



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2024년12월19일

(11) 등록번호 10-2744182

(24) 등록일자 2024년12월13일

- (51) 국제특허분류(Int. Cl.)
H04N 19/11 (2014.01) **H04N 19/132** (2014.01)
H04N 19/157 (2014.01) **H04N 19/176** (2014.01)
H04N 19/18 (2014.01) **H04N 19/593** (2014.01)
H04N 19/61 (2014.01)
- (52) CPC특허분류
H04N 19/11 (2015.01)
H04N 19/132 (2015.01)
- (21) 출원번호 10-2021-7032310
- (22) 출원일자(국제) 2020년04월16일
 심사청구일자 2023년03월22일
- (85) 번역문제출일자 2021년10월07일
- (65) 공개번호 10-2021-0145757
- (43) 공개일자 2021년12월02일
- (86) 국제출원번호 PCT/CN2020/085050
- (87) 국제공개번호 WO 2020/211807
 국제공개일자 2020년10월22일
- (30) 우선권주장
 PCT/CN2019/082813 2019년04월16일 중국(CN)
- (56) 선행기술조사문헌
 PFAFF (Fraunoffer) J et al: "CE3: Affine linear weihgted intra prediction (CE3-4.1, CE3-4.2)", JVET-N0217, 2019.3.25.
 PFAFF, Jonathan et al. "CE3: Affine linear weighted intra prediction (test 1.2.1, test 1.2.2)", JVET-M0043, 2019.1.18.

- (73) 특허권자
두인 비전 컴퍼니 리미티드
 중국, 베이징 100041, 스징산 디스트릭트, 스징 로드, 넘버 30, 넘버 3빌딩, 2층, 룸 비-0035
바이트댄스 아이엔씨
 미국, 90066 캘리포니아 로스엔젤레스 스위트룸 137호 6층 제퍼슨 블러바드 웨스트 12655
- (72) 발명자
덩, 즈펄
 중국 100080 베이징 하이덴 디스트릭트 즈춘 로드 넘버 63 차이나 세틀라이트 커뮤니케이션즈 타워 진터우타오 포스트 오피스
장, 카이
 미국 90066 캘리포니아주 로스엔젤레스 스위트 넘버 137 6층 웨스트 제퍼슨 블러바드 12655
 (뒷면에 계속)
- (74) 대리인
양영준, 임규빈, 백만기

전체 청구항 수 : 총 15 항

심사관 : 김영태

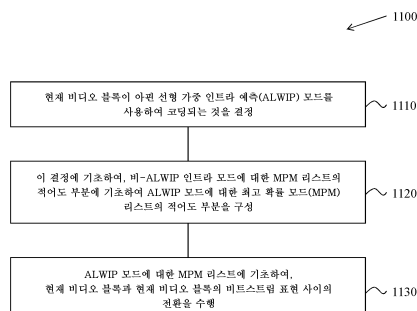
(54) 발명의 명칭 **인트라 코딩 모드에서의 행렬 도출**

(57) 요약

비디오 코딩을 위한 행렬 기반 인트라 예측 방법들을 포함하는 디지털 비디오 코딩을 위한 디바이스들, 시스템들 및 방법들이 설명된다. 대표적인 양태에서, 비디오 처리를 위한 방법은 규칙에 따라 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함하고, 규칙은 전환 동안 행렬 가중

(뒷면에 계속)

대표도 - 도11



인트라 예측(MIP) 모드에서 적용되는 행렬들 또는 오프셋 값들과 현재 비디오 블록의 샘플들 사이의 관계를 특정하고, MIP 모드는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 예측 블록을 결정하는 것을 포함한다.

(52) CPC특허분류

H04N 19/157 (2015.01)

H04N 19/176 (2015.01)

H04N 19/18 (2015.01)

H04N 19/593 (2015.01)

H04N 19/61 (2015.01)

(72) 발명자

장, 리

미국 90066 캘리포니아주 로스엔젤레스 스위트 넘버 137 6층 웨스트 제퍼슨 블러바드 12655

류, 홍빈

중국 100080 베이징 하이덴 디스트릭트 즈춘 로드 넘버 63 차이나 세틀라이트 커뮤니케이션즈 타워 진르터우타오 포스트 오피스

쉬, 지정

미국 90066 캘리포니아주 로스엔젤레스 스위트 넘버 137 6층 웨스트 제퍼슨 블러바드 12655

명세서

청구범위

청구항 1

비디오 데이터를 처리하는 방법으로서,

상기 비디오의 제1 비디오 블록과 상기 비디오의 비트스트림 간의 제1 변환을 위해, 상기 비디오의 상기 제1 비디오 블록에 제1 인트라 모드가 적용된다고 결정하는 단계;

상기 제1 비디오 블록의 참조 샘플들을 도출하는 단계;

상기 제1 비디오 블록에 대한 예측 샘플들을 생성하기 위해, 상기 제1 비디오 블록의 크기에 기초하여 상기 제1 비디오 블록의 상기 참조 샘플들에 대해 경계 다운샘플링 동작을, 이어서 행렬 벡터 곱셈 동작을, 이어서 선택적으로 업샘플링 동작을 수행하는 단계; 및

상기 제1 비디오 블록의 상기 예측 샘플들에 기초하여 상기 제1 변환을 수행하는 단계

를 포함하고,

상기 행렬 벡터 곱셈 동작에서 적용되는 행렬들 또는 오프셋 값들의 요소들은 고정된 정밀도를 갖고,

상기 경계 다운샘플링 동작은 시프트된 비트들의 수가 0보다 큰 것에 응답하여 적어도 하나의 참조 샘플의 합에 대해 우측 비트 시프트 동작을 적용함으로써 규칙에 따라 경계 샘플들을 도출하는 것을 포함하고,

상기 경계 샘플들 $y[x]$ 는 다음 수식식들:

$m > 1$ 인 경우, $y[x] = (\sum_{i=0}^{m-1} p[x * m + i])_{+(1 \ll (\text{Log2}(m) - 1)) \gg \text{Log2}(m)}$, 또는

$m = 1$ 인 경우, $y[x] = p[x]$

중 하나를 사용하여 산출되고,

m 은 상기 제1 비디오 블록 크기 및 경계 크기에 의해 결정되고,

$p[x]$ 는 참조 샘플들 x 의 수를 나타내고,

$\gg$ 는 상기 우측 비트 시프트 동작을 나타내고,

$\ll$ 는 좌측 비트 시프트 동작을 나타내는, 방법.

청구항 2

제1항에 있어서, 상기 제1 비디오 블록 및 상기 참조 샘플들의 정밀도는 10-비트 심도인, 방법.

청구항 3

제1항에 있어서, 상이한 크기들을 갖는 적어도 2개의 블록은 상기 제1 인트라 모드에서 동일한 크기를 갖는 행렬을 공유하는, 방법.

청구항 4

제1항에 있어서, 상기 참조 샘플들은 참조 샘플 필터링 동작을 호출하지 않고 도출되는, 방법.

청구항 5

제1항에 있어서, 상기 참조 샘플 필터링 동작은 상기 제1 인트라 모드와 상이한 정상 인트라 모드에서 적용되는, 방법.

청구항 6

제1항에 있어서, 상기 제1 인트라 모드가 적용되는지는 시퀀스 레벨에서 제시되는 제1 선택스 요소 및 코딩 단위 레벨에서 제시되는 제2 선택스 요소에 의해 지정되는, 방법.

청구항 7

제6항에 있어서, 상기 제2 선택스 요소의 적어도 하나의 빈은 컨텍스트-기반 코딩되는, 방법.

청구항 8

제7항에 있어서, 상기 제1 비디오 블록의 폭-높이 비율이 2보다 큰 것에 응답하여, 상기 제2 선택스 요소의 제1 빈에 대해 3의 인덱스를 갖는 컨텍스트가 사용되는, 방법.

청구항 9

제7항에 있어서, 상기 제1 비디오 블록의 폭-높이 비율이 2 이하인 것에 응답하여, 0, 1 또는 2의 인덱스들을 갖는 컨텍스트들로부터 선택된 단일 컨텍스트가 상기 제2 선택스 요소의 제1 빈에 사용되는, 방법.

청구항 10

제1항에 있어서, 상기 전환은 현재 비디오 블록을 상기 비트스트림으로 인코딩하는 것을 포함하는, 방법.

청구항 11

제1항에 있어서, 상기 전환은 상기 비트스트림으로부터 현재 비디오 블록을 디코딩하는 것을 포함하는, 방법.

청구항 12

프로세서 및 명령어들을 갖는 비일시적 메모리를 포함하는 비디오 데이터 처리를 위한 장치로서, 상기 명령어들은 상기 프로세서에 의해 실행될 때 상기 프로세서로 하여금:

상기 비디오의 제1 비디오 블록과 상기 비디오의 비트스트림 간의 제1 전환을 위해, 상기 비디오의 상기 제1 비디오 블록에 제1 인트라 모드가 적용된다고 결정하고;

상기 제1 비디오 블록의 참조 샘플들을 도출하고;

상기 제1 비디오 블록에 대한 예측 샘플들을 생성하기 위해, 상기 제1 비디오 블록의 크기에 기초하여 상기 제1 비디오 블록의 상기 참조 샘플들에 대해 경계 다운샘플링 동작을, 이어서 행렬 벡터 곱셈 동작을, 이어서 선택적으로 업샘플링 동작을 수행하고;

상기 제1 비디오 블록의 상기 예측 샘플들에 기초하여 상기 제1 전환을 수행하게 하며,

상기 행렬 벡터 곱셈 동작에서 적용되는 행렬들 또는 오프셋 값들의 요소들은 고정된 정밀도를 갖고,

상기 경계 다운샘플링 동작은 시프트된 비트들의 수가 0보다 큰 것에 응답하여 적어도 하나의 참조 샘플의 합에 대해 우측 비트 시프트 동작을 적용함으로써 규칙에 따라 경계 샘플들을 도출하는 것을 포함하고,

상기 경계 샘플들 $y[x]$ 는 다음 수학식들:

$$m > 1 \text{인 경우, } y[x] = \left(\sum_{i=0}^{m-1} p[x * m + i] \right)_{+(1 \ll (\text{Log2}(m) - 1)) \gg \text{Log2}(m)}, \text{ 또는}$$

$$m = 1 \text{인 경우, } y[x] = p[x]$$

중 하나를 사용하여 산출되고,

m 은 상기 제1 비디오 블록 크기 및 경계 크기에 의해 결정되고,

$p[x]$ 는 참조 샘플들 x 의 수를 나타내고,

$\gg$ 는 상기 우측 비트 시프트 동작을 나타내고,

$\ll$ 는 좌측 비트 시프트 동작을 나타내는, 장치.

청구항 13

명령어들을 저장하는 비밀시적 컴퓨터 판독가능 저장 매체로서, 상기 명령어들은 프로세서로 하여금:

비디오의 제1 비디오 블록과 상기 비디오의 비트스트림 간의 제1 전환을 위해, 상기 비디오의 상기 제1 비디오 블록에 제1 인트라 모드가 적용된다고 결정하고;

상기 제1 비디오 블록의 참조 샘플들을 도출하고;

상기 제1 비디오 블록에 대한 예측 샘플들을 생성하기 위해, 상기 제1 비디오 블록의 크기에 기초하여 상기 제1 비디오 블록의 상기 참조 샘플들에 대해 경계 다운샘플링 동작을, 이어서 행렬 벡터 곱셈 동작을, 이어서 선택적으로 업샘플링 동작을 수행하고;

상기 제1 비디오 블록의 상기 예측 샘플들에 기초하여 상기 제1 전환을 수행하게 하며,

상기 행렬 벡터 곱셈 동작에서 적용되는 행렬들 또는 오프셋 값들의 요소들은 고정된 정밀도를 갖고,

상기 경계 다운샘플링 동작은 시프트된 비트들의 수가 0보다 큰 것에 응답하여 적어도 하나의 참조 샘플의 합에 대해 우측 비트 시프트 동작을 적용함으로써 규칙에 따라 경계 샘플들을 도출하는 것을 포함하고,

상기 경계 샘플들 $y[x]$ 는 다음 수학식들:

$m > 1$ 인 경우, $y[x] = (\sum_{i=0}^{m-1} p[x * m + i])_{+(1 \ll (\log_2(m) - 1)) \gg \log_2(m)}$, 또는

$m = 1$ 인 경우, $y[x] = p[x]$

중 하나를 사용하여 산출되고,

m 은 상기 제1 비디오 블록 크기 및 경계 크기에 의해 결정되고,

$p[x]$ 는 참조 샘플들 x 의 수를 나타내고,

$\gg$ 는 상기 우측 비트 시프트 동작을 나타내고,

$\ll$ 는 좌측 비트 시프트 동작을 나타내는, 비밀시적 컴퓨터 판독가능 저장 매체.

