLM 및 INTRA_T_CCLM는 크로마 컴포넌트들에 대해서만 적용 가능하다.

[0121] IntraPredModeY[xCb][yCb]는 다음 순서의 단계들에 의해 유도된다:

[0122] - 인접 위치 (xNbA, yNbA) 및 (xNbB, yNbB)는 각각 (xCb - 1, yCb + cbHeight - 1) 및 (xCb + cbWidth - 1, yCb - 1)로 설정된다.

[0123] - A 또는 B 중의 어느 하나로 대체되는 X에 대해, 변수 candIntraPredModeX는 다음과 같이 유도된다:

[0124] ...

[0125] - 변수 ispDefaultMode1 및 ispDefaultMode2는 아래와 같이 정의된다:

[0126] ...

[0127] - candModeList[x] with x = 0..5 는 아래와 같이 유도된다:

[0128] ...

[0129] - IntraPredModeY[xCb][yCb]는 다음의 프로시저를 적용하여 유도된다:

[0130] - bdpcm_flag[xCb][yCb]이 1과 같으면, IntraPredModeY[xCb][yCb]는 candModeList[0]와 같도록 설정된다.

[0131] - 그렇지 않고 intra_luma_mpm_flag[xCb][yCb]이 1과 같으면, IntraPredModeY[xCb][yCb]는 candModeList[intra_luma_mpm_idx[xCb][yCb]]와 같도록 설정된다.

[0132] - 그렇지 않으면, IntraPredModeY[xCb][yCb]는 다음 순서의 단계를 적용하여 유도된다.

[0133] ...

[0134] x = xCb..xCb + cbWidth - 1 및 y = yCb..yCb + cbHeight - 1인 변수 IntraPredModeY[x][y]는 IntraPredModeY[xCb][yCb]와 같도록 설정된다.

[0135] 2.8 매트릭스 기반 인트라 예측(MIP)

[0136] 매트릭스 기반 인트라 예측은 어떤 선형 가중 인트라 예측(affine linear weighted intra prediction; ALWIP)라고도 명명되고, 이것은 인트라 예측 시그널을 유도하는데 가중 매트릭스를 사용한다.

[0137] 2.8.1 방법에 대한 설명

$$W \quad H$$

[0138] 폭 및 높이 의 직사각형 블록의 샘플을 예측하기 위해, 어떤 선형 가중 인트라 예측(ALWIP)은 블록

$$W$$

의 왼쪽에 있는 재구성된 인접 경계 샘플 H 의 한 라인과 블록 위에 있는 재구성된 인접 경계 샘플 의 한 라인을 입력으로 취한다. 재구성된 샘플을 사용할 수 없는 경우, 종래의 인트라 예측에서 수행되던 방법으로 재구성된 샘플들이 생성된다.

[0139] 예측 신호의 생성은 다음 세 단계를 기반으로 한다:

$$W = H = 4$$

[0140] 1. 경계 샘플 중에서, 인 경우의 4개의 샘플과, 다른 모든 경우의 8개의 샘플이 평균화(averaging)에 의해 추출된다.

[0141] 2. 매트릭스 벡터 곱셈, 그 후 오프셋의 덧셈이 입력으로서 평균화된 샘플들에 대해 수행된다. 결과는, 원래 블록 내의 샘플들의 서브 샘플링된 세트 상에 감소된 예측 시그널이다.

[0142] 3. 나머지 위치에서의 예측 시그널은 선형 보간(linear interpolation)에 의해 설정된 서브 샘플 상의 예측 신호로부터 생성되고, 여기에서 선형 보간은 각 방향에서의 단일 단계 선형 보간이다.

$$S_0 \quad S_1, S_2$$

[0143] 예측 시그널을 생성하는 데 필요한 매트릭스 및 오프셋 벡터는 매트릭스의 세 세트 , 로부터

$$S_0 \quad A_0^i, i \in \{0, \dots, 17\}$$

취한다. 이 세트 은 18개의 매트릭스 로 구성되고, 각각의 매트릭

$$b_0^i, i \in \{0, \dots, 17\}$$

스는 16개의 행과 4개의 열, 및 18개의 각각의 크기 16인 오프셋 벡터

$$4 \times 4$$

$$S_1$$

를 포함한다. 이러한 세트의 매트릭스 및 오프셋 벡터는 크기 의 블록에 사용된다. 세트

$$A_1^i, i \in \{0, \dots, 9\}$$

는 10개의 매트릭스 을 포함하고, 각 매트릭스는 16개의 행과 8개의 열 및

$$b_1^i, i \in \{0, \dots, 9\}$$

각각의 크기 16인 10개의 오프셋 벡터 을 포함한다. 이러한 세트의 매트릭

$$4 \times 8 \quad 8 \times 4 \quad 8 \times 8$$

스 및 오프셋 벡터는 크기 , 및 인 블록에 사용된다. 마지막으로, 세트

$$S_2 \quad A_2^i, i \in \{0, \dots, 5\}$$

는 6개의 매트릭스 를 포함하고, 각 매트릭스는 64개의 행과 8개의

$$b_2^i, i \in \{0, \dots, 5\}$$

열, 크기 64인 6개의 오프셋 벡터 을 포함한다. 이러한 세트 또는 이러한 매트릭스와 오프셋 벡터들의 일부(part)의 매트릭스들과 오프셋 벡터들은 모든 다른 블록 형상(block-shapes)을 위해 사용된다.

$$4 \cdot W \cdot H$$

[0144] 매트릭스 벡터 제품의 계산에 필요한 전체 곱셈 수는 항상 보다 작거나 같다. 즉, ALWIP 모드에는 샘플당 최소 4번의 곱셈이 필요하다.

[0145] 2.8.2 경계의 평균화

$$bdry^{top} \quad bdry^{left} \quad bdry_{red}^{top}$$

[0146] 제1 단계에서, 입력 경계 및 는 더 작은 경계 및

$$bdry_{red}^{left} \quad bdry_{red}^{top} \quad bdry_{red}^{left}$$

로 감소한다. 여기서, 및 모두 4x4-블록의 경우 2개의 샘플을 포함하고, 다른 모든 케이스에서는 둘 모두 4개의 샘플을 포함한다.

$$0 \leq i < 2$$

[0147] 4x4-블록의 경우, $bdry_{red}^{top}$ 에 대해, 하나는 아래를 정의한다

$$bdry_{red}^{top}[i] = \left(\left(\sum_{j=0}^1 bdry_{red}^{top}[i \cdot 2 + j] \right) + 1 \right) \gg 1$$

[0148]

$$bdry_{red}^{left}$$

[0149] 그리고 유사하게 $bdry_{red}^{left}$ 를 정의한다.

$$W \quad W = 4 \cdot 2^k \quad 0 \leq i < 4$$

[0150] 그렇지 않으면, 블록-폭 W 이 as $W = 4 \cdot 2^k$ 로 주어지는 경우, $0 \leq i < 4$ 에 대해, 하나는 아래를 정의한다.

$$bdry_{red}^{top}[i] = \left(\left(\sum_{j=0}^{2^k-1} bdry_{red}^{top}[i \cdot 2^k + j] \right) + (1 \ll (k - 1)) \right) \gg k$$

[0151]

$$bdry_{red}^{left}$$

[0152] 그리고 유사하게 $bdry_{red}^{left}$ 를 정의한다.

$$bdry_{red}^{top} \quad bdry_{red}^{left} \quad bdry_{red}$$

[0153] 두 개의 감소된 경계 $bdry_{red}^{top}$ 및 $bdry_{red}^{left}$ 는 감소된 경계 벡터 $bdry_{red}$ 에 연결되고, 이것은 형상 4x4의 블록에 대해 4의 크기이고, 모든 다른 형상에 대해 크기 8을 갖는다. 모드가 ALWIP 모드를 참조하는 경우, 이 연결(concatenation)은 다음과 같이 정의된다:

$$bdry_{red} = \begin{cases} [bdry_{red}^{top}, bdry_{red}^{left}] & \text{for } W = H = 4 \text{ and } mode < 18 \\ [bdry_{red}^{left}, bdry_{red}^{top}] & \text{for } W = H = 4 \text{ and } mode \geq 18 \\ [bdry_{red}^{top}, bdry_{red}^{left}] & \text{for } \max(W, H) = 8 \text{ and } mode < 10 \\ [bdry_{red}^{left}, bdry_{red}^{top}] & \text{for } \max(W, H) = 8 \text{ and } mode \geq 10 \\ [bdry_{red}^{top}, bdry_{red}^{left}] & \text{for } \max(W, H) > 8 \text{ and } mode < 6 \\ [bdry_{red}^{left}, bdry_{red}^{top}] & \text{for } \max(W, H) > 8 \text{ and } mode \geq 6. \end{cases}$$

[0154]

[0155]

마지막으로, 서브 샘플링된 예측 시그널의 보간을 위해, 큰 블록 상 평균 경계의 제2 버전이 요구된다. 즉,

$$\min(W, H) > 8 \quad W \geq H \quad = 8 * 2^l$$

및 인 경우, 하나는 W 를 기록하고,

$$0 \leq i < 8$$

에 대해 아래를 정의한다.

$$bdry_{redII}^{top}[i] = \left(\left(\sum_{j=0}^{2^l-1} bdry_{red}^{top}[i \cdot 2^l + j] \right) + (1 \ll (l-1)) \right) \gg l.$$

[0156]

$$\min(W, H) > 8 \quad H > W \quad bdry_{redII}^{left}$$

[0157]

만일 및 이면, 하나는 를 유사하게 정의한다.

[0158]

2.8.3 매트릭스 벡터 곱셈에 의한 감소된 예측 시그널 생성

$$bdry_{red} \quad pred_{red}$$

[0159]

감소된 입력 벡터 중 하나는 감소된 예측 시그널 를 생성한다. 후자

$$W_{red} \quad H_{red}$$

(latter)의 시그널은 폭 과 높이 의 다운샘플링된 블록 상의 시그널이다. 여기에서,

$$W_{red} \quad H_{red}$$

및 는 다음으로 정의된다:

$$W_{red} = \begin{cases} 4 & \text{for } \max(W, H) \leq 8 \\ \min(W, 8) & \text{for } \max(W, H) > 8 \end{cases}$$

[0160]

$$H_{red} = \begin{cases} 4 & \text{for } \max(W, H) \leq 8 \\ \min(H, 8) & \text{for } \max(W, H) > 8 \end{cases}$$

[0161]

$pred_{red}$

[0162]

감소된 예측 시그널 은 매트릭스 벡터 제곱을 계산하고 오프셋을 추가하여 연산된다:

$$pred_{red} = A \cdot bdry_{red} + b$$

[0163]

$$A \quad W = H = 4 \quad W_{red} \cdot H_{red}$$

[0164]

여기에서, 는 이면, 행 및 4개의 열을 갖는 매트릭스

$$b \quad W_{red} \cdot H_{red}$$

이고, 다른 모든 케이스에서는 8개의 열을 갖는 매트릭스이다. 는 크기 를 갖는 벡터이다.

$$A \quad b \quad S_0 \quad S_1 \quad S_2$$

[0165]

매트릭스 and 벡터 는 다음의 세트 , , 중 하나로부터 취해진다. 하나는 인덱스

$$idx = idx(W, H)$$

을 다음과 같이 정의한다:

$$idx(W, H) = \begin{cases} 0 & \text{for } W = H = 4 \\ 1 & \text{for } \max(W, H) = 8 \\ 2 & \text{for } \max(W, H) > 8. \end{cases}$$

[0166]

m

[0167] 더욱이, 하나는 m 에 다음을 대입한다:

$$m = \begin{cases} mode & \text{for } W = H = 4 \text{ and } mode < 18 \\ mode - 17 & \text{for } W = H = 4 \text{ and } mode \geq 18 \\ mode & \text{for } \max(W, H) = 8 \text{ and } mode < 10 \\ mode - 9 & \text{for } \max(W, H) = 8 \text{ and } mode \geq 10 \\ mode & \text{for } \max(W, H) > 8 \text{ and } mode < 6 \\ mode - 5 & \text{for } \max(W, H) > 8 \text{ and } mode \geq 6. \end{cases}$$

[0168]

$$idx \leq 1 \quad idx = 2 \quad \min(W, H) > 4$$

[0169] 그 다음, $idx \leq 1$ 또는 $idx = 2$ 이고 $\min(W, H) > 4$ 이면, 하나는

$$A = A_{idx}^m \quad b = b_{idx}^m \quad idx = 2$$

및 $idx = 2$ 로 대입한다. 및

$$\min(W, H) = 4$$

인 경우, 하나는 A 를, $W=4$ 인 경우 다운샘플링된 블록 내 홀수(odd) x-좌표

A_{idx}^m

에 대응하고, 또는 $H=4$ 인 경우 다운샘플링된 블록 내 홀수(odd) y-좌표에 대응하는 A_{idx}^m 의 모든 행을 제거하여 생성된 매트릭스가 되도록 한다.

[0170] 마지막으로, 감소된 예측 신호는 다음과 같은 경우에 전치(transpose)로 대체된다:

$$* \quad W = H = 4 \quad \text{및} \quad mode \geq 18$$

[0171]

$$* \quad \max(W, H) = 8 \quad \text{및} \quad mode \geq 10$$

[0172]

$$* \quad \max(W, H) > 8 \quad \text{및} \quad mode \geq 6$$

[0173]

$$W = H = 4 \quad pred_{red}$$

[0174] 인 경우의 의 계산에 필요한 곱셈 수는, 케이스 A가 4개의 열과 16개의 행을 가지므로, 4이다. 다른 모든 케이스에서, A는 8개의 열과 {4} 경우, 8 개의 열과

$$W_{red} \cdot H_{red}$$

개의 행을 가지고, 하나는 이와 같이

$$8 \cdot W_{red} \cdot H_{red} \leq 4 \cdot W \cdot H$$

곱셈이 요구되는 케이스들, 예를 들면

$$pred_{red}$$

를 계산하는데 샘플 당 최대 4번의 곱셈이 요구되는 케이스들인지를 즉시 확인한다.

[0175] 2.8.4 전체 ALWIP 프로세스의 도시

[0176] 평균화, 행렬 벡터 곱셈 및 선형 보간의 전체 프로세스가 서로 다른 형상으로 도 6, 도 7, 도 8, 및 도 9에 도시된다. 나머지 형상들은 도시된 케이스들 중의 어느 하나 내에서 처리된다는 점에 유의하라.

[0177] 1. 4x4 블록이 주어지면, ALWIP는 경계의 각 축을 따라 2개의 평균을 취한다. 결과인 4개의 입력 샘플은 매트

릭스 벡터 곱셈에 들어간다. 매트릭스들은 세트 S_0 로부터 취해진다. 선형 보간은 예측 시그널을 생성하는데

필수적인 것은 아니다. 따라서, 샘플 당 총 $(4 \cdot 16)/(4 \cdot 4) = 4$ 곱셈이 수행된다.

[0178] 도 6은 4x4 블록에 대한 ALWIP를 도시한다.

[0179] 2. 8x8 블록이 주어지면, ALWIP는 경계의 각 축을 따라 4개의 평균을 취한다. 결과인 8개의 입력 샘플은 매트

릭스 벡터 곱셈에 들어간다. 매트릭스들은 세트 S_1 로부터 취해진다. 이것은 예측 블록의 홀수 위치 상 16개의 샘플들을 제공한다. 따라서, 선형 보간은 예측 시그널을 생성하는데 필수적인 것은 아니다. 따라서, 샘플

당 $(8 \cdot 16)/(8 \cdot 8) = 2$ 곱셈이 수행된다. 오프셋을 더한 후, 이러한 샘플들은 감소된 탑(top) 경계를 이용하여 수직으로 보간된다. 원본 좌측(original left) 경계를 사용하여 수평 보간이 뒤이어 수행된다.

[0180] 도 7은 8x8 블록에 대한 ALWIP를 도시한다.

[0181] 3. 8x4 블록이 주어지면, ALWIP는 경계의 수평 축을 따라 4개의 평균 및 좌측 경계 상의 4개의 원본 경계 값을

취한다. 결과인 8개의 입력 샘플은 매트릭스 벡터 곱셈에 들어간다. 매트릭스들은 세트 S_1 로부터 취해진다. 이것은 홀수 수평 및 예측 블록의 각 수직 위치 상의 16개의 샘플들을 제공한다. 따라서, 선형 보간은 예측 시

그걸을 생성하는데 필수적인 것은 아니다. 따라서, 샘플 당 $(8 \cdot 16)/(8 \cdot 4) = 4$ 곱셈이 수행된다. 오프셋을 더한 후, 이러한 샘플들은 원본 좌측(original left) 경계를 사용하여 수평 보간이 수행된다.

[0182] 도 8은 8x4 블록에 대한 ALWIP를 도시한다.

[0183] 이에 따라, 전치된(transposed) 케이스가 처리된다.

[0184] 4. 16x16 블록이 주어지면, ALWIP는 경계의 각 축을 따라 4개의 평균을 취한다. 결과인 8개의 입력 샘플은 매

트릭스 벡터 곱셈에 들어간다. 매트릭스들은 세트 S_2 로부터 취해진다. 이것은 예측 블록의 홀수 위치 상 64개의 샘플들을 제공한다. 따라서, 선행 보간은 예측 시그널을 생성하는데 필수적인 것은 아니다. 따라서, 샘플 당

$(8 \cdot 64)/(16 \cdot 16) = 2$ 곱셈이 수행된다. 오프셋을 더한 후, 이러한 샘플들은 탑(top) 경계의 8개 평균을 이용하여 수직으로 보간된다. 원본 좌측(original left) 경계를 사용하여 수평 보간이 뒤이어 수행된다.

[0185] 도 9는 16x16 블록에 대한 ALWIP를 도시한다.

[0186] 큰 형상에 대해, 프로시저는 기본적으로 동일하고, 샘플 당 곱셈 수가 4 이하라는 점을 점검하기는 쉽다. $W > 8$ 인 $W \times 8$ 에서, 샘플들이 홀수 수평 및 각 수직 위치에서 주어질 때 수평 보간만이 필요하다. 마지막으로, $W > 8$ 인

$W \times 4$ 에서, A_k 를 다운샘플링된 블록의 수평 축을 따라 홀수 엔트리에 대응하는 모든 행을 제거하여 생성된 매트릭스라고 가정한다. 따라서, 출력 크기는 32이고, 다시 수평 보간만이 수행된다. 감소된 예측의 계산을 위

해, 샘플 당 $(8 \cdot 32)/(W \cdot 4) = 64/W$ 곱셈이 수행된다. $W=16$ 의 경우, 추가 곱셈이 요구되지 않는 반면, $W > 16$ 의 경우 샘플 당 2 보다 작은 곱셈이 선행 보간에서 요구된다. 따라서, 전체 곱셈 수는 4 보다 작거나, 4와 같다.

[0187] 이에 따라, 전치된(transposed) 케이스가 처리된다.

[0188] 2.8.5 단일 스텝 선형 보간

$$\max(W, H) \geq 8 \quad W \times H$$

[0189] 를 갖는 블록에 대해, 선형 보간에 의해

$$W_{red} \times H_{red} \quad pred_{red}$$

상에 감소된 예측 시그널로부터 예측 시그널이 생성된다. 블록 형상에 의존하여, 수직, 수평 또는 양 방향으로 선형 보간이 행해진다. 선형 보간이 양 방향에 대해 적용되는 경우, $W < H$ 이면 수평 방향에서 먼저 적용되고, 아니면 수직 방향에서 먼저 적용된다.

[0199] $predmode_{ALWIP} = map_angular_to_alwip_{idx}[predmode_{Angular}]$.

[0200] 폭 W 및 높이 H의 각 PU에 대해, 하나는 인덱스를 아래로 정의한다.

[0201] $idx(PU) = idx(W, H) \in \{0,1,2\}$

[0202] 이것은 3개의 세트로부터 ALWIP 파라미터가 위에서 설명한 섹션 1.3과 같이 취해지는 것을 가리킨다.

$$PU_{above}$$

[0203] 만일 위에서 언급한 예측 유닛(PU) PU_{above} 가 이용가능하고, 현재 PU와 동일한 CTU에 속해 있고,

$$idx(PU) = idx(PU_{above})$$

이고, $ALWIP_{mode}$ 가 $ALWIP-mode$

$$predmode_{ALWIP}^{above} = predmode_{ALWIP}^{above}(PU_{above})$$

를 갖는 PU_{above} 상에 적용되면, 하나는 아래와 같이 대입한다.

$$mode_{ALWIP}^{above} = predmode_{ALWIP}^{above}$$

[0204]

[0205] 만일 위에서 언급한 PU가 이용가능하고, 현재 PU와 동일한 CTU에 속해 있고, 기존 인트라 예측 모드

$$predmode_{Angular}^{above}$$

가 PU 상에 적용되면, 하나는 아래와 같이 대입한다.

$$mode_{ALWIP}^{above} = map_angular_to_alwip_{idx}(PU_{above})[predmode_{Angular}^{above}]$$

[0206]

[0207] 다른 모든 케이스에서, 하나는 아래와 같이 대입한다.

$$mode_{ALWIP}^{above} = -1$$

[0208]

[0209] 이것은 이 모드가 이용가능하지 않음을 의미한다. 이러한 방법으로, 그러나 레프트 PU가 현재 PU와 동일한 CTU

$$mode_{ALWIP}^{left}$$

에 속할 필요가 있다는 제한 없이, 하나는 모드 $mode_{ALWIP}^{left}$ 를 유도한다.

$$list_{idx} \quad idx \in \{0,1,2\}$$

[0210] 마지막으로, 3개의 고정된 디폴트 리스트 , 가 제공되고, 이들 각

$$list_{idx}(PU)$$

각은 3개의 구별되는 ALWIP 모드를 포함한다. 디폴트 리스트 , 모드

$$mode_{ALWIP}^{above} \quad mode_{ALWIP}^{left}$$

및 중에서, 하나는 -1을 디폴트 값으로 대체함은 물론 반복(repetition)을 제거함으로써 3개의 구별되는 MPM을 구성한다.

[0211] 2.8.7 기존의 루마 및 크로마 인트라 예측 모드를 위한 적용된 MPM 리스트 유도

[0212] 제안된 ALWIP-모드는 다음과 같이 기존 인트라 예측 모드들의 MPM-기반 코딩과 조화된다. 기존의 인트라 예측 모드를 위한 루마 및 크로마 MPM-리스트 유도 프로세스는, 주어진 PU 상의 ALWIP-모드

$$predmode_{ALWIP}$$

를 기존의 인트라 예측 모드들 중 어느 하나로 매핑하는, 고정된 테이블

$$map_alwip_to_angular_{idx} \quad idx \in \{0,1,2\}$$

를 사용한다.

[0213]
$$predmode_{Angular} = map_alwip_to_angular_{idx}(PU)[predmode_{ALWIP}]$$

$$predmode_{ALWIP}$$

[0214] 루마 MPM-리스트 유도를 위해, 인접 루마 블록이 ALWIP 모드 를 사용하게

$$predmode_{Angular}$$

될 때마다, 이 블록은 기존 인트라 예측 모드 를 사용하고 있는 것처럼 처리된다. 크로마 MPM-리스트 유도를 위해, 현재 루마 블록이 LWIP 모드

$$predmode_{ALWIP}$$

를 사용할 때마다, ALWIP-모드에서 기존 인트라 예측 모드로의 변환(translate)을 위해 동일한 매핑이 사용된다.

[0215] 2.9 67개의 인트라 예측 모드로 코딩하는 인트라 모드

[0216] 보통의(natural) 비디오에 제공되는 임의의 에지 방향을 캡처하기 위해, VTM4의 디렉셔널(directional) 인트라 모드의 수는 33개에서, HEVC에서 사용되는 것과 같이, 65개로 확장된다. HEVC에 없는 새로운 디렉셔널

(directional) 모드는 도 14의 빨간색 점선 화살표로 도시되고, 플래나(planar) 및 DC 모드는 동일하게 유지된다. 이러한 밀도가 높은 디렉셔널 인트라 예측 모드는 모든 블록 크기 및 루마와 크로나 인트라 예측 모드 둘다에 적용된다.

- [0217] MRL 및 ISP 코딩 틀이 적용되었는지 여부에 관계없이, 인트라 블록에 대해 통합된 6-MPM 리스트가 제안된다. MPM 리스트는 VTM4.0에서와 같이 왼쪽 및 인접 블록 위의 인트라 모드에 기초하여 구성된다. 왼쪽 모드가 left로 표시되고, 위의 블록의 모드가 Above와 같이 표시되며, 통합된 MPM 리스트는 다음과 같이 구성된다.
- [0218] * 인접 블록을 사용할 수 없는 경우, 해당 인트라 모드는 디폴트로 플래나(Planar)로 설정된다.
- [0219] * Left 및 Above 두 개의 모드 모두가 비-각도(non-angular) 모드인 경우:
- [0220] MPM 리스트 -> {Planar, DC, V, H, V-4, V+4}
- [0221] * Left 및 Above 중 어느 하나가 각도(angular) 모드이고, 나머지는 비-각도 모드인 경우:
- [0222] 모드 Max를 Left 및 Above 내의 더 큰 모드로 세팅함
- [0223] MPM 리스트 -> {Planar, Max, DC, Max -1, Max +1, Max -2}
- [0224] * Left 및 Above 두 개의 모드 모두가 각도(angular) 모드인 경우:
- [0225] 모드 Left 및 Above의 차이가 2에서 62 사이의 범위에 있을 때, 아래
- [0226] MPM 리스트 -> {Planar, Left, Above, DC, Max -1, Max +1}를 포함하고,
- [0227] 그렇지 않으면, MPM 리스트 -> {Planar, Left, Above, DC, Max -2, Max +2}
- [0228] * Left 및 Above 두 개의 모드 모두가 각도(angular) 모드이고, 그들이 동일한 경우:
- [0229] MPM 리스트 -> {Planar, Left, Left -1, Left +1, DC, Left -2}

[0230] 7.3.6.5 코딩 유닛 선택스

	Descriptor
coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	
if(tile_group_type != 1 sps_ibc_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA)	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] == 0 && tile_group_type != 1)	
pred_mode_flag	ae(v)
if(((tile_group_type == 1 && cu_skip_flag[x0][y0] == 0) (tile_group_type != 1 && CuPredMode[x0][y0] != MODE_INTRA)) && sps_ibc_enabled_flag)	
pred_mode_ibc_flag	ae(v)
}	
if(CuPredMode[x0][y0] == MODE_INTRA) {	
if(sps_pcm_enabled_flag && cbWidth >= MinIpcmCbSizeY && cbWidth <= MaxIpcmCbSizeY && cbHeight >= MinIpcmCbSizeY && cbHeight <= MaxIpcmCbSizeY)	
pcm_flag[x0][y0]	ae(v)
if(pcm_flag[x0][y0]) {	
while(!byte_aligned())	
pcm_alignment_zero_bit	f(1)
pcm_sample(cbWidth, cbHeight, treeType)	
} else {	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
if((y0 % CtbSizeY) > 0)	
intra_luma_ref_idx[x0][y0]	ae(v)
if((intra_luma_ref_idx[x0][y0] == 0 && (cbWidth <= MaxTbSizeY cbHeight <= MaxTbSizeY) && (cbWidth * cbHeight > MinTbSizeY * MinTbSizeY))	
intra_subpartitions_mode_flag[x0][y0]	ae(v)
if((intra_subpartitions_mode_flag[x0][y0] == 1 && cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY)	
intra_subpartitions_split_flag[x0][y0]	ae(v)
if((intra_luma_ref_idx[x0][y0] == 0 && intra_subpartitions_mode_flag[x0][y0] == 0)	
intra_luma_mpm_flag[x0][y0]	ae(v)
if(intra_luma_mpm_flag[x0][y0]) {	
if(intra_luma_ref_idx[x0][y0] == 0)	
intra_luma_not_planar_flag[x0][y0]	ae(v)
if(intra_luma_not_planar_flag[x0][y0])	
intra_luma_mpm_idx[x0][y0]	ae(v)
} else	
intra_luma_mpm_remainder[x0][y0]	ae(v)
}	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA)	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
...	
}	
if(!pcm_flag[x0][y0]) {	
if(CuPredMode[x0][y0] != MODE_INTRA && merge_flag[x0][y0] == 0)	
cu_cbf	ae(v)
if(cu_cbf) {	
if(CuPredMode[x0][y0] == MODE_INTER && sps_sbt_enabled_flag && !ciip_flag[x0][y0]) {	
if(cbWidth <= MaxSbtSize && cbHeight <= MaxSbtSize) {	
allowSbtVerH = cbWidth >= 8	
allowSbtVerQ = cbWidth >= 16	
allowSbtHorH = cbHeight >= 8	
allowSbtHorQ = cbHeight >= 16	
if(allowSbtVerH allowSbtHorH allowSbtVerQ allowSbtHorQ)	

[0231]

cu_sbt_flag	ae(v)
}	
if(cu_sbt_flag) {	
if((allowSbtVerH allowSbtHorH) && (allowSbtVerQ allowSbtHorQ))	
cu_sbt_quad_flag	ae(v)
if((cu_sbt_quad_flag && allowSbtVerQ && allowSbtHorQ) (!cu_sbt_quad_flag && allowSbtVerH && allowSbtHorH))	
cu_sbt_horizontal_flag	ae(v)
cu_sbt_pos_flag	ae(v)
}	
}	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
}	
}	
}	

[0232]

[0233]

신택스 요소 `intra_luma_mpm_flag[x0][y0]`, `intra_luma_not_planar_flag[x0][y0]`, `intra_luma_mpm_idx[x0][y0]` 및 `intra_luma_mpm_remainder[x0][y0]` 는 루마 샘플들을 위한 인트라 예측 모드를 지정한다. 어레이 인덱스 `x0`, `y0`는 탑-레프트 루마 샘플과 관련하여 고려된 코딩 블록의 탑-레프트 루마 샘플의 위치 (`x0` , `y0`)를 지정한다. `intra_luma_mpm_flag[x0][y0]`가 1과 같은 경우, 8.4.2 절에 따라 인트라 예측 모드는 인접 인트라 예측된 코딩 유닛으로부터 추론된다.

[0234]

`intra_luma_mpm_flag[x0][y0]`가 존재하지 않는 경우, 1과 같은 것으로 추론된다.

[0235]

`intra_luma_not_planar_flag[x0][y0]`가 존재하지 않는 경우, 1과 같은 것으로 추론된다.

[0236]

2.10 크로마 인트라 모드 코딩

[0237]

크로마 인트라 모드 코딩의 경우, 크로마 인트라 모드 코딩에 대해 총 8개의 인트라 모드가 허용된다. 이러한 모드에는 5개의 기존 인트라 모드와 3개의 크로스-컴포넌트(cross-component) 선형 모델 모드가 포함된다. 크로마 DM 모드는 대응하는 루마 인트라 예측 모드를 사용한다. 루마 및 크로마 성분에 대한 분리된 블록 파티셔닝 구조가 I 슬라이스에서 인에이블되어 있기 때문에, 하나의 크로마 블록은 복수의 루마 블록에 대응할 수 있다. 따라서, 크로마 DM 모드의 경우, 현재 크로마 블록의 중심 위치를 커버하는 대응 루마 블록의 인트라 예측 모드가 직접 상속(inherit)된다.

[0238]

3. 개시된 실시예에 의해 해결된 기술적 문제의 예

[0239]

QR-BDPCM은 화면 콘텐츠 코딩에서 코딩 이점을 얻을 수 있지만, 여전히 몇 가지 단점이 있을 수 있다.

[0240]

1. QR-BDPCM 모드의 예측은 수평 및 수직 인트라 예측으로만 제한되고, 이는 QR-BDPCM 모드에서 예측 효율성을 제한할 수 있다.

[0241]

2. 인트라 예측 모드는 QR-BDPCM 코딩 블록에 대해 시그널링 되고, 이는 QR-BDPCM 모드의 속도 비용을 증가시킬 수 있다.

[0242]

3. 시그널링된 메시지를 QR-BDPCM 모드의 예측 모드로 매핑할 때, 인접 정보는 고려되지 않는다.

[0243]

4. QR-BDPCM은 복잡한 잔차 블록에서 코딩 성능을 구성할 수 있는 수평 DPCM 및 수직 DPCM만 지원하여 잔차(residue)를 나타낸다.

[0244]

5. QR-BDPCM의 잔차 범위는 다른 비 QR-BDPCM 모드의 최대 범위를 초과할 수 있다.

[0245]

6. QR-BDPCM은 블록 형상을 고려하지 않는다.

[0246]

7. 루마 블록이 QR-BDPCM으로 코딩될 때 크로마를 처리하는 방법은 알 수 없다.

[0247]

8. QR-BDPCM은 제1 MPM 모드만 저장된 인트라 모드로만 사용하고, 이는 인트라 모드의 코딩 효율을 제한할 수 있다.

[0248]

4. 실시예 및 기술들

[0249]

아래 항목 리스트는 일반적인 개념을 설명하는 예로 간주되어야 한다. 이러한 발명은 좁은 방법으로 해석되어

서는 안된다. 더욱이, 이들 발명은 어떤 방식으로든 결합될 수 있다.