청구항 14

비디오 처리 장치에 의해 수행되는 방법에 의해 생성되는 비디오의 비트스트림을 저장하는 비밀시적 컴퓨터 판독가능 기록 매체로서, 상기 방법은:

상기 비디오의 제1 비디오 블록에 제1 인트라 모드가 적용된다고 결정하는 단계;

상기 제1 비디오 블록의 참조 샘플들을 도출하는 단계;

상기 제1 비디오 블록에 대한 예측 샘플들을 생성하기 위해, 상기 제1 비디오 블록의 크기에 기초하여 상기 제1 비디오 블록의 상기 참조 샘플들에 대해 경계 다운샘플링 동작을, 이어서 행렬 벡터 곱셈 동작을, 이어서 선택적으로 업샘플링 동작을 수행하는 단계; 및

상기 제1 비디오 블록의 상기 예측 샘플들에 기초하여 상기 비트스트림을 생성하는 단계

를 포함하고,

상기 행렬 벡터 곱셈 연산에서 적용되는 행렬들 또는 오프셋 값들의 요소들은 고정된 정밀도를 갖고,

상기 경계 다운샘플링 동작은 시프트된 비트들의 수가 0보다 큰 것에 응답하여 적어도 하나의 참조 샘플의 합에 대해 우측 비트 시프트 동작을 적용함으로써 규칙에 따라 경계 샘플들을 도출하는 것을 포함하고,

상기 경계 샘플들 $y[x]$ 는 다음 수학식들:

$m > 1$ 인 경우, $y[x] = (\sum_{i=0}^{m-1} p[x * m + i])_{+(1 \ll (\log_2(m) - 1)) \gg \log_2(m)}$, 또는

$m = 1$ 인 경우, $y[x] = p[x]$

중 하나를 사용하여 산출되고,

m 은 상기 제1 비디오 블록 크기 및 경계 크기에 의해 결정되고,
 $p[x]$ 는 참조 샘플들 x 의 수를 나타내고,
 $>>$ 는 상기 우측 비트 시프트 동작을 나타내고,
 $<<$ 는 좌측 비트 시프트 동작을 나타내는, 비일시적 컴퓨터 판독가능 기록 매체.

청구항 15

제1항에 있어서, 상기 고정된 정밀도는 8 비트 정밀도인, 방법.

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

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

발명의 설명

기술 분야

[0001] 관련 출원들에 대한 상호 참조

[0002] 본 출원은 2019년 4월 16일자로 출원된 국제 특허 출원 제PCT/CN2019/082813호에 대한 우선권 및 이익을 주장하는, 2020년 4월 16일자로 출원된 국제 특허 출원 제PCT/CN2020/085050호에 기초한다. 앞서 언급된 모든 특허 출원들은 그 전체가 본 명세서에 참고로 포함된다.

[0003] 기술 분야

[0004] 본 특허 문서는 비디오 코딩 기법들, 디바이스들 및 시스템들에 관한 것이다.

배경 기술

[0005] 비디오 압축의 진보에도 불구하고, 디지털 비디오는 인터넷 및 다른 디지털 통신 네트워크들에서 가장 큰 대역폭 사용을 여전히 차지한다. 비디오를 수신 및 디스플레이할 수 있는 연결된 사용자 디바이스들의 수가 증가함에 따라, 디지털 비디오 사용에 대한 대역폭 수요가 계속 증가할 것으로 예상된다.

발명의 내용

- [0006] 디지털 비디오 코딩에 관련된 디바이스들, 시스템들 및 방법들, 및 구체적으로, 비디오 코딩을 위한 행렬 기반 인트라 예측 방법들이 설명된다. 설명된 방법들은 기존의 비디오 코딩 표준들(예를 들어, 고효율 비디오 코딩(HEVC)) 및 장래의 비디오 코딩 표준들(예를 들어, 다용도 비디오 코딩(VVC)) 또는 코덱들 둘 다에 적용될 수 있다.
- [0007] 비디오 처리의 제1 예시적인 방법은 규칙에 따라 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함하고, 규칙은 전환 동안 행렬 가중 인트라 예측(MIP) 모드에서 적용되는 행렬들 또는 오프셋 값들과 현재 비디오 블록의 샘플들 사이의 관계를 특정하고, MIP 모드는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 예측 블록을 결정하는 것을 포함한다.
- [0008] 비디오 처리의 제2 예시적인 방법은, 현재 비디오 블록에 대해, 행렬 가중 인트라 예측(MIP) 모드를 사용하여 중간 예측 블록을 생성하는 단계- MIP 모드에서는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 중간 예측 블록이 결정됨 -; 중간 예측 블록에 기초하여, 추가 동작에 기초하여 최종 예측 블록을 생성하는 단계; 및 최종 예측 신호에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0009] 비디오 처리의 제3 예시적인 방법은, 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계- 전환은 행렬 가중 인트라 예측(MIP) 모드에서 현재 비디오 블록의 적어도 부분의 복수의 샘플을 예측하는 것을 포함하는데, MIP 모드에서는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 부분의 예측 블록이 결정됨 -를 포함한다.
- [0010] 비디오 처리의 제4 예시적인 방법은, 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계- 전환은 전환 동안 행렬 가중 인트라 예측(MIP) 모드를 적용하기 전에 현재 비디오 블록의 이웃 샘플들을 필터링할지를 표시하는 규칙에 기초하고, MIP 모드는, 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 예측 블록을 결정하는 것을 포함함 -를 포함한다.
- [0011] 또 다른 대표적인 양태에서, 개시된 기술은 비디오 처리 방법을 제공하기 위해 사용될 수 있다. 이 예시적인 방법은 현재 비디오 블록이 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계, 이 결정에 기초하여, 비-ALWIP 인트라 모드에 대한 MPM 리스트의 적어도 부분에 기초하여 ALWIP 모드에 대한 최고 확률 모드(MPM) 리스트의 적어도 부분을 구성하는 단계, 및 ALWIP 모드에 대한 MPM 리스트에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0012] 또 다른 대표적인 양태에서, 개시된 기술은 비디오 처리 방법을 제공하기 위해 사용될 수 있다. 이 예시적인 방법은 현재 비디오 블록의 루마 컴포넌트가 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계, 이 결정에 기초하여, 크로마 인트라 모드를 추론하는 단계, 및 크로마 인트라 모드에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0013] 또 다른 대표적인 양태에서, 개시된 기술은 비디오 처리 방법을 제공하기 위해 사용될 수 있다. 이 예시적인 방법은 현재 비디오 블록이 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계, 및 결정에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0014] 또 다른 대표적인 양태에서, 개시된 기술은 비디오 처리 방법을 제공하기 위해 사용될 수 있다. 이 예시적인 방법은 현재 비디오 블록이 아핀 선형 가중 인트라 예측(ALWIP) 모드와 상이한 코딩 모드를 사용하여 코딩되는 것을 결정하는 단계, 및 이 결정에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0015] 또 다른 대표적인 양태에서, 개시된 기술은 비디오 처리 방법을 제공하기 위해 사용될 수 있다. 이 예시적인 방법은, 현재 비디오 블록에 대해, 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 제1 예측을 생성하는 단계, 제1 예측에 기초하여, 위치 의존적 인트라 예측 조합(PDPC)을 사용하여 제2 예측을 생성하는 단계, 및 제2 예측에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.

- [0016] 또 다른 대표적인 양태에서, 개시된 기술은 비디오 처리 방법을 제공하기 위해 사용될 수 있다. 이러한 예시적인 방법은 현재 비디오 블록이 아편 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계, ALWIP 모드에 기초하여, 현재 비디오 블록의 복수의 서브-블록을 예측하는 단계, 및 예측에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0017] 또 다른 대표적인 양태에서, 전술한 방법은 프로세서 실행가능 코드의 형태로 구체화되고 컴퓨터 판독가능 프로그램 매체에 저장된다.
- [0018] 또 다른 대표적인 양태에서, 전술한 방법을 수행하도록 구성되거나 동작가능한 디바이스가 개시된다. 디바이스는 이 방법을 구현하도록 프로그래밍되는 프로세서를 포함할 수 있다.
- [0019] 또 다른 대표적인 양태에서, 비디오 디코더 장치는 본 명세서에 설명된 방법을 구현할 수 있다.
- [0020] 개시된 기술의 위의 양태들 및 다른 양태들 및 특징들은 도면들, 설명 및 청구항들에서 더 상세히 설명된다.

도면의 간단한 설명

- [0021] 도 1은 33개의 인트라 예측 방향의 예를 도시한다.
- 도 2는 67개의 인트라 예측 모드의 예를 도시한다.
- 도 3은 선형 모델의 가중치들의 도출에 사용되는 샘플들의 위치들의 예를 도시한다.
- 도 4는 예측 블록에 이웃하는 4개의 참조 라인의 예를 도시한다.
- 도 5a 및 도 5b는 블록 크기에 따른 서브-파티션들의 예를 도시한다.
- 도 6은 4x4 블록들에 대한 ALWIP의 예를 도시한다.
- 도 7은 8x8 블록들에 대한 ALWIP의 예를 도시한다.
- 도 8은 8x4 블록들에 대한 ALWIP의 예를 도시한다.
- 도 9는 16x16 블록들에 대한 ALWIP의 예를 도시한다.
- 도 10은 MPM 리스트 구성에서 사용하는 이웃 블록들의 예를 도시한다.
- 도 11은 개시된 기술에 따른, 행렬 기반 인트라 예측을 위한 예시적인 방법의 흐름도를 도시한다.
- 도 12는 개시된 기술에 따른, 행렬 기반 인트라 예측을 위한 다른 예시적인 방법의 흐름도를 도시한다.
- 도 13은 개시된 기술에 따른, 행렬 기반 인트라 예측을 위한 또 다른 예시적인 방법의 흐름도를 도시한다.
- 도 14는 개시된 기술에 따른, 행렬 기반 인트라 예측을 위한 또 다른 예시적인 방법의 흐름도를 도시한다.
- 도 15는 본 문서에 설명된 비주얼 미디어 디코딩 또는 비주얼 미디어 인코딩 기법을 구현하는 하드웨어 플랫폼의 예의 블록도이다.
- 도 16은 본 명세서에 개시된 다양한 기법들이 구현될 수 있는 예시적인 비디오 처리 시스템을 도시하는 블록도이다.
- 도 17은 본 개시의 기법들을 활용할 수 있는 예시적인 비디오 코딩 시스템을 예시하는 블록도이다.
- 도 18은 비디오 인코더의 예를 예시하는 블록도이다.
- 도 19는 비디오 디코더의 예를 예시하는 블록도이다.
- 도 20 내지 도 23은 개시된 기술에 따른, 행렬 기반 인트라 예측을 위한 추가적인 예시적인 방법들의 예시적인 흐름도들을 도시한다.

발명을 실시하기 위한 구체적인 내용

- [0022] 더 높은 해상도의 비디오의 수요가 증가함으로 인해, 비디오 코딩 방법들 및 기법들은 현대의 기술에서 편재적이다. 비디오 코덱들은 전형적으로 디지털 비디오를 압축 또는 압축 해제하고, 더 높은 코딩 효율을 제공하도록 계속 개선되고 있는 전자 회로 또는 소프트웨어를 포함한다. 비디오 코덱은 압축되지 않은 비디오를 압축된 포맷으로 전환하거나, 그 반대로 전환한다. 비디오 품질, 비디오를 표현하기 위해 사용되는 데이터의 양(비트

레이트에 의해 결정됨), 인코딩 및 디코딩 알고리즘들의 복잡도, 데이터 손실들 및 에러들에 대한 민감도, 편집의 용이성, 랜덤 액세스, 및 엔드-투-엔드 지연(레이턴시) 사이에 복잡한 관계들이 존재한다. 압축된 포맷은 보통 표준 비디오 압축 사양, 예를 들어, HEVC(High Efficiency Video Coding) 표준(H.265 또는 MPEG-H Part 2로도 알려짐), 완성될 다용도 비디오 코딩(Versatile Video Coding)(VVC) 표준, 또는 다른 현재 및/또는 미래의 비디오 코딩 표준들을 따른다.

[0023] 개시된 기술의 실시예들은 기존의 비디오 코딩 표준들(예를 들어, HEVC, H.265) 및 런타임 성능을 개선시키기 위한 미래의 표준들에 적용될 수 있다. 섹션 제목들은 본 문서에서 설명의 가독성을 개선시키기 위해 사용되며, 논의 또는 실시예들(및/또는 구현들)을 결코 각각의 섹션들에만 제한하지 않는다.

[0024] 1 HEVC에 대한 간단한 검토

[0025] 1.1 HEVC/H.265에서의 인트라 예측

[0026] 인트라 예측은 고려된 컬러 채널에서 이전에 재구성된 샘플들을 사용하여 주어진 TB(변환 블록)에 대한 샘플들을 생성하는 것을 수반한다. 인트라 예측 모드는 루마 및 크로마 채널들에 대해 개별적으로 시그널링되며, 크로마 채널 인트라 예측 모드는 'DM_CHROMA' 모드를 통해 루마 채널 인트라 예측 모드에 선택적으로 의존한다. 인트라 예측 모드가 PB(예측 블록) 레벨에서 시그널링되지만, 인트라 예측 프로세스는 CU에 대한 잔차 쿼드-트리 계층구조에 따라 TB 레벨에서 적용되고, 그에 의해 하나의 TB의 코딩이 CU 내의 다음 TB의 코딩에 영향을 미칠 수 있게 하고, 따라서 참조 값들로서 사용되는 샘플들까지의 거리를 감소시킨다.

[0027] HEVC는 35개의 인트라 예측 모드 - DC 모드, 평면(planar) 모드 및 33개의 방향, 또는 '각도' 인트라 예측 모드를 포함한다. 33개의 각도 인트라 예측 모드가 도 1에 예시된다.

[0028] 크로마 컬러 채널들과 관련된 PB들의 경우, 인트라 예측 모드는 평면, DC, 수평, 수직, 'DM_CHROMA' 모드 또는 때때로 대각선 모드 '34'로서 특징된다.

[0029] 크로마 포맷들 4:2:2 및 4:2:0에 대해, 크로마 PB는 2개 또는 4개(각각)의 루마 PB와 중첩할 수 있고; 이 경우에, DM_CHROMA에 대한 루마 방향은 이 루마 PB들의 상단 좌측으로부터 취해진다.

[0030] DM_CHROMA 모드는 루마 컬러 채널 PB의 인트라 예측 모드가 크로마 컬러 채널 PB들에 적용되는 것을 표시한다. 이것은 비교적 일반적이기 때문에, intra_chroma_pred_mode의 최고 확률 모드 코딩 스킴은 선택되고 있는 이 모드를 위해 바이어싱된다.

[0031] 2 VVC에서의 인트라 예측의 예들

[0032] 2.1 67개의 인트라 예측 모드에 의한 인트라 모드 코딩

[0033] 내추럴 비디오(natural video)에서 제시된 임의의 에지 방향들을 캡처하기 위해, 방향 인트라 모드(directional intra mode)들의 수는, HEVC에서 사용되는 바와 같이, 33에서 65로 확장된다. 추가적인 방향 모드들은 도 2에서 적색 점선 화살표들로서 도시되고, 평면 및 DC 모드들은 동일하게 유지된다. 이러한 더 조밀한 방향 인트라 예측 모드들은 모든 블록 크기들에 대해 그리고 루마 및 크로마 인트라 예측들 둘 다에 대해 적용된다.

[0034] 2.2 크로스 컴포넌트 선형 모델(CCLM)의 예들

[0035] 일부 실시예들에서, 그리고 크로스 컴포넌트 리던던시를 감소시키기 위해, 크로스 컴포넌트 선형 모델(CCLM) 예측 모드(LM이라고도 함)가 JEM에서 사용되고, 그에 대해 크로마 샘플들은 다음과 같이 선형 모델을 사용함으로써 동일한 CU의 재구성된 루마 샘플들에 기초하여 예측된다:

[0036]
$$pred_c(i, j) = \alpha \cdot rec_L'(i, j) + \beta \quad (1)$$

[0037] 여기서, $pred_c(i, j)$ 는 CU 내의 예측된 크로마 샘플들을 나타내고, $rec_L'(i, j)$ 는 동일한 CU의 다운샘플링된 재구성된 루마 샘플들을 나타낸다. 선형 모델 파라미터 α 및 β 는, 다운샘플링된 이웃 루마 샘플들의 세트 내의 최소 샘플 값 및 최대 샘플을 갖는 루마 샘플, 및 그들의 대응하는 크로마 샘플들인, 2개의 샘플로부터의 루마 값들과 크로마 값들 사이의 관계로부터 도출된다. 도 3은 CCLM 모드에 관련된 좌측 및 상부 샘플들의 위치 및 현재 블록의 샘플의 예를 도시한다.

[0038] 이 파라미터 계산은 디코딩 프로세스의 일부로서 수행되며, 인코더 탐색 동작으로서가 아니다. 그 결과, 디코더에 α 및 β 값들을 전달하기 위해 신덱스가 사용되지 않는다.

- [0039] 크로마 인트라 모드 코딩의 경우, 크로마 인트라 모드 코딩을 위해 총 8개의 인트라 모드가 허용된다. 그러한 모드들은 5개의 전통적인 인트라 모드 및 3개의 크로스 컴포넌트 선형 모델 모드(CCLM, LM_A 및 LM_L)를 포함한다. 크로마 모드 코딩은 대응하는 루마 블록의 인트라 예측 모드에 직접 의존한다. 루마 및 크로마 컴포넌트들에 대한 개별 블록 파티셔닝 구조가 I개의 슬라이스에서 인에이블되기 때문에, 하나의 크로마 블록은 다수의 루마 블록에 대응할 수 있다. 따라서, 크로마 DM 모드의 경우, 현재 크로마 블록의 중심 위치를 커버하는 대응하는 루마 블록의 인트라 예측 모드가 직접 계승된다.
- [0040] 2.3 다중 참조 라인(MRL) 인트라 예측
- [0041] 다중 참조 라인(MRL) 인트라 예측은 인트라 예측을 위해 더 많은 참조 라인들을 사용한다. 도 4에서, 4개의 참조 라인의 예가 도시되어 있으며, 여기서 세그먼트 A 및 F의 샘플들은 재구성된 이웃 샘플로부터 폐치되지 않지만, 각각 세그먼트 B 및 E로부터의 가장 가까운 샘플들로 패딩된다. HEVC 인트라-픽처 예측은 가장 가까운 참조 라인(즉, 참조 라인 0)을 사용한다. MRL에서, 2개의 추가 라인(참조 라인 1 및 참조 라인 3)이 사용된다. 선택된 참조 라인의 인덱스(mrl_idx)가 시그널링되고 인트라 예측기를 생성하기 위해 사용된다. 0보다 큰 참조 라인 idx에 대해, MPM 리스트 내의 추가적인 참조 라인 모드들만을 포함하고, 잔여 모드 없이 mpm 인덱스만을 시그널링한다.
- [0042] 2.4 인트라 서브-파티션들(ISP)
- [0043] ISP(Intra Sub-Partitions) 톨은 블록 크기에 의존하여 루마 인트라-예측 블록들을 수직으로 또는 수평으로 2 또는 4개의 서브-파티션으로 분할한다. 예를 들어, ISP에 대한 최소 블록 크기는 4x8(또는 8x4)이다. 블록 크기가 4x8(또는 8x4)보다 큰 경우에, 대응하는 블록은 4개의 서브-파티션으로 분할된다. 도 5는 2개의 가능성의 예들을 도시한다. 모든 서브-파티션들은 적어도 16개의 샘플을 갖는 조건을 충족시킨다.
- [0044] 각각의 서브-파티션에 대해, 잔차 신호를 예측 신호에 가산함으로써 재구성된 샘플들이 획득된다. 여기서, 잔차 신호는 엔트로피 디코딩, 역양자화 및 역변환과 같은 프로세스들에 의해 생성된다. 따라서, 각각의 서브-파티션의 재구성된 샘플 값들은 다음 서브-파티션의 예측을 생성하기 위해 이용가능하고, 각각의 서브-파티션은 반복적으로 처리된다. 또한, 처리될 제1 서브-파티션은 CU의 상단 좌측 샘플을 포함하고 그 후 하향으로(수평 분할) 또는 우향으로(수직 분할) 계속 가는 것이다. 그 결과, 서브-파티션 예측 신호들을 생성하기 위해 사용되는 참조 샘플들은 라인들의 좌측 및 상부 측에만 위치한다. 모든 서브-파티션들은 동일한 인트라 모드를 공유한다.
- [0045] 2.5 아핀 선형 가중 인트라 예측(ALWIP 또는 행렬 기반 인트라 예측)
- [0046] 아핀 선형 가중 인트라 예측(ALWIP, 행렬 기반 인트라 예측(MIP)이라고도 알려짐)은 JVET-N0217에서 제안된다.
- [0047] JVET-N0217에서, 2개의 테스트가 수행된다. 테스트 1에서, ALWIP는 8K 바이트의 메모리 제한 및 샘플당 최대 4개의 곱셈으로 설계된다. 테스트 2는 테스트 1과 유사하지만, 메모리 요건 및 모델 아키텍처의 관점에서 설계를 더 단순화한다.
- [0048] ○ 모든 블록 형상들에 대한 행렬들 및 오프셋 벡터들의 단일 세트.
- [0049] ○ 모든 블록 형상들에 대해 모드들의 수의 19로의 감소.
- [0050] ○ 5760 10비트 값, 즉 7.20 킬로바이트로의 메모리 요건의 감소.
- [0051] ○ 예측된 샘플들의 선형 보간은 제1 테스트에서와 같이 반복적 보간을 대체하는 방향당 단일 단계로 수행된다.
- [0052] 2.5.1 JVET-N0217의 테스트 1
- [0053] 폭 W 및 높이 H의 직사각형 블록의 샘플들을 예측하기 위해, 아핀 선형 가중 인트라 예측(ALWIP)은 블록의 좌측에 있는 H개의 재구성된 이웃 경계 샘플의 하나의 라인 및 블록 상부의 W개의 재구성된 이웃 경계 샘플의 하나의 라인을 입력으로서 취한다. 재구성된 샘플들이 이용가능하지 않은 경우, 이들은 종래의 인트라 예측에서 행해지는 바와 같이 생성된다.
- [0054] 예측 신호의 생성은 다음의 3개의 단계에 기초한다:
- [0055] 경계 샘플들 중에서, W=H=4의 경우에 4개의 샘플 및 모든 다른 경우에 8개의 샘플이 평균화에 의해 추출된다.
- [0056] 오프셋의 가산이 뒤따르는 행렬 벡터 곱셈은 입력으로서 평균화된 샘플들을 이용하여 수행된다. 그 결과는 원

래의 블록에서의 샘플들의 서브샘플링된 세트에 대한 감소된 예측 신호이다.

- [0057] 나머지 위치들에서의 예측 신호는 각각의 방향에서의 단일 단계 선형 보간인 선형 보간에 의해 서브샘플링된 세트 상의 예측 신호로부터 생성된다.
- [0058] 예측 신호를 생성하기 위해 필요한 행렬들 및 오프셋 벡터들은 행렬들의 3개의 세트 S_0 , S_1 , S_2 로부터 취해진다. 세트 S_0 은 각각이 16개의 행 및 4개의 열을 갖는 18개의 행렬 A_0^i , $i \in \{0, \dots, 17\}$ 및 각각이 크기 16인 18개의 오프셋 벡터 b_0^i , $i \in \{0, \dots, 17\}$ 로 이루어져 있다. 그 세트의 행렬들 및 오프셋 벡터들은 크기 4x4의 블록들에 사용된다. 세트 S_1 은 각각이 16개의 행 및 8개의 열을 갖는 10개의 행렬 A_1^i , $i \in \{0, \dots, 9\}$ 및 각각이 크기 16인 10개의 오프셋 벡터 b_1^i , $i \in \{0, \dots, 9\}$ 로 이루어져 있다. 그 세트의 행렬들 및 오프셋 벡터들은 크기 4x8, 8x4 및 8x8의 블록들에 사용된다. 마지막으로, 세트 S_2 는 각각이 64개의 행 및 8개의 열을 갖는 6개의 행렬 A_2^i , $i \in \{0, \dots, 5\}$ 및 각각이 크기 64인 6개의 오프셋 벡터 b_2^i , $i \in \{0, \dots, 5\}$ 로 이루어져 있다. 그 세트의 행렬들 및 오프셋 벡터들 또는 이러한 행렬들 및 오프셋 벡터들의 일부는 모든 다른 블록-형상들에 사용된다.
- [0059] 행렬 벡터 곱의 계산에 필요한 곱셈들의 총 수는 항상 $4xWxH$ 이하이다. 즉, ALWIP 모드들에 대해 샘플당 최대 4개의 곱셈이 요구된다.
- [0060] 2.5.2 경계의 평균화
- [0061] 제1 단계에서, 입력 경계들 $bdry_{red}^{top}$ 및 $bdry_{red}^{left}$ 는 더 작은 경계들 $bdry_{red}^{top}$ 및 $bdry_{red}^{left}$ 로 감소된다. 여기서, $bdry_{red}^{top}$ 및 $bdry_{red}^{left}$ 는 둘 다 4x4-블록의 경우에 2개의 샘플로 구성되고, 다른 모든 경우에 둘 다 4개의 샘플로 구성된다.
- [0062] 4x4-블록의 경우, $0 \leq i < 2$ 인 경우,
- $$bdry_{red}^{top}[i] = \left(\left(\sum_{j=0}^1 bdry_{red}^{top}[i \cdot 2 + j] \right) + 1 \right) \gg 1$$
- [0063]
- [0064] 를 정의하고 유사하게 $bdry_{red}^{left}$ 를 정의한다.
- [0065] 그렇지 않고, 블록-폭 W 가 $W=4 \cdot 2^k$ 로서 주어지면, $0 \leq i < 4$ 인 경우,
- $$bdry_{red}^{top}[i] = \left(\left(\sum_{j=0}^{2^k-1} bdry_{red}^{top}[i \cdot 2^k + j] \right) + (1 \ll (k-1)) \right) \gg k$$
- [0066]
- [0067] 를 정의하고 유사하게 $bdry_{red}^{left}$ 를 정의한다.
- [0068] 2개의 감소된 경계들 $bdry_{red}^{top}$ 및 $bdry_{red}^{left}$ 는 감소된 경계 벡터 $bdry_{red}$ 에 연결되고, 따라서 이는 형상 4x4의 블록들에 대해 크기 4이고 모든 다른 형상들의 블록들에 대해 크기 8이다. $mode$ 가 ALWIP-모드를 지칭하는 경우, 이 연결은 다음과 같이 정의된다:

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

마지막으로, 서브샘플링된 예측 신호의 보간을 위해, 큰 블록들 상에서 평균화된 경계의 제2 버전이 필요하다. 즉, $\min(W, H) > 8$ 이고 $W \geq H$ 인 경우, $W = 8 * 2^l$ 을 기입하고, $0 \leq l < 8$ 인 경우,

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

을 정의한다.

$\min(W, H) > 8$ 이고 $H > W$ 인 경우, $bdry_{redll}^{left}$ 를 유사하게 정의한다.