- [0250] 1. QR-BDPCM 코딩 블록내의 샘플 예측은 매트릭스 기반 인트라 예측(MIP) 방법에 의해 생성될 수 있다.
- [0251] a. 일 예에서, QR-BDPCM과 MIP가 하나의 블록에 대해 모두 인에이블되면, MIP에서 허용된 모드의 일부만 지원되는 것으로 제한된다.
- [0252] i. 일 예에서, 허용된 모드의 일부에는 수평 및/또는 수직 노멀(normal) 인트라 모드로 매핑될 수 있는 매트릭스 기반 인트라 예측 방법과 연관된 모드가 포함될 수 있다.
- [0253] ii. 일 예에서, 허용된 모드의 일부에는 수평 및/또는 수직 노멀 인트라 모드로 매핑될 수 있는 매트릭스 기반 인트라 예측 방법과 연관된 모드만 포함될 수 있다.
- [0254] b. 일 예에서, QR-BDPCM과 MIP가 모두 한 블록에 대해 인에이블되면 MIP의 허용된 모든 모드가 지원된다.
- [0255] 2. QR-BDPCM 코딩 블록의 샘플 예측은 수직/수평 인트라 예측 이외의 인트라 예측 모드에 의해 생성될 수 있다.
- [0256] a. 일 예에서, QR-BDPCM 코딩된 블록의 샘플은 인트라 예측 모드 K 에 의해 예측될 수 있다.
- [0257] i. 일 예에서, K 는 플래나 모드일 수 있다.
- [0258] ii. 일 예에서, K 는 DC 모드일 수 있다.
- [0259] iii. 일 예에서, K 는 수평 모드일 수 있다.
- [0260] iv. 일 예에서, K 는 수직 모드일 수 있다.
- [0261] v. 일 예에서, K 는 최대 가능(most probable) 모드 목록에서 하나의 후보일 수 있다.
- [0262] vi. 일 예에서, K 는 비트스트림에서 시그널링될 수 있다.
- [0263] b. QR-BDPCM에 대한 허용된 인트라 예측 모드는 아래에 기초할 수 있다.
 - [0264] i. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
 - [0265] ii. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
 - [0266] iii. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
 - [0267] iv. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
 - [0268] v. 현재 블록의 인접 블록의 인트라 예측 모드
 - [0269] vi. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
 - [0270] vii. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
 - [0271] viii. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
 - [0272] ix. 독립(separate)/이중 코딩 트리 구조
 - [0273] x. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0274] 3. QR-BDPCM 코딩 블록 내의 샘플 예측은 인접한 비인접 샘플에 의해 생성될 수 있다.
 - [0275] a. 일 예에서, IBC 병합 모드의 경우 QR-BDPCM도 인에이블될 수 있다.
 - [0276] b. 일 예에서, IBC AMVP 모드의 경우 QR-BDPCM도 인에이블될 수 있다.
 - [0277] c. IBC 및 QR-BDPCM에 사용되는 블록 벡터는 시그널링 또는 유도되거나 미리 정의될 수 있다.
 - [0278] i. 일 예에서, IBC 모드는 모션 벡터(블록 벡터) 및/또는 병합 인덱스로 표시될 수 있다.
 - [0279] ii. 일 예에서, IBC 모드는 디폴트 모션 벡터로 표시될 수 있다.
 - [0280] 1. 일 예에서, 디폴트 모션 벡터는 $(-w, 0)$ 일 수 있고, 여기서 w 는 양의 정수이다.
 - [0281] 2. 일 예에서, 디폴트 모션 벡터는 $(0, -h)$ 일 수 있고, 여기서 h 는 양의 정수이다.
 - [0282] 3. 일 예에서, 디폴트 모션 벡터는 $(-w, -h)$ 일 수 있고, 여기서 w 와 h 는 양의 정수이다.

- [0283] iii. 일 예에서, IBC 및 QP-BDPCM 코딩된 블록에 사용되는 모션 벡터의 표시는 다음을 기반으로 할 수 있다:
- [0284] 1. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
- [0285] 2. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
- [0286] 3. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
- [0287] 4. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
- [0288] 5. 현재 블록의 인접 블록의 모션 벡터
- [0289] 6. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
- [0290] 7. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
- [0291] 8. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
- [0292] 9. 독립(separate)/이중 코딩 트리 구조
- [0293] 10. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0294] d. 일 예에서, QR-BDPCM 모드의 샘플 예측은 인터(inter) 예측 틀(예: 아핀(affine) 모드, 병합 모드 및 인터 모드)에 의해 생성될 수 있다.
- [0295] 4. QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 즉석에서(on the fly) 유도될 수 있다.
- [0296] a. 일 예에서, QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 현재 인트라 예측 모드의 표시에 기초하여 추론(infer)될 수 있다
- [0297] i. 일 예에서, QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 인트라 예측 모드가 수직인 경우 수직으로 추론될 수 있다.
- [0298] ii. 일 예에서, QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 인트라 예측 모드가 수평인 경우 수평으로 추론될 수 있다.
- [0299] iii. 일 예에서, QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 인트라 예측 모드가 수평인 경우 수직으로 추론될 수 있다.
- [0300] iv. 일 예에서, QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 인트라 예측 모드가 수직인 경우 수평으로 추론될 수 있다.
- [0301] b. QR-BDPCM 내 양자화된 잔차 예측 방향의 표시는 아래에 기반할 수 있다:
- [0302] i. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
- [0303] ii. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
- [0304] iii. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
- [0305] iv. 현재 블록 및/또는 인접 블록의 최대 가능 모드
- [0306] v. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
- [0307] vi. 현재 블록의 인접 블록의 인트라 예측 모드
- [0308] vii. 현재 블록의 인접 블록의 모션 벡터
- [0309] viii. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
- [0310] ix. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
- [0311] x. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
- [0312] xi 독립(separate)/이중 코딩 트리 구조
- [0313] xii. 현재 블록에 적용된 변환 타입

- [0314] xiii. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0315] 5. 저장되는 QR-BDPCM 코딩 블록의 인트라 모드는 인트라 예측 프로세스에 사용되는 인트라 예측 모드와 정렬될 수 있다.
- [0316] a. 일 예에서, 저장되는 QR-BDPCM 코딩 블록의 인트라 모드는 QR-BDPCM이 수직 인트라 예측(예를 들어, 현재 블록의 `bdpcm_dir_flag`가 1)을 채택할 때 수직 모드로 추론될 수 있다.
- [0317] b. 일 예에서, 저장되는 QR-BDPCM 코딩 블록의 인트라 모드는 QR-BDPCM이 수평 인트라 예측(예를 들어, 현재 블록의 `bdpcm_dir_flag`가 0)을 채택할 때 수평 모드로 추론될 수 있다.
- [0318] c. 일 예에서, 저장되는 QR-BDPCM 코딩 블록의 인트라 모드는 QR-BDPCM이 탑-레프트(top-left) 인트라 예측 방향을 채택하는 경우 탑-레프트 모드(예: VVC의 모드 34)로 추론될 수 있다.
- [0319] d. 일 예에서, 저장되는 QR-BDPCM 코딩 블록의 인트라 모드는 QR-BDPCM 모드에서 인트라 예측 프로세스에 사용될 때 모드로 추론될 수 있다.
- [0320] e. 일 예에서, 저장되는 QR-BDPCM 코딩 블록의 인트라 모드는 QR-BDPCM 모드에서 잔차 예측 프로세스에 사용될 때 모드로 추론될 수 있다.
- [0321] f. 일 예에서, QR-BDPCM에 코딩된 블록의 인트라 모드는 MPM(최대 가능 모드) 리스트에서 하나의 모드로 추론될 수 있다.
- [0322] g. 일 예에서, QR-BDPCM에 코딩된 블록의 인트라 모드는 미리 정의된 모드로 추론될 수 있다.
- [0323] i. 일 예에서, 미리 정의된 모드는
 - [0324] 1. 플래나 모드
 - [0325] 2. DC 모드
 - [0326] 3. 수직 모드
 - [0327] 4. 수평 모드
- [0328] h. 일 예에서, QR-BDPCM 모드에서 코딩된 블록의 인트라 모드는 다음에 기초하여 결정될 수 있다.
 - [0329] i. 컬러 컴포넌트
 - [0330] ii. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
 - [0331] iii. `bdpcm_dir_flag`
 - [0332] iv. `bdpcm_flag`
 - [0333] ii. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
 - [0334] iii. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
 - [0335] iv. 현재 블록 및/또는 인접 블록의 최대 가능 모드
 - [0336] v. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
 - [0337] vi. 현재 블록의 인접 블록의 인트라 예측 모드
 - [0338] vii. 현재 블록의 인접 블록의 모션 벡터
 - [0339] viii. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
 - [0340] ix. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
 - [0341] x. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
 - [0342] xi 코딩 트리 구조
 - [0343] xii. 현재 블록에 적용된 변환 타입
 - [0344] xiii. 슬라이스/타일 그룹 타입 및/또는 픽처 타입

- [0345] i. 일 예에서, 저장된 인트라 예측 모드는 코딩될 이후 블록의 MPM 리스트 구성 등의 이후 블록을 코딩하는 데 사용될 수 있다.
- [0346] 6. QR-BDPCM의 시그널링된 인덱스에서 QR-BDPCM 모드의 인트라 예측 모드로 매핑하는 것은 아래에 기초할 수 있다
- [0347] a. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
- [0348] b. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
- [0349] c. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
- [0350] d. 현재 블록 및/또는 인접 블록의 최대 가능 모드
- [0351] e. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
- [0352] f. 현재 블록의 인접 블록의 인트라 예측 모드
- [0353] g. 현재 블록의 인접 블록의 모션 벡터
- [0354] h. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
- [0355] i. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
- [0356] j. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
- [0357] k. 독립(separate)/이중 코딩 트리 구조
- [0358] l. 현재 블록에 적용된 변환 타입
- [0359] m. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0360] 7. QR-BDPCM에서, 수평 및 수직 방향을 따라 양자화된 잔차가 예측된다. 수직 및 수평 방향 이외의 방향을 따

라 양자화된 잔차를 예측하는 것이 제안된다. $Q(r_{i,j})$ 는 양자화된 잔차를 나타내고, 잔차 $\tilde{r}_{i,j}$ 는 잔차 예측 프로세스 이후의 양자화된 잔차를 나타내는 것으로 가정하자.

- [0361] a. 일 예에서, 45도 QR-BDPCM이 지원될 수 있다.

- [0362] i. 일 예에서, DPCM은 45도 방향을 따라 수행될 수 있으며, 여기서 $\tilde{r}_{i,j}$ 는 $Q(r_{(i-1),(j-1)})$

가 사용가능한 경우 $Q(r_{i,j}) - Q(r_{(i-1),(j-1)})$ 에 의해 유도될 수 있다.

- [0363] b. 일 예에서, 135도 QR-BDPCM이 지원될 수 있다.

- [0364] i. 일 예에서, DPCM은 45도 방향으로 수행될 수 있고, 여기서 $\tilde{r}_{i,j}$ 는 $Q(r_{(i-1),(j+1)})$ 가 사

용 가능한 경우 $Q(r_{i,j}) - Q(r_{(i-1),(j+1)})$ 에 의해 유도될 수 있다.

- [0365] c. 일 예에서, QR-BDPCM에서 모든 방향이 지원될 수 있다.

[0366] i. 일 예에서, $\tilde{r}_{i,j}$ 는 $Q(r_{(i-m),(j-n)})$ 가 사용 가능한 경우,

$Q(r_{i,j}) - Q(r_{(i-m),(j-n)})$ 에 의해 유도될 수 있다.

- [0367] 1. 일 예에서, m 및/또는 n은 비트스트림 내에서 시그널링될 수 있다.
- [0368] 2. 일 예에서, m 및/또는 n은 정수이고, 아래에 기초할 수 있다.
- [0369] 3. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
- [0370] 4. i 및/또는 j
- [0371] 5. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
- [0372] 6. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
- [0373] 7. 현재 블록 및/또는 인접 블록의 최대 가능 모드
- [0374] 8. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
- [0375] 9. 현재 블록의 인접 블록의 인트라 예측 모드
- [0376] 10. 현재 블록의 인접 블록의 모션 벡터
- [0377] 11. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
- [0378] 12. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
- [0379] 13. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
- [0380] 14. 독립(separate)/이중 코딩 트리 구조

- [0381] 15. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0382] 8. QR-BDPCM은 크로마 블록(예: Cb/Cr 또는 B/R 컬러 컴포넌트)에 적용될 수 있다.
- [0383] a. 일 예에서, 루마 및 크로마 QR-BDPCM 코딩된 블록에 대해 허용된 인트라 예측 방향은, 예를 들어 수평 및 수직 뿐일 수 있다.
- [0384] b. 일 예에서, 루마 및 크로마 QR-BDPCM 코딩된 블록에 대해 허용된 예측 방법은, 예를 들어, IBC/인터/수평 및 수직 인트라 예측 모드와 동일할 수 있다.
- [0385] c. 일 예에서, 루마 및 크로마 QR-BDPCM 코딩된 블록에 대해 허용된 잔차 예측 방향은 동일할 수 있다.
- [0386] d. 일 예에서, 크로마 QR-BDPCM에 대한 잔차 예측 방향은 해당 루마 블록에 대한 잔차 예측 방향으로부터 유도될 수 있다.
- [0387] i. 일 예에서, 대응하는 루마 블록은 함께 배치된(collocated) 루마 블록일 수 있다.
- [0388] ii. 일 예에서, 대응하는 루마 블록은 크로마 블록의 어퍼-레스트(upper-left) 코너의 함께 배치된 샘플을 포함하는 루마 블록일 수 있다.
- [0389] iii. 일 예에서, 대응하는 루마 블록은 크로마 블록의 중심 샘플의 함께 배치된 샘플을 포함하는 루마 블록일 수 있다.
- [0390] e. 일 예에서, CCLM 및 QR-BDPCM은 동일한 크로마 블록에 적용할 수 없었다.
- [0391] i. 대안적으로, CCLM은 QR-BDPCM 코딩된 블록에도 적용될 수 있다.

[0392] f. 일 예에서, 조인트 크로마 잔차 코딩(예를 들어, 조인트 CB 및 cr 코딩) 방법 및 QR-BDPCM은 동일한 크로마 블록에 적용될 수 없었다.

[0393] 9. QR-BDPCM 내의 재구성되어 양자화된 잔차는 특정 범위 내에 있는 것으로 제한될 수 있다.

[0394] a. 일 예에서, 모든 양자화된 잔차 차이(예를 들어, 식 2-7-1 및 2-7-2의 $\tilde{r}_{i,j}$)가 특정 범위 내에 있을 수 있다는 제한이 추가될 수 있다.

[0395] b. 일 예에서, 모든 양자화된 잔차 차이(예를 들어, 식 2-7-3 및 2-7-4의 $Q(r_{i,j})$)가 특정 범위 내에 있을 수 있다는 제한이 추가될 수 있다.

[0396] c. 일 예에서, 클리핑 동작은 양자화된 잔차 차이(예를 들어, 식 2-7-1 및 2-7-2의 $\tilde{r}_{i,j}$)에 적용될 수 있으므로 재구성되어 양자화된 잔차는 특정 범위 내에 있을 수 있다.

[0397] d. 일 예에서, 클리핑 동작은 재구성되어 양자화된 잔차 차이(예를 들어, 식 2-7-3 및 2-7-4의

$Q(r_{i,j})$)에 적용될 수 있으므로 재구성되어 양자화된 잔차는 특정 범위 내에 있을 수 있다.

[0398] e. 일 예에서, 클리핑 동작은 $(x < \min? \min: (x > \max? \max: x))$ 로 정의될 수 있다.

[0399] f. 일 예에서, 클리핑 동작은 $(x \leq \min? \min: (x \geq \max? \max: x))$ 로 정의될 수 있다.

[0400] g. 일 예에서, 클리핑 동작은 $(x < \min? \min: (x \geq \max? \max: x))$ 로 정의될 수 있다.

[0401] h. 일 예에서, 클리핑 동작은 $(x \leq \min? \min: (x > \max? \max: x))$ 로 정의될 수 있다.

[0402] i. 일 예에서, min 및/또는 max는 음수 또는 양수일 수 있다.

[0403] j. 일 예에서, min은 -32768로 설정되고, max는 32767로 설정된다.

[0404] i. 대안적으로, min 및/또는 max는 QR-BDPCM으로 코딩되지 않은 블록에 대한 역 양자화 범위에 의존할 수 있다.

[0405] ii. 대안적으로, min 및/또는 max는 입력 샘플/재구성된 샘플의 비트 뎀스(depth)에 의존할 수 있다.

[0406] iii. 대안적으로, min 및/또는 max는 무손실 코딩이 사용되는지 여부에 의존할 수 있다.

[0407] 1. 일 예에서, min 및/또는 max는 transquant_bypass_enabled_flag에 의존할 수 있다.

[0408] 2. 일 예에서, min 및/또는 max는 cu_transquant_bypass_flag에 의존할 수 있다.

[0409] k. 일 예에서, min 및/또는 max는 다음에 기초한다

[0410] i. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지

[0411] ii. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)

[0412] iii. 현재 블록 및/또는 인접 블록의 블록 형상(shape)

[0413] iv. 현재 블록 및/또는 인접 블록의 최대 가능 모드

[0414] v. 현재 블록의 인접 블록의 예측 모드(인트라/인터)

[0415] vi. 현재 블록의 인접 블록의 인트라 예측 모드

[0416] vii. 현재 블록의 인접 블록의 모션 벡터

- [0417] viii. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
- [0418] ix. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
- [0419] x. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
- [0420] xi. 독립(separate)/이중 코딩 트리 구조
- [0421] xii. 현재 블록에 적용된 변환 타입
- [0422] xiii. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0423] 10. QR-DPCM은 마지막 행/열로부터 블록의 첫 행/열까지 적용될 수 있다.
- [0424] a. 일 예에서, 잔차 예측 방향이 수평인 경우(i+1)-th 열의 잔차를 사용하여 i-th 열의 잔차를 예측할 수 있다.
- [0425] b. 일 예에서, 잔차 예측 방향이 수직인 경우(i+1)-th 행의 잔차를 사용하여 i-th 행의 잔차를 예측할 수 있다.
- [0426] 11. QR-DPCM은 블록의 서브 세트에 적용될 수 있다.
- [0427] a. 일 예에서, 잔차 예측 방향이 수평인 경우 QR-DPCM은 잔차의 레프트 모스트(left most) k 열에는 적용되지 않는다.
- [0428] b. 일 예에서, 잔차 예측 방향이 수직인 경우 QR-DPCM은 잔차의 어퍼 모스트(upper most) k 열에는 적용되지 않는다.
- [0429] c. 일 예에서, 잔차 예측 방향이 수평인 경우 QR-DPCM은 잔차의 라이트 모스트(right most) k 열에는 적용되지 않는다.
- [0430] d. 일 예에서, 잔차 예측 방향이 수직인 경우 QR-DPCM은 잔차의 바텀 모스트(bottom most) k 열에는 적용되지 않는다.
- [0431] e. 상술한 k의 값은 다음에 기반하여 미리 정의된 값일 수 있다.
- [0432] i. SPS/VPS/PPS/픽처 헤더/슬라이스 헤더/타일 그룹 헤더/LCU 행/LCU의 그룹 내 시그널링된 메시지
- [0433] ii. 현재 블록 및/또는 인접 블록의 블록 디멘전(dimension)
- [0434] iii. 현재 블록 및/또는 인접 블록의 블록 형상(shape)
- [0435] iv. 현재 블록 및/또는 인접 블록의 최대 가능 모드
- [0436] v. 현재 블록의 인접 블록의 예측 모드(인트라/인터)
- [0437] vi. 현재 블록의 인트라 예측 모드
- [0438] vii. 현재 블록의 인접 블록의 인트라 예측 모드
- [0439] viii. 현재 블록의 인접 블록의 모션 벡터
- [0440] ix. 현재 블록의 인접 블록의 QR-BDPCM 모드 표시
- [0441] x. 현재 블록 및/또는 해당 블록의 현재 양자화 파라미터
- [0442] xi. 컬러 포맷의 표시(예: 4:2:0, 4:4:4)
- [0443] xii. 독립(separate)/이중 코딩 트리 구조
- [0444] xiii. 현재 블록에 적용된 변환 타입
- [0445] xiv. 슬라이스/타일 그룹 타입 및/또는 픽처 타입
- [0446] 12. QR-DPCM은 블록에 대해 세그먼트 별로 적용될 수 있다.

[0447] a. 일 예에서, 잔차 예측 방향이 수직이고 $N=nK$ 인 경우, 잔차 예측은 다음과 같이 수행될 수 있다.

[0448]
$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & i \% K = 0, 0 \leq j \leq (N - 1) \\ Q(r_{i,j}) - Q(r_{(i-1),j}), & i \% K > 0, i \leq (M - 1), 0 \leq j \leq (N - 1) \end{cases}$$

[0449] b. 일 예에서, 잔차 예측 방향이 수평이고 $N=mK$ 인 경우, 잔차 예측은 다음과 같이 수행될 수 있다.

[0450]
$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & 0 \leq i \leq (M - 1), j \% K = 0 \\ Q(r_{i,j}) - Q(r_{i,(j-1)}), & 0 \leq i \leq (M - 1), j \% K > 0, j \leq (N - 1) \end{cases}$$

[0451] 13. 하나의 컬러 컴포넌트에 대해 QR-DPCM을 인에이블/디스에이블하는 것은 다른 컬러 컴포넌트와 연관된 것으로부터 유도될 수 있다.

[0452] a. 일 예에서, 크로마 블록의 경우, QR-DPCM을 인에이블할지 여부는 함께 배치된(collocated) 루마 블록 내에서 하나 또는 여러 대표 블록과 연관된 QR-DPCM의 사용에 의존할 수 있다.

[0453] i. 일 예에서, 대표 블록은 DM 유도에서 사용된 것과 동일한 방식으로 정의될 수 있다.

[0454] ii. 일 예에서, 함께 배치된(collocated) 루마 블록 내의 대표 블록이 QR-DPCM 코딩되고, 현재 크로마 블록이 DM 모드로 코딩되는 경우, QR-DPCM도 현재 크로마 블록에 대해 인에이블될 수 있다.

[0455] b. 대안적으로, QR-DPCM의 사용의 표시는 크로마 성분에 대해 시그널링될 수 있다.

[0456] i. 일 예에서, 하나의 플래그는 두 크로마 구성 요소에 대한 사용을 나타내기 위해 시그널링될 수 있다.

[0457] ii. 대안적으로, 두 개의 플래그가 각각 두 개의 크로마 성분에 대한 사용을 나타내도록 시그널링될 수 있다.

[0458] iii. 일 예에서, 크로마 블록이 특정 모드(예: CCLM)로 코딩되면 QR-DPCM의 사용 표시의 시그널링은 스킵된다.

[0459] 14. 위의 방법은 DPCM/QR-DPCM의 다른 변형들에도 적용될 수 있다.

[0460] **5. 실시예**

[0461] JVET-N0413에서 제공하는 드래프트의 상단(top)에 있는 변경 사항은 굵은 기울임체(bold italic)로 하이라이트 되어 제공된다. 삭제된 텍스트는 취소선으로 마크된다.

[0462] **5.1 실시예 1**

[0463] **i. 루마 인트라 예측 모드에 대한 유도 프로세스**

[0464] 본 프로세스에 대한 입력은:

[0465] - 루마 위치 (x_{Cb} , y_{Cb})는 현재 픽처의 탑-레프트 루마 샘플들과 관련된 현재 루마 코딩 블록의 탑-레프트 샘플을 지정함,

[0466] - 변수 cbWidth는 루마 샘플 내 현재 코딩 블록의 폭을 지정함,

[0467] - 변수 cbHeight는 루마 샘플 내 현재 코딩 블록의 높이를 지정함.

[0468] 이 프로세스에서, 루마 인트라 예측 모드 $IntraPredModeY[x_{Cb}][y_{Cb}]$ 가 유도된다.

[0469] 표 8-1은 인트라 예측 모드 $IntraPredModeY[x_{Cb}][y_{Cb}]$ 의 값과 연관된 이름을 지정한다.

[0470] 표 8-1 -인트라 예측 모드 및 연관된 이름의 사양

인트라 예측 모드	연관된 이름
0	INTRA_PLANAR
1	INTRA_DC
2..66	INTRA_ANGULAR2..INTRA_ANGULAR66
81..83	INTRA_LT_CCLM, INTRA_L_CCLM, INTRA_T_CCLM

[0471]

[0472] 주의 - : 인트라 예측 모드 INTRA_LT_CCLM, INTRA_L_CCLM 및 INTRA_T_CCLM 들은 크로마 컴포넌트에만 적용 가능하다.

[0473] IntraPredModeY[xCb][yCb]는 다음 순서의 단계에 의해 유도된다:

[0474] - 인접 위치 (xNbA, yNbA) 및 (xNbB, yNbB)는 각각 (xCb - 1, yCb + cbHeight - 1) 및 (xCb + cbWidth - 1, yCb - 1)와 동일하게 설정된다.

[0475] - A 또는 B 중 어느 하나에 의해 대체되는 X에 대해, 변수 candIntraPredModeX는 다음과 같이 유도된다:

[0476] ...

[0477] - 변수 ispDefaultMode1 및 ispDefaultMode2는 다음과 같이 유도된다:

[0478] ...

[0479] - $x = 0..5$ 를 갖는 candModeList[x]는 다음과 같이 유도된다:

[0480] ...

[0481] - IntraPredModeY[xCb][yCb]는 다음의 프로시저를 적용하여 유도된다:

[0482] - *bdpcm_flag[xCb][yCb]* 이 1인 경우, *IntraPredModeY[xCb][yCb]*는 (*bdpcm_dir_flag[xCb][yCb]* == 0? INTRA_ANGULAR18 : INTRA_ANGULAR50)와 동일하게 설정됨.

[0483] - 그렇지 않고, *intra_luma_mpm_flag[xCb][yCb]*이 1인 경우, *IntraPredModeY[xCb][yCb]*는 *candModeList[intra_luma_mpm_idx[xCb][yCb]]*와 동일하게 설정됨.

[0484] - 그렇지 않으면, *IntraPredModeY[xCb][yCb]*는 다음 순서의 단계에 의해 유도됨:

[0485] ...

[0486] $x = xCb..xCb + cbWidth - 1$ 및 $y = yCb..yCb + cbHeight - 1$ 인 변수 *IntraPredModeY[x][y]*는 *IntraPredModeY[xCb][yCb]*와 동일하게 설정됨.

[0487] 5.2 실시예 2

[0488] 8.4.2 루마 인트라 예측 모드를 위한 유도 프로세스

[0489] 본 프로세스에 대한 입력은:

[0490] - 루마 위치 (xCb , yCb)는 현재 픽처의 탑-레프트 루마 샘플들과 관련된 현재 루마 코딩 블록의 탑-레프트 샘플을 지정함,

[0491] - 변수 cbWidth는 루마 샘플 내 현재 코딩 블록의 폭을 지정함,

[0492] - 변수 cbHeight는 루마 샘플 내 현재 코딩 블록의 높이를 지정함.

[0493] 이 프로세스에서, 루마 인트라 예측 모드 *IntraPredModeY[xCb][yCb]* 가 유도된다.

[0494] Table 8-1은 인트라 예측 모드 *IntraPredModeY[xCb][yCb]*의 값과 연관된 이름을 지정한다.

[0495] 표 8-1 -인트라 예측 모드 및 연관된 이름의 사양

인트라 예측 모드	연관된 이름
0	INTRA_PLANAR
1	INTRA_DC
2..66	INTRA_ANGULAR2..INTRA_ANGULAR66
81..83	INTRA_LT_CCLM, INTRA_L_CCLM, INTRA_T_CCLM

[0496]

주의 - : 인트라 예측 INTRA_LT_CCLM, INTRA_L_CCLM 및 INTRA_T_CCLM 들은 크로마 컴포넌트에만 적용 가능하다.

[0498]

IntraPredModeY[xCb][yCb]는 다음 순서의 단계에 의해 유도된다:

[0499]

- *bdpcm_flag*[xCb][yCb]가 1인 경우, *IntraPredModeY*[xCb][yCb]는 (*bdpcm_dir_flag*[xCb][yCb] == 0? INTRA_ANGULAR18: INTRA_ANGULAR50)와 동일하게 설정된다.

[0500]

- 그렇지 않으면, *IntraPredModeY*[xCb][yCb]는 다음 순서의 단계를 적용함으로써 유도된다:

[0501]

- 인접 위치 (xNbA, yNbA) 및 (xNbB, yNbB)는 각각 (xCb - 1, yCb + cbHeight - 1) 및 (xCb + cbWidth - 1, yCb - 1)와 동일하게 설정된다.

[0502]

- A 또는 B 중 어느 하나에 의해 대체되는 X에 대해, 변수 *candIntraPredModeX*는 다음과 같이 유도된다:

[0503]

...

[0504]

- 변수 *ispDefaultMode1* 및 *ispDefaultMode2*는 다음과 같이 유도된다:

[0505]

...

[0506]

- $x = 0.5$ 인 *candModeList*[x]는 다음과 같이 유도된다:

[0507]

...

[0508]

- *IntraPredModeY*[xCb][yCb]는 다음의 프로시저를 적용하여 유도된다:

[0509]

- *intra_luma_mpm_flag*[xCb][yCb]가 1인 경우, *IntraPredModeY*[xCb][yCb]는 *candModeList*[*intra_luma_mpm_idx*[xCb][yCb]]와 동일하게 설정됨.

[0510]

- 그렇지 않으면, *IntraPredModeY*[xCb][yCb]는 다음 순서의 단계를 적용함으로써 유도됨:

[0511]

...

[0512]

$x = xCb..xCb + cbWidth - 1$ 및 $y = yCb..yCb + cbHeight - 1$ 인 변수 *IntraPredModeY*[x][y]는 *IntraPredModeY*[xCb][yCb]와 동일하게 설정된다.

[0513] **6. 참조문헌**

[0514]

ITU-T and ISO/IEC, "High efficiency video coding", Rec. ITU-T H.265 | ISO/IEC 23008-2 (02/2018).

[0515]

B. Bross, J. Chen, S. Liu, Versatile Video Coding (Draft 4), JVET-M1001, Jan.2019

[0516]

도 10은 비디오 프로세싱 장치(1000)의 블록도이다. 장치(1000)는 여기에 기재된 하나 이상의 방법을 구현하는 데 사용될 수 있다. 장치(1000)는 스마트폰, 태블릿, 컴퓨터, 사물인터넷(IoT) 수신기 등에서 구현될 수 있다. 장치(1000)는 하나 이상의 프로세서(1002), 하나 이상의 메모리(1004) 및 비디오 프로세싱 하드웨어(1006)를 포함할 수 있다. 프로세서(들)(1002)는 본 문서에 기재된 하나 이상의 방법을 구현하도록 구성될 수 있다. 메모리(들)(1004)는 여기에 기재된 방법 및 기술을 구현하는 데 사용되는 데이터 및 코드를 저장하는 데 사용될 수 있다. 비디오 프로세싱 하드웨어(1006)는 하드웨어 회로에서 여기에 기재된 일부 기술을 구현하는 데 사용될 수 있다.

[0517]

도 11은 비디오 프로세싱의 일 예의 방법(1100)에 대한 순서도이다. 방법(1100)은 차동 코딩 모드를 이용하여 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 변환을 수행하고 공존 규칙에 기초하여 인트라

예측 모드를 선택적으로 사용하는 것을 포함한다. 인트라 예측 모드는 현재 비디오 블록의 샘플에 대한 예측을 생성하는 데 사용된다. 차동 코딩 모드는 차동 펄스 코딩 변조 표현을 사용하여 픽셀의 예측으로부터 양자화된 잔차 블록을 나타내는데 사용된다.

- [0518] 도 15는 본 개시의 기술을 활용할 수 있는 예제 비디오 코딩 시스템(100)을 예시하는 블록도이다. 도 15에 도시된 바와 같이, 비디오 코딩 시스템(100)은 소스 디바이스(110)와 목적 디바이스(120)를 포함할 수 있다. 소스 디바이스(110)는 인코딩된 비디오 데이터를 생성하고, 비디오 인코딩 디바이스로 참조될 수 있다. 목적 디바이스(120)는 소스 디바이스(110)에 의해 생성된 인코딩된 비디오 데이터를 디코딩할 수 있고, 비디오 디코딩 장치로 참조될 수 있다. 소스 디바이스(110)는 비디오 소스(112), 비디오 인코더(114) 및 입력/출력(I/O) 인터페이스(116)를 포함할 수 있다.
- [0519] 비디오 소스(112)는 비디오 캡처 장치, 비디오 콘텐츠 공급자로부터 비디오 데이터를 수신하는 인터페이스, 비디오 데이터를 생성하기 위한 컴퓨터 그래픽 시스템 또는 이러한 소스의 조합을 포함할 수 있다. 상기 비디오 데이터는 하나 이상의 픽처를 포함할 수 있다. 비디오 인코더(114)는 비디오 소스(112)의 비디오 데이터를 인코딩하여 비트스트림을 생성한다. 비트스트림에는 비디오 데이터의 코딩된 표현을 형성하는 비트들의 시퀀스가 포함될 수 있다. 비트스트림에는 코딩된 픽처 및 연관 데이터가 포함될 수 있다. 코딩된 픽처는 픽처의 코딩된 표현(representation)이다. 연관(associated) 데이터에는 시퀀스 파라미터 세트, 픽처 파라미터 세트 및 기타 선택 구조가 포함될 수 있다. I/O 인터페이스(116)는 변조기/복조기(모뎀) 및/또는 송신기를 포함할 수 있다. 인코딩된 비디오 데이터는 네트워크(130a)를 거쳐, I/O 인터페이스(116)를 통해 목적 디바이스(120)로 직접 전송될 수 있다. 인코딩된 비디오 데이터는 목적 디바이스(120)에 의한 액세스를 위해 저장 매체/서버(130b)에 저장될 수도 있다.
- [0520] 목적 디바이스(120)는 I/O 인터페이스(126), 비디오 디코더(124) 및 디스플레이 장치(122)를 포함할 수 있다.
- [0521] I/O 인터페이스(126)는 수신기 및/또는 모뎀을 포함할 수 있다. I/O 인터페이스(126)는 소스 디바이스(110) 또는 저장 매체/서버(130b)로부터 인코딩된 비디오 데이터를 획득할 수 있다. 비디오 디코더(124)는 인코딩된 비디오 데이터를 디코딩할 수 있다. 디스플레이 디바이스(122)는 디코딩된 비디오 데이터를 사용자에게 표시할 수 있다. 디스플레이 디바이스(122)는 목적 디바이스(120)와 통합될 수 있거나, 외부 디스플레이 디바이스와 인터페이스하도록 구성된 목적 디바이스(120)의 외부에 위치할 수 있다.
- [0522] 비디오 인코더(114) 및 비디오 디코더(124)는 고효율 비디오 코딩(HEVC) 표준, 다목적 비디오 코딩(VVC) 표준 및 기타 현재 및/또는 추가 표준과 같은 비디오 압축 표준에 따라 동작할 수 있다.
- [0523] 도 16은 도 15에 도시된 시스템(100)에서 비디오 인코더(114)일 수 있는 비디오 인코더(200)의 예를 보여주는 블록도이다.
- [0524] 비디오 인코더(200)는 본 개시의 기술 중 어느 또는 전부를 수행하도록 구성될 수 있다. 도 16의 예에서, 비디오 인코더(200)는 복수의 기능 컴포넌트를 포함한다. 본 개시에 기재된 기술은 비디오 인코더(200)의 다양한 컴포넌트들 간에 공유될 수 있다. 일부 예에서, 프로세서는 본 개시에 기재된 임의 또는 전부를 수행하기 위해 구성될 수 있다.
- [0525] 비디오 인코더(200)의 기능 컴포넌트는 파티션 유닛(201)와, 모드 선택 유닛(203), 모션 추정 유닛(204), 모션 보상 유닛(205) 및 인트라 예측 유닛(206)을 포함할 수 있는 예측 유닛(202)과, 잔차 생성 유닛(207), 변환 유닛(208), 양자화 유닛(209), 역양자화 유닛(210), 역변환 유닛(211), 재구성 유닛(212), 버퍼(213), 및 엔트로피 인코딩 유닛(214)를 포함할 수 있다.
- [0526] 다른 예에서, 비디오 인코더(200)에는 더 많거나, 적거나, 다른 기능 컴포넌트가 포함될 수 있다. 예를 들어, 예측 유닛(202)은 인트라 블록 커피(IBC) 유닛을 포함할 수 있다. IBC 유닛은 IBC 모드에서 적어도 하나의 레퍼런스 픽처가 현재 비디오 블록이 있는 픽처인 경우의 예측을 수행할 수 있다.
- [0527] 더욱이, 모션 추정부(204) 및 모션 보정부(205)와 같은 일부 컴포넌트는 고도로(highly) 통합될 수 있으나, 도 16의 예에서는 설명의 목적을 위해 분리된 것으로 표현되었다.
- [0528] 파티션 디바이스(201)는 픽처를 하나 이상의 비디오 블록으로 파티셔닝(partitioning)할 수 있다.
- [0529] 비디오 인코더(200) 및 비디오 디코더(300)는 다양한 비디오 블록 크기를 지원할 수 있다. 모드 선택 유닛(203)은 오류 결과에 기초하여 코딩 모드, 예를 들면, 인트라 또는 인터 등을 선택하고, 결과인 인트라 또는 인터 코딩된 블록을, 잔차 생성 유닛(207)에 제공하여 잔차 블록 데이터를 생성하고, 재구성 유닛(212)으로 제공

하여 레퍼런스 픽처로 사용하기 위한 인커딩된 블록을 재구성한다. 일부 예에서, 모드 선택 유닛(203)은 인터라 및 인터 예측(CIIP) 모드의 조합을 선택할 수 있고, 이 모드에서 예측은 인터 예측 시그널 및 인터라 예측 시그널을 기반으로 한다. 모드 선택 유닛(203)은 또한, 인터 예측의 경우 블록에 대한 모션 벡터의 해상도(예: 서브 픽셀 또는 정수 픽셀 정밀도)를 선택할 수 있다.