2.5.3 행렬 벡터 곱셈에 의한 감소된 예측 신호의 생성

감소된 입력 벡터 $bdry_{red}$ 중에서, 하나는 감소된 예측 신호 $pred_{red}$ 를 생성한다. 후자의 신호는 폭 W_{red} 및 높이 H_{red} 의 다운샘플링된 블록 상의 신호이다. 여기서, W_{red} 및 H_{red} 는 다음과 같이 정의된다:

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

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

감소된 예측 신호 $pred_{red}$ 는 행렬 벡터 곱을 계산하고 오프셋을 가산함으로써 계산된다:

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

여기서, A는 $W_{red} \cdot H_{red}$ 개의 행 및 $W=H=4$ 인 경우 4개의 열 및 모든 다른 경우들에서 8개의 열을 갖는 행렬이다. b는 $W_{red} \cdot H_{red}$ 크기의 벡터이다.

행렬 A 및 벡터 b는 다음과 같이 세트들 S_0, S_1, S_2 중 하나로부터 취해진다. 인덱스 $idx = idx(W, H)$ 를 다음과 같이 정의한다:

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

또한, m을 다음과 같이 표현한다:

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

이어서, $idx \leq 1$ 이거나 $idx=2$ 이고 $\min(W, H) > 4$ 인 경우, $A = A_{idx}^m$ 이고 $b = b_{idx}^m$ 로 표현한다. $idx=2$ 이고 $\min(W, H)=4$ 인 경우, A를, $W=4$ 인 경우, 다운샘플링된 블록 내의 홀수 x 좌표에 대응하거나, $H=4$ 인 경우, 다운샘플링된 블록 내의 홀수 y 좌표에 대응하는 A_{idx}^m 의 모든 행을 무시함으로써 발생하는 행렬로 한다.

- [0085] 마지막으로, 감소된 예측 신호는 다음의 경우들에서 그의 전치로 대체된다:
- [0086] ○ $W=H=4$ 이고 $mode \geq 18$
- [0087] ○ $\max(W, H)=8$ 이고 $mode \geq 10$
- [0088] ○ $\max(W, H)>8$ 이고 $mode \geq 6$
- [0089] $pred_{red}$ 의 계산에 필요한 곱셈의 수는 $W=H=4$ 의 경우에 4인데, 그 이유는 이 경우에 A가 4개의 열 및 16개의 행을 갖기 때문이다. 모든 다른 경우들에서, A는 8개의 열 및 $W_{red} \cdot H_{red}$ 행들을 가지며, 이들 경우에 $8 \cdot W_{red} \cdot H_{red} \leq 4 \cdot W \cdot H$ 개의 곱셈이 요구되는 것, 즉, 또한 이들 경우에, $pred_{red}$ 를 계산하기 위해 샘플당 최대 4개의 곱셈이 필요한 것을 즉시 검증한다.
- [0090] 2.5.4 전체 ALWIP 프로세스의 예시
- [0091] 평균화, 행렬 벡터 곱셈 및 선형 보간의 전체 프로세스가 도 6 내지 도 9에서 상이한 형상들에 대해 도시된다. 나머지 형상들은 도시된 경우들 중 하나에서와 같이 취급된다는 점에 유의한다.
- [0092] 1. 4×4 블록이 주어지면, ALWIP는 경계의 각각의 축을 따라 2개의 평균을 취한다. 결과적인 4개의 입력 샘플은 행렬 벡터 곱셈에 입력된다. 행렬들은 세트 S_0 로부터 취해진다. 오프셋을 가산한 후, 이것은 16개의 최종 예측 샘플을 산출한다. 선형 보간은 예측 신호를 생성하기 위해 필수적이지 않다. 따라서, 샘플당 총 $(4 \cdot 16)/(4 \cdot 4)=4$ 개의 곱셈이 수행된다.
- [0093] 2. 8×8 블록이 주어지면, ALWIP는 경계의 각각의 축을 따라 4개의 평균을 취한다. 결과적인 8개의 입력 샘플은 행렬 벡터 곱셈에 입력된다. 행렬들은 세트 S_1 로부터 취해진다. 이것은 예측 블록의 홀수 위치들에 대해 16개의 샘플을 산출한다. 따라서, 샘플당 총 $(8 \cdot 16)/(8 \cdot 8)=2$ 개의 곱셈이 수행된다. 오프셋을 가산한 후, 이러한 샘플들은 감소된 상단 경계를 사용하여 수직으로 보간된다. 원래의 좌측 경계를 사용하여 수평 보간이 뒤따른다.
- [0094] 3. 8×4 블록이 주어지면, ALWIP는 경계의 수평 축을 따라 4개의 평균을 취하고 좌측 경계 상의 4개의 원래의 경계값을 취한다. 결과적인 8개의 입력 샘플은 행렬 벡터 곱셈에 입력된다. 행렬들은 세트 S_1 로부터 취해진다. 이것은 예측 블록의 홀수 수평 및 각각의 수직 위치들 상에 16개의 샘플을 산출한다. 따라서, 샘플당 총 $(8 \cdot 16)/(8 \cdot 4)=4$ 개의 곱셈이 수행된다. 오프셋을 가산한 후에, 이들 샘플은 원래의 좌측 경계를 사용하여 수평으로 보간된다.
- [0095] 4. 16×16 블록이 주어지면, ALWIP는 경계의 각각의 축을 따라 4개의 평균을 취한다. 결과적인 8개의 입력 샘플은 행렬 벡터 곱셈에 입력된다. 행렬들은 세트 S_2 로부터 취해진다. 이것은 예측 블록의 홀수 위치들에 대해 64개의 샘플을 산출한다. 따라서, 샘플당 총 $(8 \cdot 64)/(16 \cdot 16)=2$ 개의 곱셈이 수행된다. 오프셋을 가산한 후에, 이들 샘플은 상단 경계의 8개의 평균을 사용하여 수직으로 보간된다. 원래의 좌측 경계를 사용하여 수평 보간이 뒤따른다. 이 경우, 보간 프로세스는 어떠한 곱셈도 추가하지 않는다. 따라서, 총괄적으로, ALWIP 예측을 계산하기 위해 샘플당 2개의 곱셈이 요구된다.
- [0096] 더 큰 형상들에 대해, 절차는 본질적으로 동일하고, 샘플당 곱셈들의 수가 4 미만임을 검사하는 것이 용이하다.
- [0097] $W>8$ 인 $W \times 8$ 블록들의 경우, 샘플들이 홀수 수평 및 각각의 수직 위치들에 주어지기 때문에 수평 보간만이 필요하다.
- [0098] 마지막으로, $W>8$ 인 $W \times 4$ 블록들의 경우, A_{kbe} 를 다운샘플링된 블록의 수평 축을 따라 홀수 엔트리에 대응하는 모든 행을 무시함으로써 발생하는 행렬이라고 한다. 따라서, 출력 크기는 32이고, 다시, 수행될 수평 보간만이 남아 있다.
- [0099] 전치된 경우들은 그에 따라 처리된다.
- [0100] 2.5.5 단일 단계 선형 보간
- [0101] $\max(W, H) \geq 8$ 인 $W \times H$ 블록의 경우, 예측 신호는 선형 보간에 의해 $W_{red} \times H_{red}$ 에서의 감소된 예측 신호 $pred_{red}$ 로부터 발생한다. 블록 형상에 따라, 선형 보간은 수직, 수평 또는 둘 다의 방향으로 행해진다. 선형 보간이 둘 다의 방향으로 적용되어야 하는 경우, 선형 보간은 $W < H$ 인 경우 수평 방향으로 먼저 적용되고, 그 밖의 경우, 선

형 보간은 수직 방향으로 먼저 적용된다.

[0102] 일반성을 잃지 않고, $\max(W, H) \geq 8$ 이고 $W \geq H$ 인 $W \times H$ 블록을 고려한다. 이어서, 1차원 선형 보간이 다음과 같이 수행된다. 일반성을 잃지 않고, 수직 방향에서의 선형 보간을 기술하면 충분하다. 먼저, 감소된 예측 신호는 경계 신호에 의해 상단으로 확장된다. 수직 업샘플링 인자 $U_{ver} = H/H_{red}$ 를 정의하고 $U_{ver} = 2^{u_{ver}} > 1$ 을 기입한다. 그 후, 다음에 의해 확장된 감소된 예측 신호를 정의한다.

$$pred_{red}[x][-1] = \begin{cases} bdry_{red}^{top}[x] & W = 8 \text{인 경우} \\ bdry_{red}^{top}[x] & W > 8 \text{인 경우} \end{cases}.$$

[0104] 그 후, 이 확장된 감소된 예측 신호로부터, 수직 선형 보간된 예측 신호가 다음에 의해 생성된다.

$$pred_{red}^{ups,ver}[x][U_{ver} \cdot y + k] = \left((U_{ver} - k - 1) \cdot pred_{red}[x][y - 1] + (k + 1) \cdot pred_{red}[x][y] + \frac{U_{ver}}{2} \right) \gg u_{ver}$$

[0106] $0 \leq x < W_{red}$, $0 \leq y < H_{red}$ 및 $0 \leq k < U_{ver}$ 의 경우.

[0107] 2.5.6 제안된 인트라 예측 모드들의 신호화

[0108] 인트라 모드에서의 각각의 코딩 유닛(CU)에 대해, ALWIP 모드가 대응하는 예측 유닛(PU)에 적용될지 여부를 표시하는 플래그가 비트스트림에서 전송된다. 후자의 인덱스의 신호화는 JVET-M0043에서와 동일한 방식으로 MRL과 조화된다. ALWIP 모드가 적용되는 경우, ALWIP 모드의 인덱스 $predmode$ 는 3 MPMS를 갖는 MPM-리스트를 사용하여 시그널링된다.

[0109] 여기서, MPM들의 도출은 다음과 같이 상부 및 좌측 PU의 인트라 모드들을 사용하여 수행된다. ALWIP 모드를 각각의 종래의 인트라 예측 모드 $predmode_{Angular}$ 에 할당하는 3개의 고정된 테이블 $map_angular_to_alwip_{idx}$, $idx \in \{0,1,2\}$ 가 존재하며,

$$predmode_{ALWIP} = map_angular_to_alwip_{idx}[predmode_{Angular}] \text{ 이다.}$$

[0111] 폭 W 및 높이 H 의 각각의 PU에 대해, 인덱스

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

[0113] 를 정의하고, 이는 섹션 2.5.3에서와 같이 ALWIP-파라미터들이 3세트 중 어느 것으로부터 취해져야 하는지를 표시한다.

[0114] 상부 예측 유닛 PU_{above} 이 이용가능하고, 현재 PU와 동일한 CTU에 속하며, 인트라 모드에 있는 경우, $idx(PU) = idx(PU_{above})$ 인 경우, 및 ALWIP가 ALWIP-모드 $predmode_{ALWIP}^{above}$ 로 PU_{above} 에 적용되는 경우,

$$mode_{ALWIP}^{above} = predmode_{ALWIP}^{above} \text{로 표현한다.}$$

[0116] 상부 PU가 이용가능하고, 현재 PU와 동일한 CTU에 속하고 인트라 모드에 있는 경우, 그리고 종래의 인트라 예측 모드 $predmode_{Angular}^{above}$ 가 상부 PU에 적용되는 경우,

$$mode_{ALWIP}^{above} = map_angular_to_alwip_{idx(PU_{above})}[predmode_{Angular}^{above}] \text{로 표현한다.}$$

[0118] 모든 다른 경우들에서,

$$mode_{ALWIP}^{above} = -1 \text{로 표현하고,}$$

[0120] 이는 이 모드가 이용가능하지 않다는 것을 의미한다. 동일한 방식으로 그러나 좌측 PU가 현재 PU와 동일한 CTU에 속할 필요가 있다는 제한 없이, 모드 $mode_{ALWIP}^{left}$ 를 도출한다.