- [0530] 현재 비디오 블록에서 인터 예측을 수행하기 위해, 모션 추정 유닛(204)는 버퍼(213)에서 현재 비디오 블록에 하나 이상의 레퍼런스 프레임을 비교하여 현재 비디오 블록에 대한 모션 정보를 생성할 수 있다. 모션 보상 유닛(205)는 현재 비디오 블록과 관련된 그림 이외의 버퍼(213)로부터의 모션 정보 및 디코딩된 픽처 샘플을 기반으로 현재 비디오 블록에 대한 예측 비디오 블록을 결정할 수 있다.
- [0531] 모션 추정 유닛(204) 및 모션 보상 유닛(205)는 현재 비디오 블록이 I 슬라이스, P 슬라이스 또는 B 슬라이스에 있는지 여부에 따라 현재 비디오 블록에 대해 서로 다른 작업을 수행할 수 있다.
- [0532] 일부 예에서, 모션 추정 유닛(204)는 현재 비디오 블록에 대한 단-방향(uni-directional) 예측을 수행할 수 있고, 모션 추정 유닛(204)는 현재 비디오 블록에 대한 레퍼런스 비디오 블록에 대한 리스트 0 또는 리스트 1의 레퍼런스 픽처를 검색할 수 있다. 그런 다음, 모션 추정 유닛(204)는 현재 비디오 블록과 레퍼런스 비디오 블록 사이의 공간 변위(spatial displacement)를 나타내는 레퍼런스 비디오 블록 및 모션 벡터를 포함하는 리스트 0 또는 리스트 1에서 레퍼런스 픽처를 나타내는 레퍼런스 인덱스를 생성할 수 있다. 모션 추정 유닛(204)는 레퍼런스 인덱스, 예측 방향 표시자 및 모션 벡터를 현재 비디오 블록의 모션 정보로 출력할 수 있다. 모션 보상 유닛(205)는 현재 비디오 블록의 동작 정보에 의해 표시된 레퍼런스 비디오 블록에 기초하여 현재 블록의 예측된 비디오 블록을 생성할 수 있다.
- [0533] 다른 예에서, 모션 추정 유닛(204)는 현재 비디오 블록에 대한 양-방향(bi-directional) 예측을 수행할 수 있고, 모션 추정 유닛(204)는 현재 비디오 블록에 대한 레퍼런스 비디오 블록에 대한 리스트 0에서 레퍼런스 픽처를 검색할 수 있으며, 또한 현재 비디오 블록에 대한 다른 레퍼런스 비디오 블록에 대한 리스트 1에서 레퍼런스 픽처를 검색할 수 있다. 그런 다음, 모션 추정 유닛(204)는 레퍼런스 비디오 블록과 현재 비디오 블록 사이의 공간 변위를 나타내는 레퍼런스 비디오 블록 및 모션 벡터를 포함하는 리스트 0 및 리스트 1에서 레퍼런스 픽처를 나타내는 레퍼런스 인덱스를 생성할 수 있다. 모션 추정 유닛(204)는 현재 비디오 블록의 모션 정보로서 현재 비디오 블록의 레퍼런스 인덱스 및 모션 벡터를 출력할 수 있다. 모션 보상 유닛(205)는 현재 비디오 블록의 동작 정보에 의해 표시된 레퍼런스 비디오 블록에 기초하여 현재 비디오 블록의 예측 비디오 블록을 생성할 수 있다.
- [0534] 일부 예에서, 모션 추정 유닛(204)는 디코더의 디코딩 처리를 위한 전체 모션 정보 세트를 출력할 수 있다.
- [0535] 일부 예에서, 모션 추정 유닛(204)는 현재 비디오에 대한 전체 모션 정보 세트를 출력하지 않을 수 있다. 오히려, 모션 추정 유닛(204)는 다른 비디오 블록의 동작 정보를 참조하여 현재 비디오 블록의 모션 정보를 시그널링할 수 있다. 예를 들어, 모션 추정 유닛(204)는 현재 비디오 블록의 모션 정보가 인접 비디오 블록의 동작 정보와 충분히 유사하다고 결정할 수 있다.
- [0536] 일 예에서, 모션 추정 유닛(204)는 현재 비디오 블록과 연결된 신텍스 구조에서 현재 비디오 블록이 다른 비디오 블록과 동일한 모션 정보를 가지고 있음을 비디오 디코더(300)에 나타내는 값을 나타낼 수 있다.
- [0537] 또 다른 예에서, 모션 추정 유닛(204)는 현재 비디오 블록과 연관된 신텍스 구조에서, 다른 비디오 블록 및 모션 벡터 차이(MVD)를 식별할 수 있다. 모션 벡터 차이는 현재 비디오 블록의 모션 벡터와 표시된 비디오 블록의 모션 벡터 간의 차이를 나타낸다. 비디오 디코더(300)는 현재 비디오 블록의 모션 벡터를 결정하기 위해 표시된 비디오 블록의 모션 벡터 및 모션 벡터 차이를 사용할 수 있다.
- [0538] 위에서 설명한 바와 같이, 비디오 인코더(200)는 모션 벡터를 예측적으로 시그널링할 수 있다. 비디오 인코더(200)에 의해 구현될 수 있는 예측 시그널링 기술의 두 가지 예는, 어드밴스드 모션 벡터 예측(AMVP) 및 병합 모드 시그널링을 포함한다.
- [0539] 인터라 예측 유닛(206)은 현재 비디오 블록에 대한 인터라 예측을 수행할 수 있다. 인터라 예측 유닛(206)가 현재 비디오 블록에 대한 인터라 예측을 수행하는 경우, 인터라 예측 유닛(206)는 동일 픽처에서 다른 비디오 블록의 디코딩된 샘플을 기반으로 현재 비디오 블록에 대한 예측 데이터를 생성할 수 있다. 현재 비디오 블록에 대한 예측 데이터에는 예측된 비디오 블록 및 다양한 신텍스 요소가 포함될 수 있다.
- [0540] 잔차 생성 유닛(207)는 현재 비디오 블록으로부터 예측된 비디오 블록의 예측 비디오 블록(예를 들어, 마이너스

기호로 표시)을 빼서 현재 비디오 블록에 대한 잔차 데이터를 생성할 수 있다. 현재 비디오 블록의 잔차 데이터는 현재 비디오 블록에서 샘플의 상이한 샘플 컴포넌트에 대응하는 잔차 비디오 블록을 포함할 수 있다.

- [0541] 다른 예에서, 현재 비디오 블록에 대한 현재 비디오 블록에 대한 잔차 데이터가 없을 수 있고, 예를 들어 스킵(skip) 모드에서, 잔차 생성 유닛(207)는 뺄셈 동작을 수행하지 않을 수 있다.
- [0542] 변환 프로세싱 유닛(208)는 현재 비디오 블록과 연결된 잔차 비디오 블록에 하나 이상의 변환을 적용하여 현재 비디오 블록에 대해 하나 이상의 변환 계수 비디오 블록을 생성할 수 있다.
- [0543] 변환 프로세싱 유닛(208)는 현재 비디오 블록과 관련된 변환 계수 비디오 블록을 생성한 후, 양자화 유닛(209)는 현재 비디오 블록과 연관된 하나 이상의 양자화 파라미터(QP) 값을 기반으로 현재 비디오 블록과 관련된 변환 계수 비디오 블록을 양자화할 수 있다.
- [0544] 역양자화 유닛(210)와 역변환부 유닛(211)는 변환 계수 비디오 블록으로부터 잔차 비디오 블록을 재구성하기 위해, 변환 계수 비디오 블록에 역양자화 및 역변환을 각각 적용할 수 있다. 재구성 유닛(212)는 예측 유닛(202)에 의해 생성된 하나 이상의 예측된 비디오 블록으로부터 대응하는 샘플에 재구성된 잔차 비디오 블록을 추가하여 버퍼(213)에 저장하기 위한 현재 블록과 연관된 재구성된 비디오 블록을 생성할 수 있다.
- [0545] 재구성 유닛(212)가 비디오 블록을 재구성한 후, 비디오 블록에서 비디오 차단(blocking) 아티팩트를 줄이기 위해 루프 필터링 동작이 수행된다.
- [0546] 엔트로피 인코딩 유닛(214)은 비디오 인코더(200)의 다른 기능 컴포넌트로부터 데이터를 수신할 수 있다. 엔트로피 인코딩 유닛(214)가 데이터를 수신하는 경우, 엔트로피 인코딩 유닛(214)는 엔트로피 인코딩된 데이터를 생성하고 엔트로피 인코딩된 데이터를 포함하는 비트스트림을 출력하기 위해 하나 이상의 엔트로피 인코딩 동작을 수행할 수 있다.
- [0547] 도 17은 도 15에 도시된 시스템(100)에서 비디오 디코더(114)일 수 있는 비디오 디코더(300)의 예를 보여주는 블록도이다.
- [0548] 비디오 디코더(300)는 본 개시의 기술 중 어느 또는 전부를 수행하도록 구성될 수 있다. 도 17의 예에서, 비디오 디코더(300)는 복수의 기능 컴포넌트를 포함한다. 본 개시에 기재된 기술은 비디오 디코더(300)의 다양한 컴포넌트들 간에 공유될 수 있다. 몇몇 실시예에서, 본 문서에서 설명된 임의의 또는 모든 기술을 수행하도록 프로세서가 구성될 수 있다.
- [0549] 도 17의 예에서, 비디오 디코더(300)에는 엔트로피 디코딩 유닛(301), 모션 보상 유닛(302), 인트라 예측 유닛(303), 역양자화 유닛(304), 역변환 유닛(305), 및 재구성 유닛(306) 및 버퍼(307)가 포함된다. 비디오 디코더(300)는, 일부 예에서, 비디오 인코더(200)에 대하여 기술된 인코딩 패스(pass)와 일반적으로 서로 주고 받는(reciprocal) 디코딩 패스(pass)를 수행할 수 있다(도 16).
- [0550] 엔트로피 디코딩 유닛(301)는 인코딩된 비트스트림을 검색할 수 있다. 인코딩된 비트스트림에는 엔트로피 코딩된 비디오 데이터(예: 비디오 데이터의 인코딩된 블록)가 포함될 수 있다. 엔트로피 디코딩 유닛(301)는 엔트로피 코딩된 비디오 데이터를 디코딩할 수 있고, 엔트로피 디코딩된 비디오 데이터로부터 모션 보상 유닛(302)는 모션 벡터, 모션 벡터 정밀도, 레퍼런스 픽처 리스트 인덱스 및 기타 모션 정보를 포함하는 모션 정보를 결정할 수 있다. 모션 보상 유닛(302)는, 예를 들어 AMVP 및 병합 모드를 수행하여 이러한 정보를 결정할 수 있다.
- [0551] 모션 보상 유닛(302)는 보간 필터를 기반으로 보간(interpolation)을 수행하여 모션 보정 블록을 생성할 수 있다. 서브 픽셀 정밀도와 함께 사용되는 보간 필터에 대한 식별자가 신텍스 요소에 포함될 수 있다.
- [0552] 모션 보상 유닛(302)는 비디오 블록을 인코딩하는 동안 비디오 인코더(20)에서 사용하는 보간 필터를 사용하여 레퍼런스 블록의 서브 정수 픽셀에 대한 보간 값을 계산할 수 있다. 모션 보상 유닛(302)는 수신된 신텍스 정보에 따라 비디오 인코더(200)에서 사용하는 보간 필터를 결정하고 보간 필터를 사용하여 예측 블록을 생성할 수 있다.
- [0553] 모션 보상 유닛(302)는 인코딩된 비디오 시퀀스의 프레임 및/또는 슬라이스를 인코딩하는 데 사용되는 블록의 크기, 인코딩된 비디오 시퀀스의 각 매크로 블록이 파티셔닝되는 방식을 설명하는 파티션 정보, 각 파티션이 인코딩된 방법, 각 파티션이 인코딩되는 방식을 나타내는 모드, 각 인터-인코딩된 블록에 대한 하나 이상의 레퍼런스 프레임(및 레퍼런스 프레임 목록) 및 인코딩된 비디오 시퀀스를 디코딩하는 다른 정보들을 결정하기 위해

일부 선택스 정보를 사용할 수 있다.

- [0554] 인트라 예측 유닛(303)는, 예를 들어 비트스트림에서 수신된 인트라 예측 모드를 사용하여 공간적으로 인접한 블록으로부터 예측 블록을 형성할 수 있다. 역양자화 유닛(303)는 비트스트림에서 제공되고 엔트로피 디코딩 유닛(301)에서 디코딩된 양자화된 비디오 블록 계수를 역 양자화(예를 들어, 디-퀀타이즈(de-quantize))한다. 역변환 유닛(303)는 역변환을 적용한다.
- [0555] 재구성 유닛(306)는 모션 보상 유닛(202) 또는 인트라 예측 유닛(303)에 의해 생성된 해당 예측 블록과 잔차 블록을 합산하여 디코딩된 블록을 형성할 수 있다. 원하는 경우, 차단 해제(deblocking) 필터를 적용하여 차단 아티팩트를 제거하기 위해 디코딩된 블록을 필터링할 수도 있다. 디코딩된 비디오 블록은 버퍼(307)에 저장되고, 이는 이후의 모션 보상/인트라 예측에 대한 레퍼런스 블록을 제공하고, 디스플레이 디바이스에서 재생하기 위한 디코딩된 비디오를 생성한다.
- [0556] 몇몇 실시예에서, ALWIP 모드 또는 MIP 모드에서, 현재 비디오 블록에 대한 예측 블록은 행 및 열과 관련된 평균, 그 다음에 행렬 곱셈이, 그 다음에 예측 블록을 결정하는 보간에 의해 결정된다.
- [0557] 도 18은 여기에 개시된 다양한 기술들이 구현될 수 있는 예시적인 비디오 프로세싱 시스템(video processing system)(2100)을 도시하는 블록도이다. 다양한 구현에는 시스템(2100)의 컴포넌트의 일부 또는 전부가 포함될 수 있다. 시스템(2100)은 비디오 콘텐츠를 수신하기 위한 입력(1302)을 포함할 수 있다. 비디오 콘텐츠는, 예를 들어, 8 또는 10비트 다중 컴포넌트 픽셀 값과 같이 원시(raw) 또는 압축되지 않은(uncompressed) 포맷으로 수신될 수 있고, 또는 압축 또는 인코딩된 형식으로 수신될 수도 있다. 입력(2102)은 네트워크 인터페이스, 주변 버스 인터페이스 또는 스토리지 인터페이스를 나타낼 수 있다. 네트워크 인터페이스의 예로는 이더넷, 수동 광 네트워크(PON) 등과 같은 유선 인터페이스와 Wi-Fi 또는 셀룰러 인터페이스와 같은 무선 인터페이스가 있다.
- [0558] 시스템(2100)은 본 문서에 설명된 다양한 코딩 또는 인코딩 방법을 구현할 수 있는 코딩 컴포넌트(2104)를 포함할 수 있다. 코딩 컴포넌트(2104)는 입력(2102)에서 코딩 컴포넌트(2104)의 출력으로 비디오의 평균 비트레이트를 감소시킬 수 있으며, 비디오의 코딩된 표현을 생성할 수 있다. 따라서, 코딩 기법을 비디오 압축 또는 비디오 트랜스코딩 기술이라고도 한다. 코딩 컴포넌트(2104)의 출력은 컴포넌트(2106)에 의해 표현되는 바와 같이, 연결된 통신을 통해 저장되거나 전송될 수 있다. 입력(2102)에서 수신된 비디오의 저장되거나 통신된 비트스트림(또는 코딩된) 표현은 디스플레이 인터페이스(2110)로 전송되는 픽셀 값 또는 디스플레이 가능한 비디오를 생성하기 위해 컴포넌트(2108)에 의해 사용될 수 있다. 비트스트림 표현에서 사용자가 볼 수 있는 비디오를 생성하는 프로세스를 비디오 압축해제(video decompression)라고도 한다. 또한 특정 비디오 처리 작업을 "코딩" 작업 또는 툴로 지칭하지만, 코딩 툴 또는 작업이 인코더에서 사용되고 코딩 결과를 되돌리는 대응하는 디코딩 툴 또는 작업이 디코더에 의해 수행되는 것으로 인식될 것이다.
- [0559] 주변 버스 인터페이스 또는 디스플레이 인터페이스의 예로는 범용 직렬 버스(USB) 또는 고화질 멀티미디어 인터페이스(HDMI) 또는 디스플레이 포트 등이 포함될 수 있다. 스토리지 인터페이스의 예로는 SATA(직렬 첨단 기술 첨부 파일), PCI, IDE 인터페이스 등이 있다. 본 문서에 기재된 기술은 휴대전화, 노트북, 스마트폰 또는 디지털 데이터 처리 및/또는 비디오 디스플레이를 수행할 수 있는 기타 장치와 같은 다양한 전자 기기에 구현될 수 있다.
- [0560] 개시된 기술의 일부 실시예에는 비디오 프로세싱 툴 또는 모드를 가능하게 하기 위한 결정 또는 결정을 내리는 것이 포함된다. 예를 들어 비디오 프로세싱 툴 또는 모드를 사용하도록 설정하면 인코더가 비디오 블록 프로세싱에서 툴 또는 모드를 사용하거나 구현하지만 툴 또는 모드의 사용에 따라 결과 비트스트림을 반드시 수정할 필요는 없다. 즉, 비디오 블록에서 비디오의 비트스트림 표현으로 변환하면 결정 또는 결정에 따라 인에이블될 때 비디오 프로세싱 툴 또는 모드를 사용한다. 또 다른 예에서 비디오 프로세싱 툴 또는 모드를 사용하도록 설정하면 디코더는 비디오 프로세싱 툴 또는 모드에 따라 비트스트림이 수정되었다는 사실을 알고 비트스트림을 프로세싱 한다. 즉, 비디오의 비트스트림 표현에서 비디오 블록으로의 변환은 결정 또는 결정에 따라 인에이블된 비디오 프로세싱 툴 또는 모드를 사용하여 수행된다.
- [0561] 개시된 기술의 일부 실시예에는 비디오 프로세싱 툴 또는 모드를 디스에이블하는 결정 또는 결정을 내리는 것이 포함된다. 예를 들어 비디오 프로세싱 툴 또는 모드를 사용하지 않는 경우 인코더는 비디오 블록을 비디오 블록으로 변환하는 데 툴 또는 모드를 사용하지 않는다. 또 다른 예에서 비디오 프로세싱 툴 또는 모드를 사용하지 않도록 설정하면 디코더는 결정 또는 결정에 따라 디스에이블된 비디오 프로세싱 툴 또는 모드를 사용하여 비트스트림이 수정되지 않았다는 사실을 알고 비트스트림을 프로세싱한다.