- [0121] 마지막으로, 3개의 고정된 디폴트 리스트 $list_{idx}$, $idx \in \{0,1,2\}$ 가 제공되고, 이들 각각은 3개의 별개의 ALWIP 모드를 포함한다. 디폴트 리스트 $list_{idx(PU)}$ 및 모드들 $mode_{ALWIP}^{above}$ 및 $mode_{ALWIP}^{left}$ 중에서, 반복들을 제거할 뿐만 아니라 디폴트 값들에 의해 -1을 치환함으로써 3개의 별개의 MPM을 구성한다.
- [0122] ALWIP MPM 리스트 구성에서 사용되는 좌측 이웃 블록 및 상부 이웃 블록은 도 10에 도시된 바와 같이 A1 및 B1이다.
- [0123] 2.5.7 종래의 루마 및 크로마 인트라 예측 모드들에 대한 적응된 MPM 리스트 도출
- [0124] 제안된 ALWIP 모드들은 다음과 같이 종래의 인트라 예측 모드들의 MPM 기반 코딩과 조화된다. 종래의 인트라-예측 모드들에 대한 루마 및 크로마 MPM-리스트 도출 프로세스들은 주어진 PU 상의 ALWIP-모드 $predmode_{ALWIP}$ 를 종래의 인트라-예측 모드들 중 하나에 매핑하는 고정된 테이블들 $map_alwip_to_angular_{idx}$, $idx \in \{0,1,2\}$ 를 사용한다.
- [0125] $predmode_{Angular} = map_alwip_to_angular_{idx(PU)}[predmode_{ALWIP}]$
- [0126] 루마 MPM-리스트 도출을 위해, ALWIP-모드 $predmode_{ALWIP}$ 를 사용하는 이웃 루마 블록이 직면될 때마다, 이 블록은 종래의 인트라-예측 모드 $predmode_{Angular}$ 을 사용하고 있었던 것처럼 취급된다. 크로마 MPM-list 도출을 위해, 현재 루마 블록이 LWIP-모드를 사용할 때마다, ALWIP 모드를 종래의 인트라 예측 모드로 변환하기 위해 동일한 매핑이 사용된다.
- [0127] 2.5.8 대응하는 수정된 작업 드래프트
- [0128] 일부 실시예들에서, 이 섹션에서 설명된 바와 같이, $intra_lwip_flag$, $intra_lwip_mpm_flag$, $intra_lwip_mpm_idx$ 및 $intra_lwip_mpm_remainder$ 과 관련된 부분들은 개시된 기술의 실시예들에 기초하여 작업 드래프트에 추가되었다.
- [0129] 일부 실시예들에서, 이 섹션에서 설명된 바와 같이, <begin> 및 <end> 태그들은 개시된 기술의 실시예들에 기초하여 작업 드래프트에 대한 추가들 및 수정들을 나타내기 위해 사용된다.
- [0130] 선택스 테이블들

[0131] 코딩 유닛 선택스

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	디스크립터
if(tile_group_type != I sps_abc_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA)	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] == 0 && tile_group_type != I)	
pred_mode_flag	ae(v)
if(((tile_group_type == I && cu_skip_flag[x0][y0] == 0) (tile_group_type != I && CuPredMode[x0][y0] != MODE_INTRA)) && sps_abc_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(Abs(Log2(cbWidth) - Log2(cbHeight)) <= 2)	
intra_lwip_flag[x0][y0]	ae(v)
if(intra_lwip_flag[x0][y0]) {	
intra_lwip_mpm_flag[x0][y0]	ae(v)
if(intra_lwip_mpm_flag[x0][y0])	
intra_lwip_mpm_idx[x0][y0]	ae(v)
else	
intra_lwip_mpm_remainder[x0][y0]	ae(v)
} else {	
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 */	
...	

[0132]

[0133] 시맨틱스

[0134] 1과 동일한 <begin>intra_lwip_flag[x0][y0]는 루마 샘플들에 대한 인트라 예측 타입이 아핀 선형 가중 인트라 예측인 것을 특징한다. 0과 동일한 intra_lwip_flag[x0][y0]는 루마 샘플들에 대한 인트라 예측 타입이 아핀 선형 가중 인트라 예측이 아니라는 것을 특징한다.

[0135] intra_lwip_flag[x0][y0]가 존재하지 않을 때, 이는 0과 동일한 것으로 추론된다.

[0136] 선택스 요소들 intra_lwip_mpm_flag[x0][y0], intra_lwip_mpm_idx[x0][y0] 및 intra_lwip_mpm_remainder[x0][y0]는 루마 샘플들에 대한 아핀 선형 가중 인트라 예측 모드를 특징한다. 어레이 인덱스들 x0, y0은 픽처의 상단 좌측 루마 샘플에 대한 고려된 코딩 블록의 상단 좌측 루마 샘플의 위치(x0 , y0)를 특징한다. intra_lwip_mpm_flag[x0][y0]이 1과 동일할 때, 아핀 선형 가중 인트라 예측 모드는 조항 8.4.X에 따라 이웃하는 인트라 예측된 코딩 유닛으로부터 추론된다.

[0137] intra_lwip_mpm_flag[x0][y0]이 존재하지 않을 때, 그것은 1과 동일한 것으로 추론된다.<end>

[0138] intra_subpartitions_split_flag[x0][y0]은 인트라 서브파티션 분할 타입이 수평인지 또는 수직인지를

특정한다. $\text{intra_subpartitions_split_flag}[x0][y0]$ 이 존재하지 않을 때, 이는 다음과 같이 추론된다:

[0139] - $\text{intra_lwp_flag}[x0][y0]$ 이 1과 동일한 경우, $\text{intra_subpartitions_split_flag}[x0][y0]$ 은 0과 동일한 것으로 추론된다.

[0140] - 그렇지 않으면, 다음이 적용된다:

[0141] - cbHeight 가 MaxTbSizeY 보다 큰 경우에, $\text{intra_subpartitions_split_flag}[x0][y0]$ 은 0과 동일한 것으로 추론된다.

[0142] - 그렇지 않으면(cbWidth 가 MaxTbSizeY 보다 큰 경우에), $\text{intra_subpartitions_split_flag}[x0][y0]$ 은 1과 동일한 것으로 추론된다.

[0143] 디코딩 프로세스

[0144] 8.4.1 인트라 예측 모드에서 코딩된 코딩 유닛들에 대한 일반적인 디코딩 프로세스

[0145] 이 프로세스에 대한 입력들은:

[0146] - 현재 픽처의 상단 좌측 루마 샘플에 대한 현재 코딩 블록의 상단 좌측 샘플을 특징하는 루마 위치(x_{Cb} , y_{Cb}),

[0147] - 루마 샘플들에서 현재 코딩 블록의 폭을 특징하는 변수 $cbWidth$,

[0148] - 루마 샘플들에서 현재 코딩 블록의 높이를 특징하는 변수 $cbHeight$,

[0149] - 단일 또는 이중 트리가 사용되는지를 특징하고, 이중 트리가 사용되는 경우에, 그것이 현재 트리가 루마 컴포넌트에 대응하는지 또는 크로마 컴포넌트에 대응하는지를 특징하는 변수 $treeType$ 이다.

[0150] 이러한 프로세스의 출력은 인-루프 필터링 전의 수정된 재구성된 픽처이다.

[0151] 조항 8.7.1에 특정된 양자화 파라미터들에 대한 도출 프로세스는 루마 위치(x_{Cb} , y_{Cb}), 루마 샘플들 내의 현재 코딩 블록의 폭 $cbWidth$ 및 루마 샘플들 내의 현재 코딩 블록의 높이 $cbHeight$, 및 변수 $treeType$ 를 입력들로 하여 호출된다.

[0152] $treeType$ 가 $SINGLE_TREE$ 와 동일하거나 $treeType$ 가 $DUAL_TREE_LUMA$ 와 동일할 때, 루마 샘플들에 대한 디코딩 프로세스는 다음과 같이 특정된다:

[0153] - $pcm_flag[x_{Cb}][y_{Cb}]$ 가 1과 동일한 경우, 재구성된 픽처는 다음과 같이 수정된다:

[0154] $S_L[x_{Cb} + i][y_{Cb} + j] =$

[0155]
$$pcm_sample_luma[(cbHeight * j) + i] \ll (BitDepth_L - PcmBitDepth_L),$$

(8-6)

[0156] 여기서 $i = 0..cbWidth - 1$, $j = 0..cbHeight - 1$

[0157] - 그렇지 않으면, 다음이 적용된다:

[0158] 1. 루마 인트라 예측 모드는 다음과 같이 도출된다:

[0159] - $\text{intra_lwp_flag}[x_{Cb}][y_{Cb}]$ 가 1과 동일한 경우, 조항 8.4.X에 특정된 바와 같은 아핀 선형 가중 인트라 예측 모드에 대한 도출 프로세스가 루마 위치(x_{Cb} , y_{Cb}), 루마 샘플들 내의 현재 코딩 블록의 폭 $cbWidth$ 및 루마 샘플들 내의 현재 코딩 블록의 높이 $cbHeight$ 를 입력으로 하여 호출된다.

[0160] - 그렇지 않으면, 조항 8.4.2에 특정된 바와 같은 루마 인트라 예측 모드에 대한 도출 프로세스는 루마 위치(x_{Cb} , y_{Cb}), 루마 샘플들 내의 현재 코딩 블록의 폭 $cbWidth$ 및 루마 샘플들 내의 현재 코딩 블록의 높이 $cbHeight$ 를 입력으로 하여 호출된다.

[0161] 2. 조항 8.4.4.1에 특정된 바와 같은 인트라 블록들에 대한 일반적인 디코딩 프로세스는 루마 위치(x_{Cb} , y_{Cb}), 트리 타입 $treeType$, $cbWidth$ 와 동일하게 설정된 변수 $nTbW$, $cbHeight$ 와 동일하게 설정된 변수 $nTbH$, $\text{IntraPredModeY}[x_{Cb}][y_{Cb}]$ 와 동일하게 설정된 변수 predModeIntra , 및 0과 동일하게 설정된 변수 $cIdx$ 를 입력들로 하여 호출되고, 출력은 인-루프 필터링 전의 수정된 재구성된 픽처이다.

- [0162] ...
- [0163] <begin>
- [0164] 8.4.X 아핀 선형 가중 인트라 예측 모드에 대한 도출 프로세스
- [0165] 이 프로세스에 대한 입력은 다음과 같다:
- [0166] - 현재 픽처의 상단 좌측 루마 샘플에 대한 현재 루마 코딩 블록의 상단 좌측 샘플을 특정하는 루마 위치 (x_{Cb} , y_{Cb}),
 - [0167] - 루마 샘플들에서 현재 코딩 블록의 폭을 특정하는 변수 $cbWidth$,
 - [0168] - 루마 샘플들에서 현재 코딩 블록의 높이를 특정하는 변수 $cbHeight$.
- [0169] 이 프로세스에서, 아핀 선형 가중 인트라 예측 모드 $IntraPredModeY[x_{Cb}][y_{Cb}]$ 가 도출된다.
- [0170] $IntraPredModeY[x_{Cb}][y_{Cb}]$ 는 다음의 순서화된 단계들에 의해 도출된다:
- [0171] 1. 이웃 위치들(x_{NbA} , y_{NbA}) 및 (x_{NbB} , y_{NbB})은 각각 ($x_{Cb} - 1$, y_{Cb}) 및 (x_{Cb} , $y_{Cb} - 1$)과 동일하게 설정된다.
 - [0172] 2. X 가 A 또는 B 로 대체되는 경우, 변수들 $candLwipModeX$ 는 다음과 같이 도출된다:
 - [0173] - 조항 6.4.X [에디션(Ed.) (BB): 이웃 블록들 이용가능성 검사 프로세스 추후결정(tbd)]에 특정된 바와 같은 블록에 대한 이용가능성 도출 프로세스는, (x_{Cb} , y_{Cb})과 동일하게 설정된 위치(x_{Curr} , y_{Curr}) 및 (x_{NbX} , y_{NbX})과 동일하게 설정된 이웃 위치(x_{NbY} , y_{NbY})를 입력들로 하여 호출되고, 출력은 $availableX$ 에 할당된다.
 - [0174] - 후보 아핀 선형 가중 인트라 예측 모드 $candLwipModeX$ 는 다음과 같이 도출된다:
 - [0175] - 다음의 조건들 중 하나 이상이 참인 경우, $candLwipModeX$ 는 -1과 동일하게 설정된다.
 - [0176] - 변수 $availableX$ 는 $FALSE$ 와 동일하다.
 - [0177] - $CuPredMode[x_{NbX}][y_{NbX}]$ 는 $MODE_INTRA$ 와 동일하지 않고 $mh_intra_flag[x_{NbX}][y_{NbX}]$ 는 1과 동일하지 않다.
 - [0178] - $pcm_flag[x_{NbX}][y_{NbX}]$ 는 1과 동일하다.
 - [0179] - X 는 B 와 동일하고, $y_{Cb} - 1$ 은 (($y_{Cb} >> CtbLog2SizeY$) $<< CtbLog2SizeY$)보다 작다.
 - [0180] - 그렇지 않으면, 다음이 적용된다:
 - [0181] - 조항 8.4.X.1에 특정된 바와 같은 블록에 대한 크기 타입 도출 프로세스는 루마 샘플들 $cbWidth$ 에서의 현재 코딩 블록의 폭 및 루마 샘플들 $cbHeight$ 에서의 현재 코딩 블록의 높이를 입력으로 하여 호출되고, 출력은 변수 $sizeId$ 에 할당된다.
 - [0182] - $intra_lwip_flag[x_{NbX}][y_{NbX}]$ 가 1과 동일한 경우, 조항 8.4.X.1에 특정된 바와 같은 블록에 대한 크기 타입 도출 프로세스가 루마 샘플들 내의 이웃 코딩 블록의 폭 $nbWidthX$ 및 루마 샘플들 내의 이웃 코딩 블록의 높이 $nbHeightX$ 를 입력으로 하여 호출되고, 출력은 변수 $sizeIdX$ 에 할당된다.
 - [0183] - $sizeId$ 가 $sizeIdX$ 와 동일한 경우에, $candLwipModeX$ 는 $IntraPredModeY[x_{NbX}][y_{NbX}]$ 와 동일하게 설정된다.
 - [0184] - 그렇지 않으면, $candLwipModeX$ 는 -1과 동일하게 설정된다.
 - [0185] - 그렇지 않으면, $candLwipModeX$ 는 표 8-X1에 특정된 바와 같이 $IntraPredModeY[x_{NbX}][y_{NbX}]$ 및 $sizeId$ 를 사용하여 도출된다.
- [0186] 3. $x = 0..2$ 인 $candLwipModeList[x]$ 는 표 8-X2에 특정된 바와 같은 $lwipMpmCand[sizeId]$ 를 사용하여 다음과 같이 도출된다:
- [0187] - $candLwipModeA$ 및 $candLwipModeB$ 둘 다 -1과 동일한 경우, 다음이 적용된다:

- [0188] $candLwipModeList[0] = lwipMpmCand[sizeId][0]$ (8-X1)
- [0189] $candLwipModeList[1] = lwipMpmCand[sizeId][1]$ (8-X2)
- [0190] $candLwipModeList[2] = lwipMpmCand[sizeId][2]$ (8-X3)
- [0191] - 그렇지 않으면, 다음이 적용된다:
- [0192] - $candLwipModeA$ 가 $candLwipModeB$ 와 동일하거나 $candLwipModeA$ 또는 $candLwipModeB$ 중 어느 하나가 -1과 동일한 경우에, 다음이 적용된다:
- [0193] $candLwipModeList[0] = (candLwipModeA \neq -1) ? candLwipModeA : candLwipModeB$ (8-X4)
- [0194] - $candLwipModeList[0]$ 이 $lwipMpmCand[sizeId][0]$ 과 동일한 경우에, 다음이 적용된다:
- [0195] $candLwipModeList[1] = lwipMpmCand[sizeId][1]$ (8-X5)
- [0196] $candLwipModeList[2] = lwipMpmCand[sizeId][2]$ (8-X6)
- [0197] - 그렇지 않으면, 다음이 적용된다:
- [0198] $candLwipModeList[1] = lwipMpmCand[sizeId][0]$ (8-X7)
- [0199] $candLwipModeList[2] = (candLwipModeList[0] \neq lwipMpmCand[sizeId][1]) ?$
- [0200] $lwipMpmCand[sizeId][1] : lwipMpmCand[sizeId][2]$ (8-X8)
- [0201] - 그렇지 않으면, 다음이 적용된다:
- [0202] $candLwipModeList[0] = candLwipModeA$ (8-X9)
- [0203] $candLwipModeList[1] = candLwipModeB$ (8-X10)
- [0204] - $candLwipModeA$ 및 $candLwipModeB$ 가 둘 다 $lwipMpmCand[sizeId][0]$ 과 동일하지 않는 경우에, 다음이 적용된다:
- [0205] $candLwipModeList[2] = lwipMpmCand[sizeId][0]$ (8-X11)
- [0206] - 그렇지 않으면, 다음이 적용된다:
- [0207] - $candLwipModeA$ 및 $candLwipModeB$ 둘 다 $lwipMpmCand[sizeId][1]$ 이 아닌 경우에, 다음이 적용된다:
- [0208] $candLwipModeList[2] = lwipMpmCand[sizeId][1]$ (8-X12)
- [0209] - 그렇지 않으면, 다음이 적용된다:
- [0210] $candLwipModeList[2] = lwipMpmCand[sizeId][2]$ (8-X13)
- [0211] 4. $IntraPredModeY[xCb][yCb]$ 는 다음의 절차를 적용함으로써 도출된다:
- [0212] - $intra_lwip_mpm_flag[xCb][yCb]$ 가 1과 동일한 경우, $IntraPredModeY[xCb][yCb]$ 는 $candLwipModeList[intra_lwip_mpm_idx[xCb][yCb]]$ 와 동일하게 설정된다.
- [0213] - 그렇지 않으면, $IntraPredModeY[xCb][yCb]$ 는 다음의 순서화된 단계들을 적용함으로써 도출된다:
- [0214] 1. $i = 0..1$ 인 경우 그리고 각각의 i 에 대해 $j = (i + 1)..2$ 인 경우 $candLwipModeList[i]$ 가 $candLwipModeList[j]$ 보다 클 때, 둘 다의 값들은 다음과 같이 스와핑된다:
- [0215] $(candLwipModeList[i], candLwipModeList[j]) = Swap(candLwipModeList[i], candLwipModeList[j])$ (8-X14)
- [0216] 2. $IntraPredModeY[xCb][yCb]$ 는 다음의 순서화된 단계들에 의해 도출된다:
- [0217] i. $IntraPredModeY[xCb][yCb]$ 는 $intra_lwip_mpm_remainder[xCb][yCb]$ 와 동일하게 설정된다.
- [0218] ii. i 가 0 내지 2(경계 포함)와 동일한 경우, $IntraPredModeY[xCb][yCb]$ 가 $candLwipModeList[i]$ 이상일 때, $IntraPredModeY[xCb][yCb]$ 의 값은 1씩 증분된다.

[0219] $x = xCb..xCb + cbWidth - 1$ 및 $y = yCb..yCb + cbHeight - 1$ 인 변수 $IntraPredModeY[x][y]$ 는 $IntraPredModeY[xCb][yCb]$ 와 동일하게 설정된다.

[0220] 8.4.X.1 예측 블록 크기 타입에 대한 도출 프로세스

[0221] 이 프로세스에 대한 입력은 다음과 같다:

[0222] - 루마 샘플들에서 현재 코딩 블록의 폭을 특징하는 변수 $cbWidth$,

[0223] - 루마 샘플들에서 현재 코딩 블록의 높이를 특징하는 변수 $cbHeight$.

[0224] 이 프로세스의 출력은 변수 $sizeId$ 이다.

[0225] 변수 $sizeId$ 는 다음과 같이 도출된다:

[0226] - $cbWidth$ 와 $cbHeight$ 둘 다 4와 동일한 경우, $sizeId$ 는 0으로 설정된다.

[0227] - 그렇지 않고, $cbWidth$ 및 $cbHeight$ 둘 다 8 이하인 경우, $sizeId$ 는 1과 동일하게 설정된다.

[0228] - 그렇지 않으면, $sizeId$ 는 2와 동일하게 설정된다.

[0229] 표 8-X1 - 인트라 예측 모드와 아핀 선형 가중 인트라 예측 모드들 사이의 매핑의 사양

$IntraPredModeY[xNbX][yNbX]$	블록 크기 타입 $sizeId$		
	0	1	2
0	17	0	5
1	17	0	1
2, 3	17	10	3
4, 5	9	10	3
6, 7	9	10	3
8, 9	9	10	3
10, 11	9	10	0
12, 13	17	4	0
14, 15	17	6	0
16, 17	17	7	4
18, 19	17	7	4
20, 21	17	7	4
22, 23	17	5	5
24, 25	17	5	1
26, 27	5	0	1
28, 29	5	0	1
30, 31	5	3	1
32, 33	5	3	1
34, 35	34	12	6
36, 37	22	12	6
38, 39	22	12	6
40, 41	22	12	6
42, 43	22	14	6
44, 45	34	14	10
46, 47	34	14	10
48, 49	34	16	9
50, 51	34	16	9
52, 53	34	16	9
54, 55	34	15	9
56, 57	34	13	9
58, 59	26	1	8
60, 61	26	1	8
62, 63	26	1	8
64, 65	26	1	8
66	26	1	8

[0230]

표 8-X2 - 아핀 선형 가중 인트라 예측 후보 모드들의 사양

	후보 모드		
	0	1	2
<i>hwipMpmCand</i> [0]	17	34	5
<i>hwipMpmCand</i> [1]	0	7	16
<i>hwipMpmCand</i> [2]	1	4	6

<end>

8.4.2. 루마 인트라 예측 모드에 대한 도출 프로세스

이 프로세스에 대한 입력은 다음과 같다:

- 현재 픽처의 상단 좌측 루마 샘플에 대한 현재 루마 코딩 블록의 상단 좌측 샘플을 특징하는 루마 위치 (*xCb* , *yCb*),
- 루마 샘플들에서 현재 코딩 블록의 폭을 특징하는 변수 *cbWidth*,
- 루마 샘플들에서 현재 코딩 블록의 높이를 특징하는 변수 *cbHeight*.

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

표 8-1은 인트라 예측 모드 *IntraPredModeY*[*xCb*][*yCb*]에 대한 값 및 연관된 명칭들을 특징한다.

표 8-1 - 인트라 예측 모드의 사양 및 연관된 명칭들

인트라 예측 모드	연관된 명칭
0	<i>INTRA_PLANAR</i>
1	<i>INTRA_DC</i>
2..66	<i>INTRA_ANGULAR2</i> .. <i>INTRA_ANGULAR66</i>
81..83	<i>INTRA_LT_CCLM</i> , <i>INTRA_L_CCLM</i> , <i>INTRA_T_CCLM</i>

노트 - : 인트라 예측 모드들 *INTRA_LT_CCLM*, *INTRA_L_CCLM* 및 *INTRA_T_CCLM*은 크로마 컴포넌트들에만 적용 가능하다.

IntraPredModeY[*xCb*][*yCb*]는 다음의 순서화된 단계들에 의해 도출된다:

1. 이웃 위치들 (*xNbA* , *yNbA*) 및 (*xNbB* , *yNbB*)은 각각 (*xCb* - 1, *yCb* + *cbHeight* - 1) 및 (*xCb* + *cbWidth* - 1, *yCb* - 1)과 동일하게 설정된다.
2. *X*가 *A* 또는 *B*로 대체되는 경우, 변수들 *candIntraPredModeX*는 다음과 같이 도출된다:
 - 조항 <begin>6.4.X [에디션(Ed.) (BB): 이웃 블록들 이용가능성 검사 프로세스 추후결정(tbd)] <end>에 특정된 바와 같은 블록에 대한 이용가능성 도출 프로세스는, (*xCb* , *yCb*)와 동일하게 설정된 위치(*xCurr* , *yCurr*) 및 (*xNbX* , *yNbX*)와 동일하게 설정된 이웃 위치(*xNbY* , *yNbY*)를 입력들로 하여 호출되고, 출력은 *availableX*에 할당된다.
 - 후보 인트라 예측 모드 *candIntraPredModeX*는 다음과 같이 도출된다:
 - 다음의 조건들 중 하나 이상이 참(true)인 경우, *candIntraPredModeX*는 *INTRA_PLANAR*과 동일하게 설정된다.
 - 변수 *availableX*는 *FALSE*와 동일하다.
 - *CuPredMode*[*xNbX*][*yNbX*]는 *MODE_INTRA*와 동일하지 않고 *ciip_flag*[*xNbX*][*yNbX*]는 1과 동일하지 않다.
 - *pcm_flag*[*xNbX*][*yNbX*]는 1과 동일하다.
 - *X*는 *B*와 동일하고, *yCb* - 1은 ((*yCb* >> *CtbLog2SizeY*) << *CtbLog2SizeY*)보다 작다.

- [0254] - 그렇지 않으면, *candIntraPredModeX*는 다음과 같이 도출된다:
- [0255] - *intra_lwip_flag[xCb][yCb]*가 1과 동일한 경우에, *candIntraPredModeX*는 다음의 순서화된 단계들에 의해 도출된다:
- [0256] *i.* 조항 8.4.X.1에 특정된 바와 같은 블록에 대한 크기 타입 도출 프로세스는 루마 샘플들 *cbWidth*에서의 현재 코딩 블록의 폭 및 루마 샘플들 *cbHeight*에서의 현재 코딩 블록의 높이를 입력으로 하여 호출되고, 출력은 변수 *sizeId*에 할당된다.
- [0257] *ii.* *candIntraPredModeX*는 표 8-X3에 특정된 바와 같이 *IntraPredModeY[xNbX][yNbX]* 및 *sizeId*를 사용하여 도출된다.
- [0258] - 그렇지 않으면, *candIntraPredModeX*는 *IntraPredModeY[xNbX][yNbX]*와 동일하게 설정된다.
- [0259] 3. 변수들 *ispDefaultMode1* 및 *ispDefaultMode2*는 다음과 같이 정의된다:
- [0260] - *IntraSubPartitionsSplitType*가 *ISP_HOR_SPLIT*와 동일한 경우, *ispDefaultMode1*은 *INTRA_ANGULAR18*과 동일하게 설정되고 *ispDefaultMode2*는 *INTRA_ANGULAR5*와 동일하게 설정된다.
- [0261] - 그렇지 않으면, *ispDefaultMode1*은 *INTRA_ANGULAR50*과 동일하게 설정되고, *ispDefaultMode2*는 *INTRA_ANGULAR63*과 동일하게 설정된다.
- [0262] ...

[0263] 표 8-X3 - 아핀 선형 가중 인트라 예측 모드와 인트라 예측 모드 사이의 매핑의 사양

IntraPredModeY[xNbX][yNbX]	블록 크기 타입 sizeId		
	0	1	2
0	0	0	1
1	18	1	1
2	18	0	1
3	0	1	1
4	18	0	18
5	0	22	0
6	12	18	1
7	0	18	0
8	18	1	1
9	2	0	50
10	18	1	0
11	12	0	
12	18	1	
13	18	0	
14	1	44	
15	18	0	
16	18	50	
17	0	1	
18	0	0	
19	50		
20	0		
21	50		
22	0		
23	56		
24	0		
25	50		
26	66		
27	50		
28	56		
29	50		
30	50		
31	1		
32	50		
33	50		
34	50		

[0264]

[0265] 8.4.3 크로마 인트라 예측 모드의 도출 프로세스

[0266] 이 프로세스에 대한 입력은 다음과 같다:

[0267] - 현재 픽처의 상단 좌측 루마 샘플에 대한 현재 크로마 코딩 블록의 상단 좌측 샘플을 특징하는 루마 위치(*xCb*, *yCb*),

[0268] - 루마 샘플들에서 현재 코딩 블록의 폭을 특징하는 변수 *cbWidth*,

[0269] - 루마 샘플들에서 현재 코딩 블록의 높이를 특징하는 변수 *cbHeight*.

[0270] 이 프로세스에서, 크로마 인트라 예측 모드 *IntraPredModeC*[*xCb*][*yCb*]가 도출된다.