- [0562] 도 19는 일 예로서 비디오 프로세싱 방법(1900)에 대한 흐름도를 도시한다. 방법(1900)은, 차동 코딩 모드를 사용하여, 현재 비디오 블록과 연관되어 저장되는 제1 인트라 코딩 모드를 결정함으로써 비디오의 상기 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환을 수행하는 단계를 포함하고, 상기 현재 비디오 블록과 연관된 상기 제1 인트라 코딩 모드는 상기 차동 코딩 모드에 의해 사용되는 제2 예측 모드에 따라 결정되며, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 인트라 예측의 양자화된 잔차와 상기 양자화된 잔차의 예측 간의 차이는 차동 펄스 코딩 변조(Differential Pulse Coding Modulation; DPCM) 표현을 사용하여 상기 현재 비디오 블록의 비트스트림 표현으로 나타난다.
- [0563] 방법(1900)에 대한 일부 실시예에서, 상기 제1 인트라 코딩 모드는, 상기 제2 예측 모드가 수직 예측 모드임에 응답하여 수직 인트라 예측 모드로 추론(inferred)된다. 방법(1900)에 대한 일부 실시예에서, 상기 제1 인트라 코딩 모드는, 상기 제2 예측 모드가 수평 예측 모드임에 응답하여 수평 인트라 예측 모드로 추론된다. 방법(1900)에 대한 일부 실시예에서, 상기 제1 인트라 코딩 모드는, 상기 제2 예측 모드가 탑-레프트(top-left) 대각선(diagonal) 예측 모드임에 응답하여 탑-레프트 대각선 인트라 예측 모드로 추론된다.
- [0564] 방법(1900)에 대한 일부 실시예에서, 상기 제2 예측 모드는 상기 제1 인트라 코딩 모드와 동일한 것으로 추론된다. 방법(1900)에 대한 일부 실시예에서, 상기 제2 예측 모드는 상기 제1 인트라 코딩 모드와 동일한 것으로 추론된다. 방법(1900)에 대한 일부 실시예에서, 상기 제1 인트라 코딩 모드는 최대 가능(most probable) 모드 내의 모드에 기초하여 추론된다. 방법(1900)에 대한 일부 실시예에서, 상기 제1 인트라 코딩 모드는 미리 정의된 인트라 예측 모드이다. 방법(1900)에 대한 일부 실시예에서, 상기 미리 정의된 인트라 예측 모드는 플래나(Planar) 모드를 포함한다. 방법(1900)에 대한 일부 실시예에서, 상기 미리 정의된 인트라 예측 모드는 DC 모드를 포함한다. 방법(1900)에 대한 일부 실시예에서, 미리 정의된 인트라 예측 모드는 수직 모드를 포함한다. 방법(1900)에 대한 일부 실시예에서, 미리 정의된 인트라 예측 모드는 수평 모드를 포함한다. 방법(1900)에 대한 일부 실시예에서, 상기 비디오의 추가 비디오 블록들은 상기 제1 인트라 코딩 모드로 코딩되고, 상기 현재 비디오 블록은 시간 영역에서 상기 추가 비디오 블록들에 선행(precede)한다. 방법(1900)에 대한 일부 실시예에서, 최대 가능 모드(Most Probable Mode; MPM) 리스트는 상기 제1 인트라 코딩 모드를 사용하여 상기 추가 비디오 블록들에 대해 구성된다.
- [0565] 도 20은 일 예로서 비디오 프로세싱 방법(2000)에 대한 흐름도를 도시한다. 방법(2000)은, 규칙에 따라, 비디오의 현재 비디오 블록과 상기 현재 비디오 블록의 비트 스트림 표현 간의 변환 중에 차동 코딩 모드에 의해 사용되는 인트라 코딩 모드를 결정하는 단계(2002)를 포함한다. 동작(2004)는, 상기 결정에 기초하여, 상기 차동 코딩 모드를 사용하여 상기 현재 비디오 블록과 상기 현재 비디오 블록의 상기 비트 스트림 표현 사이의 변환을 수행하는 단계를 포함하고, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 인트라 예측의 양자화된 잔차와 상기 양자화된 잔차의 예측 간의 차이는 차동 펄스 코딩 변조(DPCM) 표현을 이용하여 현재 비디오 블록에 대한 비트스트림 표현으로 나타나며, 상기 양자화된 잔차의 상기 예측은 상기 인트라 코딩 모드에 따라 수행된다.
- [0566] 방법(2000)에 대한 몇몇 실시예에서, 상기 규칙은 상기 현재 비디오 블록의 코딩을 위해 사용되는 컬러 포맷에 기초하여 상기 인트라 코딩 모드가 결정되는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 상기 규칙은 시퀀스 매개 변수 세트(SPS), 비디오 매개 변수 세트(VPS), 픽처 매개 변수 세트(PPS), 픽처 헤더, 슬라이스 헤더, 타일 그룹 헤더, 최대 코딩 유닛(LCU) 행 또는 LCU 그룹 내에서 시그널링된 메시지에 기초하여 상기 인트라 코딩 모드가 결정되는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 상기 규칙은 상기 인트라 코딩 모드가 상기 차동 코딩 모드에서 수행되는 방향(direction)을 나타내는 플래그(flag)에 기초하여 결정되는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 상기 규칙은 상기 인트라 코딩 모드가 상기 양자화된 잔차의 상기 예측의 방향을 나타내는 플래그에 기초하여 결정되는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 상기 규칙은 상기 현재 비디오 블록 또는 상기 비디오 블록의 인접(neighboring) 비디오 블록 중 어느 하나의 블록 디멘전(dimension)에 기초하여 상기 인트라 코딩 모드가 결정되는 것을 지정한다.
- [0567] 방법(2000)에 대한 일부 실시예에서, 규칙은 현재 비디오 블록 또는 현재 비디오 블록의 인접 비디오 블록의 형상에 따라 인트라 코딩 모드가 결정된다는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 규칙은 현재 비디오 블록의 가장 가능성 있는 모드(MPM) 또는 현재 비디오 블록의 인접 비디오 블록에 기초하여 인트라 코딩 모드가 결정된다는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 규칙은 현재 비디오 블록의 인접 비디오 블록의 인트라 예측 모드 또는 인트라 예측 모드에 기초하여 인트라 코딩 모드가 결정된다는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 규칙은 현재 비디오 블록의 인접 비디오 블록의 모션 벡터에 기초하여 인트라 코딩 모드가 결정된다는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 규칙은 현재 비디오 블록의

인접 비디오 블록이 차동 코딩 모드를 사용하여 코딩되는지 여부를 나타내는 지표에 기초하여 인트라 코딩 모드가 결정된다는 것을 지정한다.

- [0568] 방법(2000)에 대한 몇몇 실시예에서, 규칙은 현재 비디오 블록 또는 현재 비디오 블록의 인접한 비디오 블록의 양자화 파라미터의 값에 기초하여 인트라 코딩 모드가 결정된다는 것을 지정한다. 방법(2000)에 대한 몇몇 실시예에서, 상기 규칙은 상기 현재 비디오 블록의 코딩을 위해 사용되는 컬러 포맷에 기초하여 상기 인트라 코딩 모드가 결정되는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 상기 규칙은 독립(separate) 또는 듀얼 코딩 트리 구조가 상기 현재 비디오 블록을 코딩하는데 사용되는지 여부에 기초하여 상기 인트라 코딩 모드가 결정되는 것을 지정한다. 방법(2000)에 대한 몇몇 실시예에서, 상기 규칙은 상기 현재 비디오 블록에 적용된 변환 타입(transform type)에 따라 상기 인트라 코딩 모드가 결정되는 것을 지정한다. 방법(2000)에 대한 일부 실시예에서, 상기 규칙은 상기 현재 비디오 블록과 연관된, 슬라이스(slice), 타일(tile) 그룹 타입, 또는 픽처 타입에 기초하여 상기 인트라 코딩 모드가 결정되는 것을 지정한다.
- [0569] 예제의 다음 목록은 추가 실시예에 대한 설명이다.
- [0570] 1. 비디오 처리 방법에 있어서, 차동 코딩 모드 및 선택적으로 공존 규칙에 기반한 인트라 예측 모드를 이용하여 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현(representation) 간의 변환을 수행하는 단계를 포함하고, 상기 인트라 예측 모드는 상기 현재 비디오 블록의 샘플들의 예측들을 생성하는데 이용되고, 상기 차동 코딩 모드는 차동 펄스 코딩 변조 표현을 이용하여 상기 픽셀들의 상기 예측으로부터 양자화된 잔차 블록을 표현하는데 사용되는, 비디오 처리 방법.
- [0571] 2. 예 1의 방법에서, 상기 인트라 예측 모드는 매트릭스 기반 인트라 예측 모드 (MIP)이고, 상기 공존 규칙은 상기 MIP의 허용된 모드의 일부로 상기 MIP를 제한하는, 비디오 처리 방법.
- [0572] 3. 예 2의 방법에서, 허용된 모드의 상기 일부는 수평 또는 수직 노말 인트라 모드를 포함하는, 비디오 처리 방법.
- [0573] 예 1-3의 추가 실시예는 섹션 4의 항목 1에 설명된다. 예를 들면, 차동 코딩 모드는 QR-BDPCM 코딩 모드의 현재 버전을 나타낼 수 있다.
- [0574] 4. 예 1의 방법에서, 상기 인트라 예측 모드는 비-수평 또는 비-수직 방향을 따른 예측을 포함한다.
- [0575] 5. 예 1 또는 4의 방법에서, 인트라 예측 모드는 플래나(planar) 또는 DC 예측 모드이다.
- [0576] 6. 예 1 또는 4의 방법에서, 인트라 예측 모드는 수직 또는 수평 예측 모드이다.
- [0577] 7. 예 1 또는 4의 방법에서, 인트라 예측 모드는 비트스트림 표현의 필드에 의해 식별된다.
- [0578] 8. 예 1 또는 4의 방법에서, 상기 인트라 예측 모드는 현재 비디오 블록 또는 인접 블록의 블록 디멘전에 의존한다.
- [0579] 9. 예 1 또는 4의 방법에서, 상기 인트라 예측 모드는 현재 블록 또는 인접 블록의 형상에 의존한다.
- [0580] 10. 예 1 또는 4의 방법에서, 상기 인트라 예측 모드는 현재 비디오 블록 또는 인접 비디오 블록이 인트라 예측 또는 인트라 예측을 사용하여 코딩되는지에 의존한다.
- [0581] 11. 예 1 또는 4의 방법에서, 상기 인트라 예측 모드는 인접 비디오 블록이 차동 코딩 모드를 사용하여 코딩되는지 여부에 의존한다.
- [0582] 12. 예 1 또는 4의 방법에서, 상기 인트라 예측 모드는 현재 비디오 블록 또는 인접 비디오 블록에 사용되는 양자화 파라미터의 값에 의존한다.
- [0583] 13. 예 1 또는 4의 방법에서, 여기서 인트라 예측 모드는 현재 비디오 블록을 코딩하는 데 사용되는 컬러 포맷에 의존한다.
- [0584] 14. 예 1 또는 4의 방법에서, 상기 인트라 예측 모드는 독립(separate) 또는 듀얼 코딩 트리 구조가 상기 현재 비디오 블록을 코딩하는데 사용되는지 여부에 의존한다.
- [0585] 예 4 내지 14의 추가 실시예는 섹션 4의 항목 2에서 제공된다.
- [0586] 15. 예 1의 방법에서, 현재 비디오 블록의 샘플에 대한 생성 예측은 인접한 비디오 영역의 비-주변(non-adjacent) 샘플로부터 수행된다.

- [0587] 16. 예 1의 방법에서, 상기 인트라 예측 모드는 인트라 블록 카피 병합 모드를 포함한다.
- [0588] 17. 예 1의 방법에서, 상기 인트라 예측 모드는 인트라 블록 카피 어디밴스드 모션 벡터 예측 모드를 포함한다.
- [0589] 18. 예 15 내지 17 중 어느 하나의 방법에서, 상기 인트라 예측 모드는 블록 벡터 또는 병합 인덱스로 표시된다.
- [0590] 예 15 내지 18의 추가 실시예는 섹션 4의 항목 3에서 제공된다.
- [0591] 19. 예 1의 방법에서, 상기 공존 규칙은 비트스트림 표현 내의 필드에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0592] 20. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록 또는 인접한 블록의 디멘전에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0593] 21. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록 또는 인접한 블록의 형상에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0594] 22. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록 또는 인접한 블록의 예측 모드에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0595] 23. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록 또는 인접한 블록의 최대 가능 모드에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0596] 24. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록 또는 인접한 블록의 모션 벡터에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0597] 25. 예 1의 방법에서, 상기 공존 규칙은 인접한 블록이 차동 코딩 모드를 사용하여 코딩되었는지 여부에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0598] 26. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록 또는 인접한 블록에 의해 사용되는 양자화 파라미터에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0599] 27. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록의 컬러 포맷에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0600] 28. 예 1의 방법에서, 상기 공존 규칙은 독립(separate) 또는 듀얼 코딩 트리 구조가 상기 현재 비디오 블록을 코딩하는데 사용되는지 여부에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0601] 29. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록에 적용된 변환에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0602] 30. 예 1의 방법에서, 상기 공존 규칙은 현재 비디오 블록의 슬라이스 타입 또는 타일 그룹 타입 또는 픽처 타입에 기초하여, 차동 코딩 모드 내의 시그널링된 인덱스로부터 인트라 예측 모드로의 매핑을 지정한다.
- [0603] 예 19-30의 추가 실시예는 섹션 4의 항목 2에서 제공된다.
- [0604] 31. 비디오 프로세싱 방법은, 현재 비디오 블록의 픽셀 예측으로부터 양자화된 잔차 블록이 차동 펄스 코딩 변조 표현을 사용하여 표현되는 차동 코딩 모드를 사용하여 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 변환을 수행하는 단계를 포함하고, 상기 예측의 제1 방향(direction) 또는 상기 차동 코딩 모드의 제2 방향은 상기 비트스트림 표현으로부터 추론될 수 있다.
- [0605] 32. 예 31의 방법에서, 픽셀의 상기 예측의 상기 제1 방향은 상기 예측에 사용되는 인트라 예측 모드로부터 암시적으로(implicitly) 추론할 수 있다.
- [0606] 33. 예 32의 방법에서, 상기 차동 코딩 모드의 상기 제2 방향은 상기 예측의 상기 제1 방향과 동일한 방향으로 추론할 수 있다.
- [0607] 34. 예 31의 방법에서, 상기 제2 방향은 예측에 사용되는 인트라 예측 모드로부터 추론할 수 있다.
- [0608] 35. 예 31의 방법에서, 상기 제2 방향은 현재 비디오 블록 또는 인접 블록의 디멘전 또는 현재 비디오 블록 또는 인접 블록의 형상으로부터 추론할 수 있다.

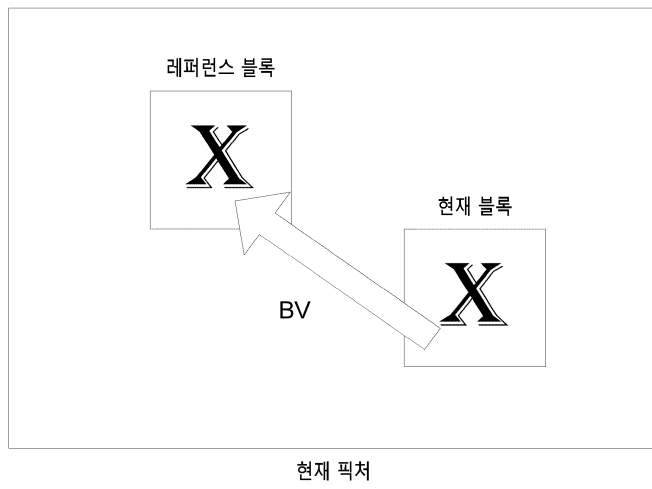
- [0609] 36. 예 31의 방법에서, 상기 제2 방향은 인접 블록의 모션 벡터로부터 추론할 수 있다.
- [0610] 37. 예 31의 방법에서, 상기 제2 방향은 현재 비디오 블록 또는 인접 블록의 최대 가능 모드로부터 추론할 수 있다.
- [0611] 38. 예 31의 방법에서, 상기 제2 방향은 인접 블록의 예측 모드로부터 추론할 수 있다.
- [0612] 39. 예 31의 방법에서, 상기 제2 방향은 인접 블록의 인트라 예측 모드로부터 추론할 수 있다.
- [0613] 40. 예 31의 방법에서, 상기 제2 방향은 인접 블록이 차동 코딩 모드를 사용하는지 여부로부터 추론할 수 있다.
- [0614] 예 31 내지 40의 추가 실시예는 섹션 4의 항목 3에 제공된다.
- [0615] 41. 비디오 프로세싱 방법은, 적용 가능성 규칙에 기초하여, 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환에 차동 코딩 모드가 적용 가능함을 결정하는 단계; 상기 차동 코딩 모드를 사용하여 현재 비디오 블록과 비트스트림 표현 간의 변환을 수행하는 단계를 포함하고, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 픽셀의 인트라 예측으로부터 양자화된 잔차 블록은 수평 또는 수직 방향과 다른 잔차 예측 방향에서 수행되는 차동 펄스 코딩 변조 표현을 사용하여 표현된다.
- [0616] 42. 예 41의 방법에서, 잔차 예측 방향은 45도 방향이다.
- [0617] 43. 예 41의 방법에서, 잔차 예측 방향은 135도 방향이다.
- [0618] 44. 예 41의 방법에서, 잔차 예측 방향은 비트스트림 표현 내의 필드 또는 현재 비디오 블록 또는 인접 블록 또는 현재 비디오 블록 또는 인접 블록의 형상과 관련이 있다.
- [0619] 예 41 내지 44의 추가 실시예는 섹션 4의 항목 7에서 제공된다.
- [0620] 45. 예 41의 방법에서, 상기 적용 가능성 규칙은 크로마 블록인 현재 비디오 블록으로 인한 차동 코딩 모드를 사용하도록 지정한다.
- [0621] 46. 예 41의 방법에서, 상기 적용 가능성 규칙은 현재 비디오 블록에 대한 잔차 예측 방향이 현재 비디오 블록에 대응하는 루마 블록과 동일한 방향임을 더 지정한다.
- [0622] 47. 예 41의 방법에서, 상기 적용 가능성 규칙은 크로스-컴포넌트 선형 모델(Cross-Component Linear Model; CCLM) 코딩 모드를 사용하지 않는 현재 비디오 블록으로 인해 차동 코딩을 사용하도록 지정한다.
- [0623] 예 45 내지 47의 추가 실시예는 섹션 4의 항목 8에서 제공된다.
- [0624] 48. 예 41의 방법에서, 상기 적용 가능성 규칙은 다른 컬러 컴포넌트에 대한 차동 코딩 모드의 적용 가능성으로부터 하나의 컬러 컴포넌트에 대한 차동 코딩 모드의 적용 가능성을 지정한다.
- [0625] 예 48의 추가 실시예는 섹션 4의 항목 12에서 제공된다.
- [0626] 49. 비디오 프로세싱 방법은, 차동 코딩 모드가 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환에 적용 가능한지 결정하는 단계; 및 상기 차동 코딩 모드의 구현 규칙을 사용하여 상기 현재 비디오 블록과 상기 비트스트림 표현 간의 변환을 수행하는 단계를 포함하고, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 픽셀의 인트라 예측으로부터 양자화된 잔차 블록은 수평 또는 수직 방향과 다른 잔차 예측 방향에서 수행되는 차동 펄스 코딩 변조 표현을 사용하여 표현된다.
- [0627] 50. 예 49의 방법에서, 상기 구현 규칙은 범위 내에서 양자화 잔차 블록의 값을 제한하도록 지정한다.
- [0628] 51. 예 49의 방법에서, 상기 구현 규칙은 클리핑을 사용하여 양자화된 잔차 블록을 획득하도록 지정한다.
- [0629] 예 49 내지 51의 추가 실시예는 섹션 4의 항목 9에서 제공된다.
- [0630] 52. 예 49의 방법에서, 상기 구현 규칙은 현재 비디오 블록의 마지막 행에서 현재 비디오 블록의 첫 번째 행까지 예측을 수행하도록 지정한다.
- [0631] 53. 예 49의 방법에서, 상기 구현 규칙은 현재 비디오 블록의 마지막 열에서 현재 비디오 블록의 첫 번째 열까지 예측을 수행하도록 지정한다.
- [0632] 예 52 내지 53의 추가 실시예는 섹션 4의 항목 10에서 제공된다.
- [0633] 54. 예 49의 방법에서, 상기 구현 규칙은 현재 비디오 블록의 서브 세트에만 차동 코딩 모드를 적용하도록 지정

한다.

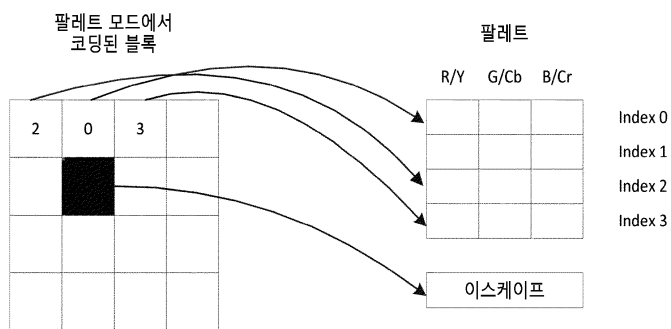
- [0634] 55. 예 54의 방법에서, 상기 서브 세트는 잔차의 k 왼쪽 열을 배제하는데, 여기에서 k는 블록의 픽셀 폭보다 작은 정수이다.
- [0635] 56. 예 54의 방법에서, 상기 서브 세트는 잔차의 k 탑(top) 열을 배제하는데, 여기에서 k는 블록의 픽셀 높이보다 작은 정수이다.
- [0636] 예 54 내지 56의 추가 실시예는 섹션 4의 항목 10에서 제공된다.
- [0637] 57. 예 49의 방법에서, 상기 구현 규칙은 세그먼트 별 기반으로 차동 코딩 모드를 변환에 적용하도록 지정한다.
- [0638] 예 57의 추가 실시예는 섹션 4의 항목 12에서 제공된다.
- [0639] 58. 비디오 프로세싱 방법은, 차동 코딩 모드가 현재 비디오 블록과 상기 현재 비디오 블록의 비트스트림 표현 간의 변환 중에 사용되는 차동 코딩 모드가 상기 현재 비디오 블록과 연관된 인트라 코딩 모드와 동일한지를 결정하는 단계; 및 상기 차동 코딩 모드의 구현 규칙을 사용하여 상기 현재 비디오 블록과 상기 비트스트림 표현 간의 변환을 수행하는 단계를 포함하고, 상기 차동 코딩 모드에서, 상기 현재 비디오 블록의 픽셀의 인트라 예측으로부터 양자화된 잔차 블록은 수평 또는 수직 방향과 다른 잔차 예측 방향에서 수행되는 차동 펄스 코딩 변조 표현을 사용하여 표현된다.
- [0640] 59. 예 58의 방법에서, 상기 차동 코딩 모드는 수직 인트라 예측 모드이다.
- [0641] 60. 예 58의 방법에서, 상기 차동 코딩 모드는 수평 인트라 예측 모드이다.
- [0642] 61. 예 58의 방법에서, 상기 차동 코딩 모드는 미리 정의된 인트라 예측 모드이다.
- [0643] 예 58 내지 61의 추가 실시예는 섹션 4의 항목 5에서 제공된다.
- [0644] 62. 예 1 내지 61 중 하나 이상에 기재된 방법을 구현하도록 구성된 프로세서를 포함하는, 비디오 프로세싱 장치.
- [0645] 63. 프로세서에 의해서 실행될 때, 프로세서로 하여금 예 1 내지 61 중 하나 이상의 방법을 구현하게 하는 코드가 저장된 컴퓨터 판독 가능한 매체.
- [0646] 본 문서의 예제 목록에서 변환이라는 용어는 현재 비디오 블록에 대한 비트스트림 표현의 생성을 참조하거나 비트스트림 표현에서 현재 비디오 블록을 생성할 수 있다. 비트스트림 표현은 연속적인(contiguous) 비트 그룹을 나타낼 필요가 없고, 헤더 필드 또는 코딩된 픽셀 값 정보를 나타내는 코드 워드에 포함된 비트로 나눌 수 있다.
- [0647] 위의 예에서 적용 규칙은 미리 정의되고 인코더 및 디코더에 알려질 수 있다.
- [0648] 본 문서에 설명된 바와 같이 개시된 기술이 비디오 인코더 또는 디코더에 구현되어 압축 효율을 향상시킬 수 있으며, 이는 본 문서에 설명된 바와 같이 인트라 코딩에서 차동 코딩 모드의 사용에 관한 다양한 구현 규칙의 사용을 포함하는 기술을 사용하여 압축 효율을 향상시킬 수 있다는 점을 인식할 수 있을 것이다.
- [0649] 본 문서에 기재된 개시 및 기타 솔루션, 예, 실시예, 모듈 및 기능적 연산은 디지털 전자 회로, 또는 컴퓨터 소프트웨어, 펌웨어 또는 하드웨어에서 구현될 수 있으며, 여기에는 이 문서 및 그 구조적 등가물 또는 그 중 하나 이상의 조합으로 구현될 수 있다. 개시된 및 기타 실시예는 하나 이상의 컴퓨터 프로그램 제품으로 구현될 수 있고, 즉, 컴퓨터 프로그램 지침의 하나 이상의 모듈을 컴퓨터 판독 가능한 매체에 인코딩 하여 실행하거나, 데이터 처리 장치의 작동을 제어할 수 있다. 컴퓨터가 읽을 수 있는 매체는 기계가 읽을 수 있는 저장 장치, 기계가 읽을 수 있는 스토리지 기관, 메모리 장치, 기계가 읽을 수 있는 전파 신호에 영향을 미치는 물질의 조성 또는 하나 이상의 조합일 수 있다. 용어 "데이터 프로세싱 장치"는 예를 들어 프로그래밍 가능한 프로세서, 컴퓨터 또는 다중 프로세서 또는 컴퓨터를 포함하여 데이터를 프로세싱하기 위한 모든 장치, 장치 및 컴퓨터를 포함한다. 이 장치에는 대응하는 컴퓨터 프로그램에 대한 실행 환경을 만드는 코드(예를 들어, 프로세서 펌웨어, 프로토콜 스택, 데이터베이스 관리 시스템, 운영 체제 또는 하나 이상의 조합)가 포함될 수 있다. 전파된 시그널은 인위적으로 생성된 시그널, 예를 들어, 기계에서 생성된 전기, 광학 또는 전자기 시그널이며, 이는 적합한 수신기 장치로 전송하기 위한 정보를 인코딩 하기 위해 생성된다.
- [0650] 컴퓨터 프로그램(프로그램, 소프트웨어, 소프트웨어 응용 프로그램, 스크립트 또는 코드라고도 함)은 컴파일된 언어를 비롯한 모든 형태의 프로그래밍 언어로 작성할 수 있으며 독립 실행형 프로그램 또는 컴퓨팅 환경에서