[0271] 대응하는 루마 인트라 예측 모드 *lumaIntraPredMode*는 다음과 같이 도출된다:

[0272] - *intra_lwip_flag*[*xCb*][*yCb*]가 1과 동일한 경우에, *lumaIntraPredMode*는 다음의 순서화된 단계들에 의해 도출된다:

[0273] i. 조항 8.4.X.1에 특정된 바와 같은 블록에 대한 크기 타입 도출 프로세스는 루마 샘플들 *cbWidth*에서의 현재 코딩 블록의 폭 및 루마 샘플들 *cbHeight*에서의 현재 코딩 블록의 높이를 입력으로 하여 호출되고, 출력은 변수

*sizeId*에 할당된다.

- [0274] ii. 루마 인트라 예측 모드는 표 8-X3에 특정된 바와 같이 $\text{IntraPredModeY}[x_{Cb} + cbWidth / 2][y_{Cb} + cbHeight / 2]$ 및 *sizeId*를 사용하여 도출되고, *candIntraPredModeX*의 값을 *lumaIntraPredMode*에 할당한다.
- [0275] - 그렇지 않으면, *lumaIntraPredMode*는 $\text{IntraPredModeY}[x_{Cb} + cbWidth / 2][y_{Cb} + cbHeight / 2]$ 와 동일하게 설정된다.
- [0276] 크로마 인트라 예측 모드 $\text{IntraPredModeC}[x_{Cb}][y_{Cb}]$ 는 표 8-2 및 표 8-3에 특정된 바와 같이 *intra_chroma_pred_mode*[*xCb*][*yCb*] 및 *lumaIntraPredMode*를 사용하여 도출된다.
- [0277] ...
- [0278] xxx. 인트라 샘플 예측
- [0279] <begin>
- [0280] 이 프로세스에 대한 입력들은:
- [0281] - 현재 픽처의 상단 좌측 샘플에 대한 현재 변환 블록의 상단 좌측 샘플을 특정하는 샘플 위치(*xTbCmp*, *yTbCmp*),
- [0282] - 인트라 예측 모드를 특정하는 변수 *predModeIntra*,
- [0283] - 변환 블록 폭을 특정하는 변수 *nTbW*,
- [0284] - 변환 블록 높이를 특정하는 변수 *nTbH*,
- [0285] - 코딩 블록 폭을 특정하는 변수 *nCbW*,
- [0286] - 코딩 블록 높이를 특정하는 변수 *nCbH*,
- [0287] - 현재 블록의 컬러 컴포넌트를 특정하는 변수 *cIdx*이다.
- [0288] 이러한 프로세스의 출력들은, $x = 0..nTbW - 1$, $y = 0..nTbH - 1$ 인 예측된 샘플들 *predSamples*[*x*][*y*]이다.
- [0289] 예측된 샘플들 *predSamples*[*x*][*y*]는 다음과 같이 도출된다:
- [0290] - *intra_lwip_flag*[*xTbCmp*][*yTbCmp*]가 1과 동일하고 *cIdx*가 0과 동일한 경우, 조항 8.4.4.2.X1에 특정된 바와 같은 아핀 선형 가중 인트라 샘플 예측 프로세스가 위치(*xTbCmp*, *yTbCmp*), 인트라 예측 모드 *predModeIntra*, 변환 블록 폭 *nTbW* 및 높이 *nTbH*를 입력들로 하여 호출되고, 출력은 *predSamples*이다.
- [0291] - 그렇지 않으면, 조항 8.4.4.2.X1에 특정된 바와 같은 일반적인 인트라 샘플 예측 프로세스는 위치(*xTbCmp*, *yTbCmp*), 인트라 예측 모드 *predModeIntra*, 변환 블록 폭 *nTbW* 및 높이 *nTbH*, 코딩 블록 폭 *nCbW* 및 높이 *nCbH*, 및 변수 *cIdx*를 입력들로 하여 호출되고, 출력은 *predSamples*이다.
- [0292] 8.4.4.2. X1 아핀 선형 가중 인트라 샘플 예측
- [0293] 이 프로세스에 대한 입력들은:
- [0294] - 현재 픽처의 상단 좌측 샘플에 대한 현재 변환 블록의 상단 좌측 샘플을 특정하는 샘플 위치(*xTbCmp*, *yTbCmp*),
- [0295] - 인트라 예측 모드를 특정하는 변수 *predModeIntra*,
- [0296] - 변환 블록 폭을 특정하는 변수 *nTbW*,
- [0297] - 변환 블록 높이를 특정하는 변수 *nTbH*이다.
- [0298] 이러한 프로세스의 출력들은, $x = 0..nTbW - 1$, $y = 0..nTbH - 1$ 인 예측된 샘플들 *predSamples*[*x*][*y*]이다.
- [0299] 조항 8.4.X.1에 특정된 바와 같은 블록에 대한 크기 타입 도출 프로세스는 변환 블록 폭 *nTbW* 및 변환 블록 높이 *nTbH*를 입력으로 하여 호출되고, 출력은 변수 *sizeId*에 할당된다.

[0300] 변수들 $numModes$, $boundarySize$, $predW$, $predH$ 및 $predC$ 는 표 8-X4에 특정된 바와 같이 $sizeId$ 를 사용하여 도출된다.

[0301] 표 8-X4 - $sizeId$ 에 따른 모드들의 수, 경계 샘플 크기 및 예측 크기들의 사양

$sizeId$	$numModes$	$boundarySize$	$predW$	$predH$	$predC$
0	35	2	4	4	4
1	19	4	4	4	4
2	11	4	$Min(nTbW, 8)$	$Min(nTbH, 8)$	8

[0302]

[0303] 플래그 $isTransposed$ 는 다음과 같이 도출된다: $isTransposed = (predModeIntra > (numModes / 2)) ? 1 : 0$ (8-X15)

[0304] 플래그들 $needUpsBdryHor$ 및 $needUpsBdryVer$ 은 다음과 같이 도출된다:

[0305] $needUpsBdryHor = (nTbW > predW) ? TRUE : FALSE$ (8-X16)

[0306] $needUpsBdryVer = (nTbH > predH) ? TRUE : FALSE$ (8-X17)

[0307] 변수들 $upsBdryW$ 및 $upsBdryH$ 는 다음과 같이 도출된다:

[0308] $upsBdryW = (nTbH > nTbW) ? nTbW : predW$ (8-X18)

[0309] $upsBdryH = (nTbH > nTbW) ? predH : nTbH$ (8-X19)

[0310] 변수들 $lwipW$ 및 $lwipH$ 는 다음과 같이 도출된다:

[0311] $lwipW = (isTransposed == 1) ? predH : predW$ (8-X20)

[0312] $lwipH = (isTransposed == 1) ? predW : predH$ (8-X21)

[0313] 참조 샘플들 $refT[x]$ ($x = 0..nTbW - 1$ 임) 및 $refL[y]$ ($y = 0..nTbH - 1$ 임)의 생성을 위해, 조항 8.4.4.2.X2에 특정된 바와 같은 참조 샘플 도출 프로세스가 샘플 위치($xTbCmp$, $yTbCmp$), 변환 블록 폭 $nTbW$, 변환 블록 높이 $nTbH$ 를 입력들로 하고, 각각, 상단 및 좌측 참조 샘플들 $refT[x]$ ($x = 0..nTbW - 1$ 임) 및 $refL[y]$ ($y = 0..nTbH - 1$ 임)를 출력들로 하여 호출된다.

[0314] $x = 0..2 * boundarySize - 1$ 인 경계 샘플들 $p[x]$ 의 생성을 위해, 다음이 적용된다:

[0315] - 조항 8.4.4.2.X3에 특정된 바와 같은 경계 축소 프로세스는 상단 참조 샘플들에 대해 블록 크기 $nTbW$, 참조 샘플들 $refT$, 경계 크기 $boundarySize$, 업샘플링 경계 플래그 $needUpsBdryVer$, 및 업샘플링 경계 크기 $upsBdryW$ 를 입력들로 하고, $x = 0..boundarySize - 1$ 인 감소된 경계 샘플들 $redT[x]$ 및 $x = 0..upsBdryW - 1$ 인 업샘플링 경계 샘플들 $upsBdryT[x]$ 를 출력들로 하여 호출된다.

[0316] - 조항 8.4.4.2.X3에 특정된 바와 같은 경계 축소 프로세스는 좌측 참조 샘플들에 대해 블록 크기 $nTbH$, 참조 샘플들 $refL$, 경계 크기 $boundarySize$, 업샘플링 경계 플래그 $needUpsBdryHor$, 및 업샘플링 경계 크기 $upsBdryH$ 를 입력들로 하고, $x = 0..boundarySize - 1$ 인 감소된 경계 샘플들 $redL[x]$ 및 $x = 0..upsBdryH - 1$ 인 업샘플링 경계 샘플들 $upsBdryL[x]$ 를 출력들로 하여 호출된다.

[0317] - 감소된 상단 및 좌측 경계 샘플들 $redT$ 및 $redL$ 은 다음과 같이 경계 샘플 어레이 p 에 할당된다:

[0318] - $isTransposed$ 가 1과 동일한 경우에, $p[x]$ 는 $x = 0..boundarySize - 1$ 인 $redL[x]$ 와 동일하게 설정되고, $p[x + boundarySize]$ 는 $x = 0..boundarySize - 1$ 인 $redT[x]$ 와 동일하게 설정된다.

[0319] - 그렇지 않으면, $p[x]$ 는 $x = 0..boundarySize - 1$ 인 $redT[x]$ 와 동일하게 설정되고, $p[x + boundarySize]$ 는 $x = 0..boundarySize - 1$ 인 $redL[x]$ 와 동일하게 설정된다.

[0320] $predModeIntra$ 에 따른 인트라 샘플 예측 프로세스에 대해, 다음의 순서화된 단계들이 적용된다:

[0321] 1. $x = 0..lwipW - 1$, $y = 0..lwipH - 1$ 인 아핀 선형 가중 샘플들 $predLwip[x][y]$ 는 다

음과 같이 도출된다:

- 변수 $modeId$ 는 다음과 같이 도출된다:

$$modeId = predModeIntra - (isTransposed == 1) ? (numModes / 2) : 0 \quad (8-X22)$$

- $x = 0..2 * boundarySize - 1$, $y = 0..predC * predC - 1$ 인 가중치 행렬 $mWeight[x][y]$ 는 표 8-XX [추후결정(TBD): 가중치 행렬들을 추가]에 특정된 바와 같이 $sizeId$ 및 $modeId$ 를 사용하여 도출된다.

- $y = 0..predC * predC - 1$ 인 바이어스 벡터 $vBias[y]$ 는 표 8-XX [추후결정(TBD): 바이어스 벡터들 추가]에 특정된 바와 같이 $sizeId$ 및 $modeId$ 를 사용하여 도출된다.

- 변수 sW 는 표 8-X5에 특정된 바와 같이 $sizeId$ 및 $modeId$ 를 사용하여 도출된다.

- $x = 0..lwipW - 1$, $y = 0..lwipH - 1$ 인 아핀 선형 가중 샘플들 $predLwip[x][y]$ 는 다음과 같이 도출된다:

$$oW = 1 \ll (sW - 1) \quad (8-X23)$$

$$sB = BitDepth_N - 1 \quad (8-X24)$$

$$incW = (predC > lwipW) ? 2 : 1 \quad (8-X25)$$

$$incH = (predC > lwipH) ? 2 : 1 \quad (8-X26)$$

$$predLwip[x][y] = (\sum_{i=0}^{2 * boundarySize - 1} mWeight[i][y * incH * predC + x * incW] * p[i]) + (vBias[y * incH * predC + x * incW] \ll sB) \gg sW \quad (8-X27)$$

2. $x = 0..nTbW - 1$, $y = 0..nTbH - 1$ 인 예측된 샘플 $predSamples[x][y]$ 는 다음과 같이 도출된다:

- $isTransposed$ 가 1과 동일할 때, $x = 0..predW - 1$, $y = 0..predH - 1$ 인 $predLwip[x][y]$ 는 $predLwip[y][x]$ 와 동일하게 설정된다.

- $needUpsBdryVer$ 이 TRUE와 동일하거나 $needUpsBdryHor$ 이 TRUE와 동일한 경우, 조항 8.4.4.2.X4에 특정된 예측 업샘플링 프로세스는 입력 블록 폭 $predW$, 입력 블록 높이 $predH$, 아핀 선형 가중 샘플들 $predLwip$, 변환 블록 폭 $nTbW$, 변환 블록 높이 $nTbH$, 업샘플링 경계 폭 $upsBdryW$, 업샘플링 경계 높이 $upsBdryH$, 상단 업샘플링 경계 샘플들 $upsBdryT$, 및 좌측 업샘플링 경계 샘플들 $upsBdryL$ 을 입력들로 하여 호출되고, 출력은 예측된 샘플 어레이 $predSamples$ 이다.

- 그렇지 않으면, $x = 0..nTbW - 1$, $y = 0..nTbH - 1$ 인 $predSamples[x][y]$ 는 $predLwip[x][y]$ 와 동일하게 설정된다.