도면

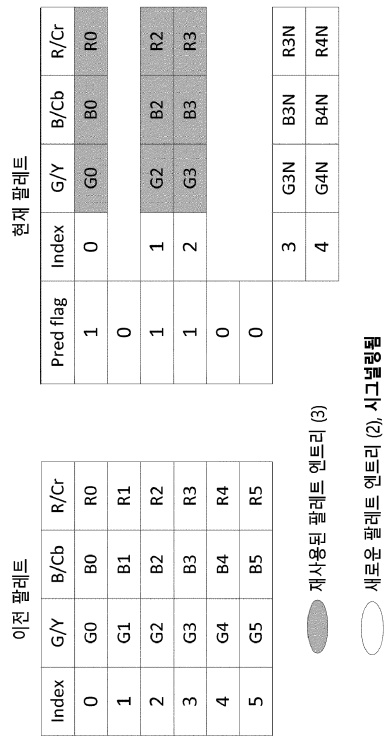
도면1



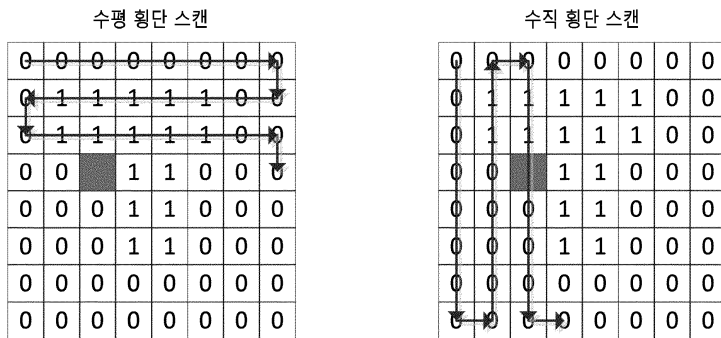
도면2



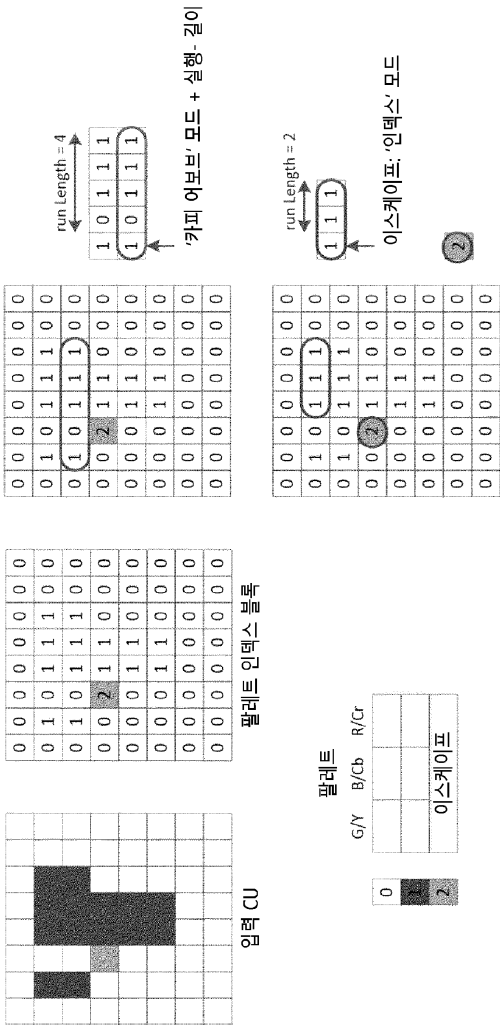
도면3



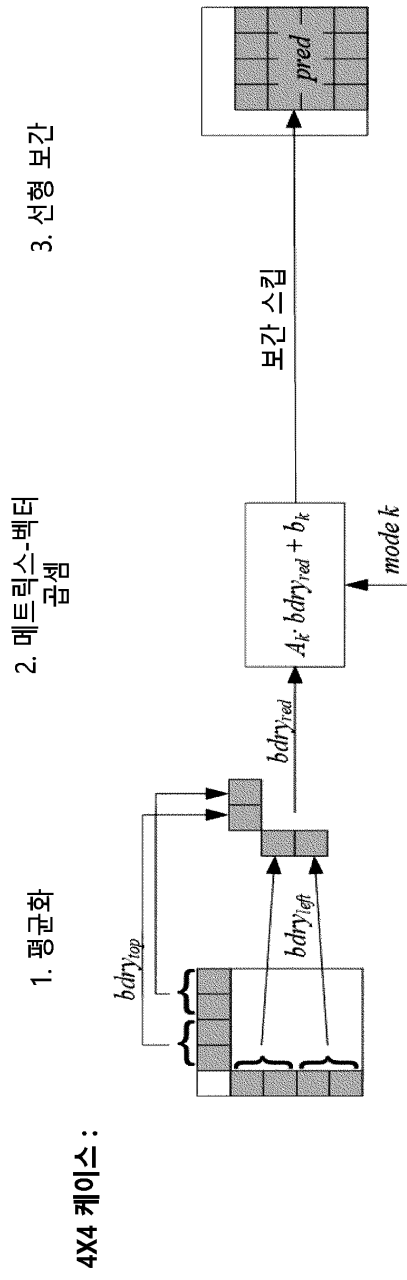
도면4



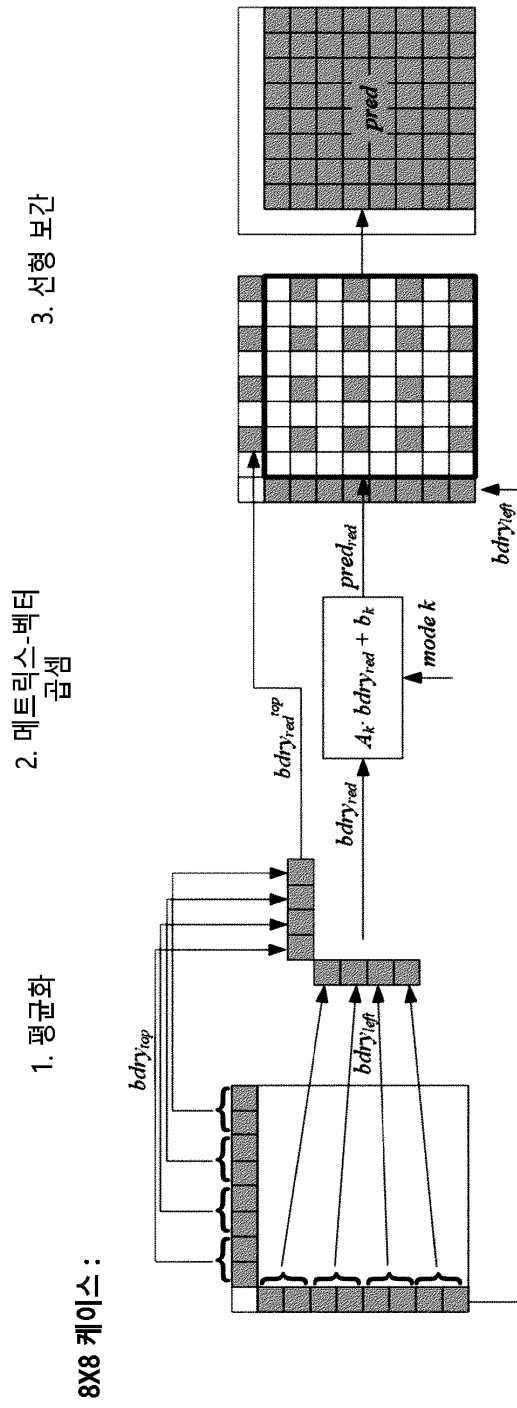
도면5



도면6

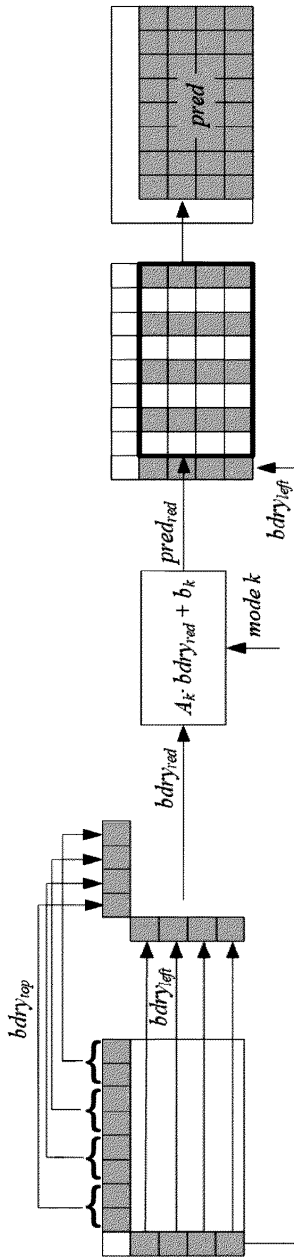


도면7

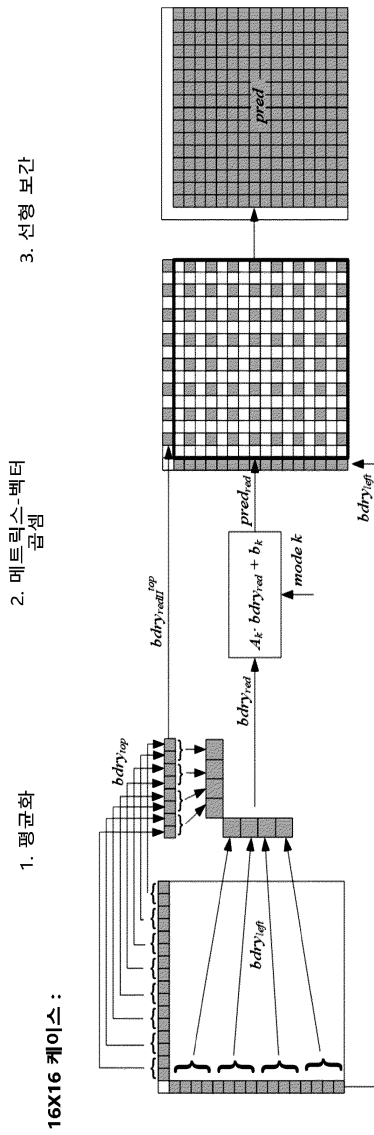


도면8

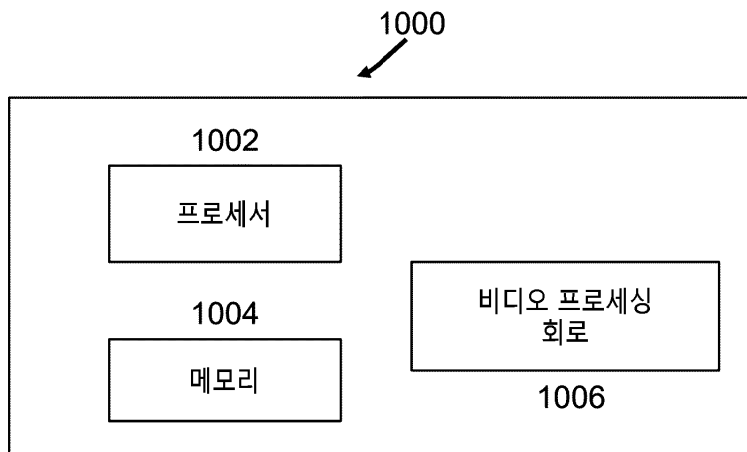
8x4 케이스 :
 1. 평균화
 2. 매트릭스-벡터
 곱셈
 3. 선형 보간



도면9

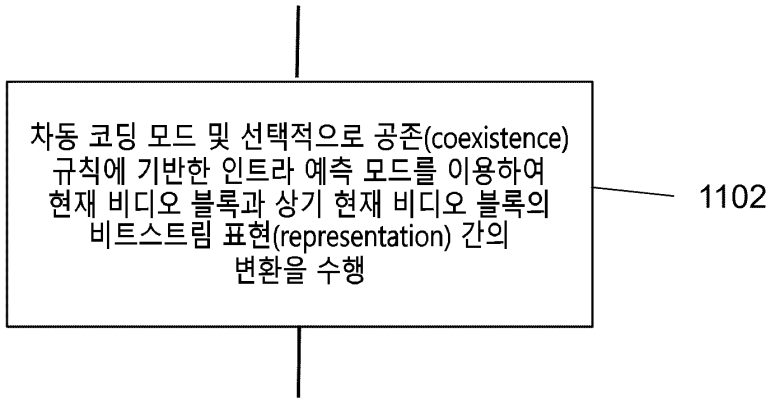


도면10

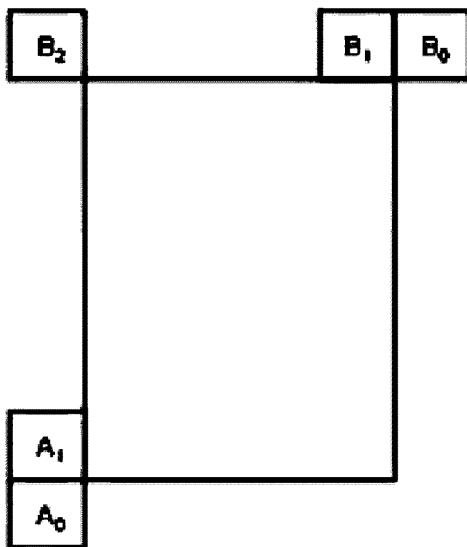


도면11

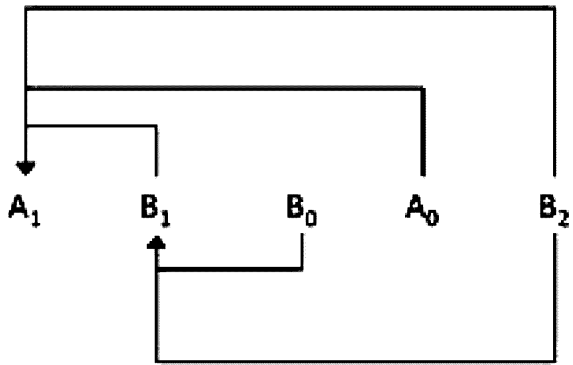
1100
↓



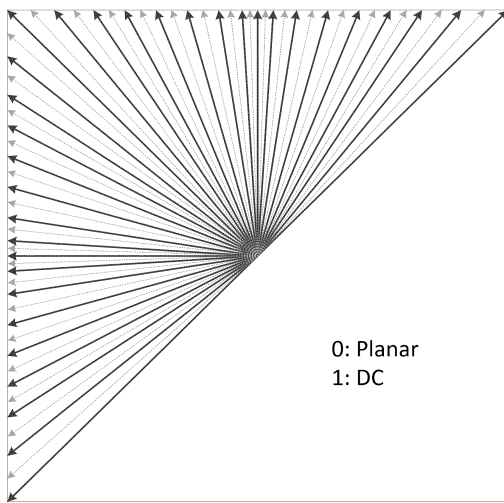
도면12



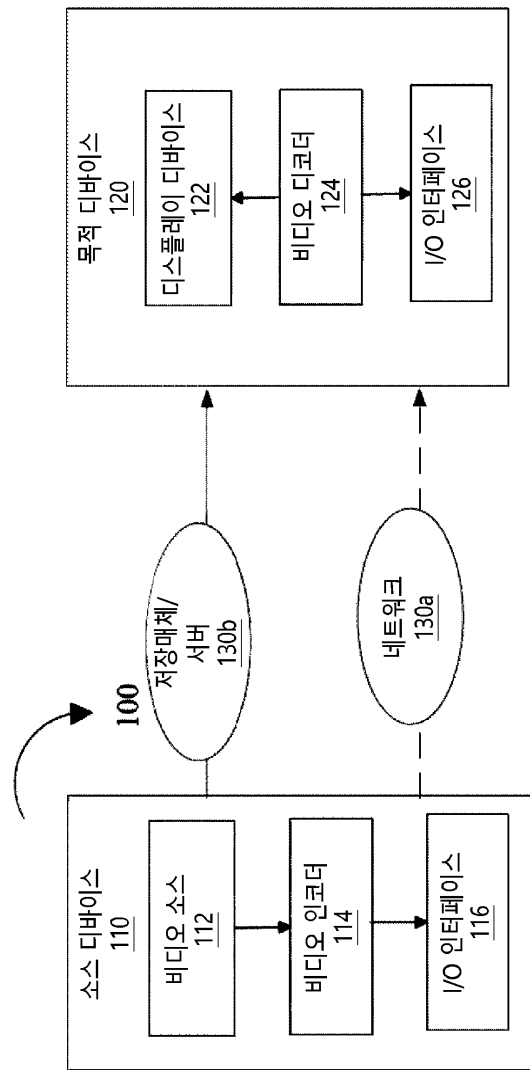
도면13



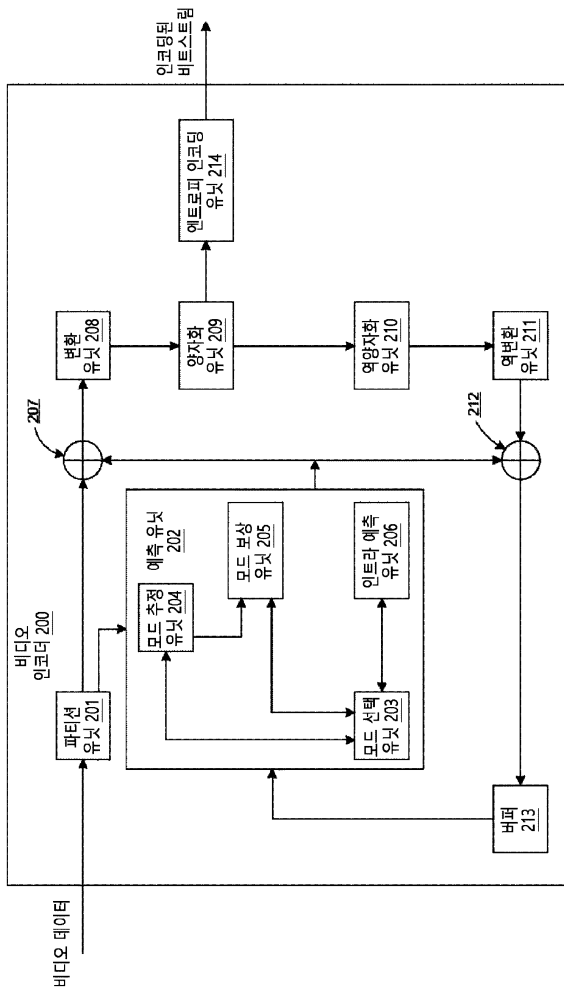
도면14



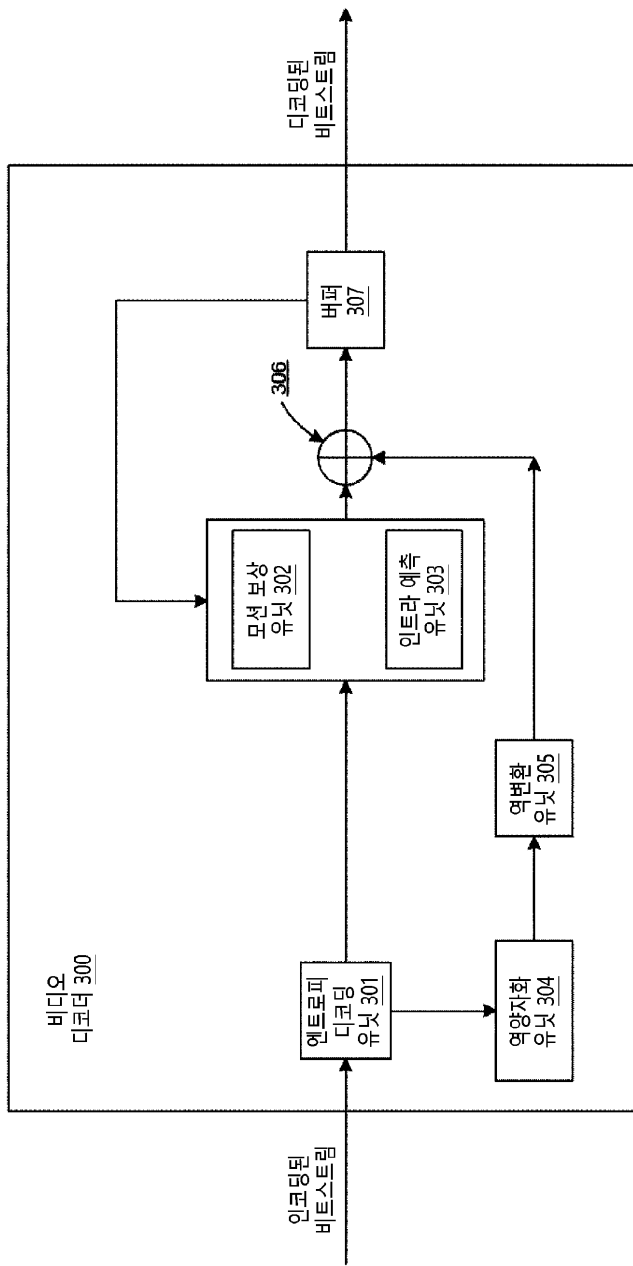
도면15



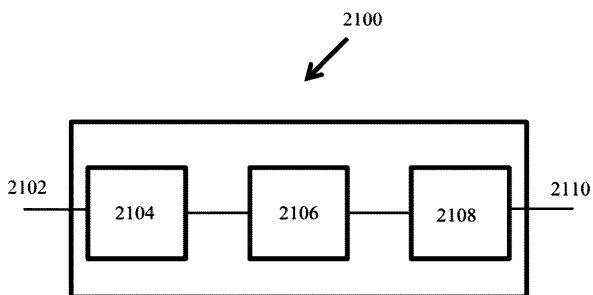
도면16



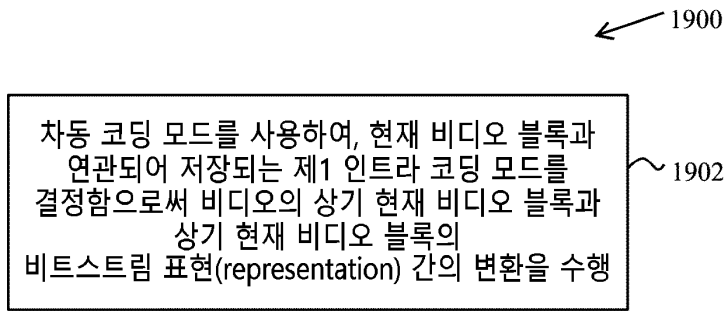
도면17



도면18



도면19



도면20