표 8-X5 - $sizeId$ 및 $modeId$ 에 따른 가중치 시프트들 sW 의 사양

	<i>modeId</i>																	
<i>sizeId</i>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
1	8	8	8	9	8	8	8	8	9	8								
2	8	8	8	8	8	8												

8.4.4.2. X2 참조 샘플 도출 프로세스

이 프로세스에 대한 입력들은:

- 현재 픽처의 상단 좌측 루마 샘플에 대한 현재 변환 블록의 상단 좌측 루마 샘플을 특정하는 샘플 위치 ($xTbY$, $yTbY$),

- 변환 블록 폭을 특정하는 변수 $nTbW$,

- [0343] - 변환 블록 높이를 특정하는 변수 $nTbH$ 이다.
- [0344] 이 프로세스의 출력들은 각각 상단 및 좌측 참조 샘플들 $refT[x]$ ($x = 0..nTbW - 1$ 임) 및 $refL[y]$ ($y = 0..nTbH - 1$ 임)이다.
- [0345] 이웃 샘플들 $refT[x]$ ($x = 0..nTbW - 1$ 임) 및 $refL[y]$ ($y = 0..nTbH - 1$ 임)는 인-루프 필터 프로세스 전에 구성된 샘플들이고 다음과 같이 도출된다:
- [0346] - 상단 및 좌측 이웃 루마 위치들 ($xNbT$, $yNbT$) 및 ($xNbL$, $yNbL$)은 다음에 의해 특정된다:
- [0347] ($xNbT$, $yNbT$) = ($xTbY + x$, $yTbY - 1$) (8-X28)
- [0348] ($xNbL$, $yNbL$) = ($xTbY - 1$, $yTbY + y$) (8-X29)
- [0349] - 조항 6.4.X [에디션(Ed.) (BB): 이웃 블록들 이용가능성 검사 프로세스 추후결정(tbd)]에 특정된 바와 같은 블록에 대한 이용가능성 도출 프로세스는, ($xTbY$, $yTbY$)와 동일하게 설정된 현재 루마 위치($xCurr$, $yCurr$) 및 상단 이웃 루마 위치($xNbT$, $yNbT$)를 입력들로 하여 호출되고, 출력은 $x = 0..nTbW - 1$ 인 $availTop[x]$ 에 할당된다.
- [0350] - 조항 6.4.X [에디션(Ed.) (BB): 이웃 블록들 이용가능성 검사 프로세스 추후결정(tbd)]에 특정된 바와 같은 블록에 대한 이용가능성 도출 프로세스는, ($xTbY$, $yTbY$)와 동일하게 설정된 현재 루마 위치($xCurr$, $yCurr$) 및 좌측 이웃 루마 위치($xNbL$, $yNbL$)를 입력들로 하여 호출되고, 출력은 $y = 0..nTbH - 1$ 인 $availLeft[y]$ 에 할당된다.
- [0351] - $x = 0..nTbW - 1$ 인 상단 참조 샘플들 $refT[x]$ 는 다음과 같이 도출된다:
- [0352] - $x = 0..nTbW - 1$ 인 모든 $availTop[x]$ 가 TRUE와 동일한 경우에, 위치($xNbT$, $yNbT$)에서의 샘플은 $x = 0..nTbW - 1$ 인 $refT[x]$ 에 할당된다.
- [0353] - 그렇지 않고, $availTop[0]$ 이 FALSE와 동일한 경우에, $x = 0..nTbW - 1$ 인 모든 $refT[x]$ 는 $1 \ll (BitDepth_i - 1)$ 과 동일하게 설정된다.
- [0354] - 그렇지 않으면, $x = 0..nTbW - 1$ 인 참조 샘플들 $refT[x]$ 는 다음의 순서화된 단계들에 의해 도출된다:
- [0355] 1. 변수 $lastT$ 는 FALSE와 동일한 시퀀스 $availTop[x]$ ($x = 1..nTbW - 1$ 임) 내의 제1 요소의 위치 x 와 동일하게 설정된다.
- [0356] 2. 모든 $x = 0..lastT - 1$ 에 대해, 위치($xNbT$, $yNbT$)에서의 샘플은 $refT[x]$ 에 할당된다.
- [0357] 3. 모든 $x = lastT..nTbW - 1$ 에 대해, $refT[x]$ 는 $refT[lastT - 1]$ 과 동일하게 설정된다.
- [0358] - $x = 0..nTbH - 1$ 인 좌측 참조 샘플들 $refL[y]$ 는 다음과 같이 도출된다:
- [0359] - $y = 0..nTbH - 1$ 인 모든 $availLeft[y]$ 가 TRUE와 동일한 경우에, 위치($xNbL$, $yNbL$)에서의 샘플은 $y = 0..nTbH - 1$ 인 $refL[y]$ 에 할당된다.
- [0360] - 그렇지 않고, $availLeft[0]$ 이 FALSE와 동일한 경우에, $y = 0..nTbH - 1$ 인 모든 $refL[y]$ 는 $1 \ll (BitDepth_i - 1)$ 과 동일하게 설정된다.
- [0361] - 그렇지 않으면, $y = 0..nTbH - 1$ 인 참조 샘플들 $refL[y]$ 는 다음의 순서화된 단계들에 의해 도출된다:
- [0362] 1. 변수 $lastL$ 은 FALSE와 동일한 시퀀스 $availLeft[y]$ ($y = 1..nTbH - 1$ 임) 내의 제1 요소의 위치 y 와 동일하게 설정된다.
- [0363] 2. 모든 $y = 0..lastL - 1$ 에 대해, 위치($xNbL$, $yNbL$)에서의 샘플은 $refL[y]$ 에 할당된다.
- [0364] 3. 모든 $y = lastL..nTbH - 1$ 에 대해, $refL[y]$ 는 $refL[lastL - 1]$ 과 동일하게 설정된다.
- [0365] 경계 축소 프로세스의 사양
- [0366] 이 프로세스에 대한 입력들은:

- [0367] - 변환 블록 크기를 특정하는 변수 $nTbX$,
- [0368] - $x = 0..nTbX - 1$ 인 참조 샘플들 $refX[x]$,
- [0369] - 다운샘플링된 경계 크기를 특정하는 변수 $boundarySize$,
- [0370] - 중간 경계 샘플들이 업샘플링을 위해 요구되는지를 특정하는 플래그 $needUpsBdryX$,
- [0371] - 업샘플링을 위한 경계 크기를 특정하는 변수 $upsBdrySize$ 이다.
- [0372] 이 프로세스의 출력들은 감소된 경계 샘플들 $redX[x]$ ($x = 0..boundarySize - 1$ 임) 및 업샘플링 경계 샘플 $upsBdryX[x]$ ($x = 0..upsBdrySize - 1$ 임)이다.
- [0373] $x = 0..upsBdrySize - 1$ 인 업샘플링 경계 샘플들 $upsBdryX[x]$ 는 다음과 같이 도출된다:
- [0374] - $needUpsBdryX$ 가 $TRUE$ 와 동일하고 $upsBdrySize$ 가 $nTbX$ 보다 작은 경우, 다음과 같이 적용된다:
- [0375]
$$uDwn = nTbX / upsBdrySize \quad (8-X30)$$
- [0376]
$$upsBdryX[x] = (\sum_{i=0}^{uDwn-1} refX[x * uDwn + i] + (1 << (Log2(uDwn) - 1))) >> Log2(uDwn) \quad (8-X31)$$
- [0377] - 그렇지 않으면($upsBdrySize$ 가 $nTbX$ 와 동일하면), $upsBdryX[x]$ 는 $refX[x]$ 와 동일하게 설정된다.
- [0378] $x = 0..boundarySize - 1$ 인 감소된 경계 샘플들 $redX[x]$ 는 다음과 같이 도출된다:
- [0379] - $boundarySize$ 가 $upsBdrySize$ 보다 작은 경우, 다음이 적용된다:
- [0380]
$$bDwn = upsBdrySize / boundarySize \quad (8-X32)$$
- [0381]
$$redX[x] = (\sum_{i=0}^{bDwn-1} upsBdryX[x * bDwn + i] + (1 << (Log2(bDwn) - 1))) >> Log2(bDwn) \quad (8-X33)$$
- [0382] - 그렇지 않으면($boundarySize$ 가 $upsBdrySize$ 와 동일하면), $redX[x]$ 는 $upsBdryX[x]$ 와 동일하게 설정된다.
- [0383] 8.4.4.2. X4 예측 업샘플링 프로세스의 사양
- [0384] 이 프로세스에 대한 입력들은:
- [0385] - 입력 블록 폭을 특정하는 변수 $predW$,
- [0386] - 입력 블록 높이를 특정하는 변수 $predH$,
- [0387] - $x = 0..predW - 1$, $y = 0..predH - 1$ 인 아핀 선형 가중된 샘플들 $predLwip[x][y]$
- [0388] - 변환 블록 폭을 특정하는 변수 $nTbW$,
- [0389] - 변환 블록 높이를 특정하는 변수 $nTbH$,
- [0390] - 업샘플링 경계 폭을 특정하는 변수 $upsBdryW$,
- [0391] - 업샘플링 경계 높이를 특정하는 변수 $upsBdryH$,
- [0392] - $x = 0..upsBdryW - 1$ 인 상단 업샘플링 경계 샘플들 $upsBdryT[x]$,
- [0393] - $x = 0..upsBdryH - 1$ 인 좌측 업샘플링 경계 샘플들 $upsBdryL[x]$ 이다.
- [0394] 이러한 프로세스의 출력들은, $x = 0..nTbW - 1$, $y = 0..nTbH - 1$ 인 예측된 샘플들 $predSamples[x][y]$ 이다.
- [0395] 회소 예측된 샘플들 $predSamples[m][n]$ 은 다음과 같이 $predLwip[x][y]$ ($x = 0..predW - 1$, $y = 0..predH - 1$ 임)로부터 도출된다:
- [0396]
$$upHor = nTbW / predW \quad (8-X34)$$

- [0397] $upVer = nTbH / predH$ (8-X35)
- [0398] $predSamples[(x + 1) * upHor - 1][(y + 1) * upVer - 1] = predLwip[x][y]$ (8-X36)
- [0399] $x = 0..upsBdryW - 1$ 인 상단 경계 샘플들 $upsBdryT[x]$ 는 다음과 같이 $predSamples[m][-1]$ 에 할당된다:
- [0400] $predSamples[(x + 1) * (nTbW / upsBdryW) - 1][-1] = upsBdryT[x]$ (8-X37)
- [0401] $y = 0..upsBdryH - 1$ 인 좌측 경계 샘플들 $upsBdryL[y]$ 는 다음과 같이 $predSamples[-1][n]$ 에 할당된다:
- [0402] $predSamples[-1][(y + 1) * (nTbH / upsBdryH) - 1] = upsBdryL[y]$ (8-X38)
- [0403] $x = 0..nTbW - 1, y = 0..nTbH - 1$ 인 예측된 샘플들 $predSamples[x][y]$ 는 다음과 같이 도출된다:
- [0404] - $nTbH$ 가 $nTbW$ 보다 큰 경우에, 다음의 순서화된 단계들이 적용된다:
- [0405] 1. $upHor$ 이 1보다 클 때, 모든 회소 위치들 $(xHor, yHor) = (m * upHor - 1, n * upVer - 1)$ ($m = 0..predW - 1, n = 1..predH$ 임)에 대한 수평 업샘플링은 다음과 같이 $dX = 1..upHor - 1$ 로 적용된다:
- [0406] $predSamples[xHor + dX][yHor] = ((upHor - dX) * predSamples[xHor][yHor] + dX * predSamples[xHor + upHor][yHor]) / upHor$ (8-X39)
- [0407] 2. 모든 회소 위치들 $(xVer, yVer) = (m, n * upVer - 1)$ ($m = 0..nTbW - 1, n = 0..predH - 1$ 임)에 대한 수직 업샘플링은 다음과 같이 $dY = 1..upVer - 1$ 을 이용하여 적용된다:
- [0408] $predSamples[xVer][yVer + dY] = ((upVer - dY) * predSamples[xVer][yVer] + dY * predSamples[xVer][yVer + upVer]) / upVer$ (8-X40)
- [0409] - 그렇지 않은 경우, 다음의 순서화된 단계들이 적용된다:
- [0410] 1. $upVer$ 이 1보다 클 때, 모든 회소 위치들 $(xVer, yVer) = (m * upHor - 1, n * upVer - 1)$ ($m = 1..predW, n = 0..predH - 1$ 임)에 대한 수직 업샘플링이 (8-X40)에 특정된 바와 같이 $dY = 1..upVer - 1$ 을 이용하여 적용된다.
- [0411] 2. 모든 회소 위치들 $(xHor, yHor) = (m * upHor - 1, n)$ ($m = 0..predW - 1, n = 0..nTbH - 1$ 임)에 대한 수평 업샘플링은 (8-X39)에 특정된 바와 같이 $dX = 1..upHor - 1$ 을 이용하여 적용된다.
- [0412] <end>

[0413] 표 9-9 - 선택스 요소들 및 연관된 이진화들

선택스 구조	선택스 요소	이진화	
		프로세스	입력 파라미터들
coding_unit()	cu_skip_flag[][]	FL	cMax = 1
	pred_mode_ibc_flag	FL	cMax = 1
	pred_mode_flag	FL	cMax = 1
	<begin>intra_lwip_flag[][]	FL	cMax = 1
	intra_lwip_mpm_flag[][]	FL	cMax = 1
	intra_lwip_mpm_idx[][]	TR	cMax = 2, cRiceParam = 0
	intra_lwip_mpm_remainder[][]	FL	cMax = (cbWidth == 4 && cbHeight == 4) ? 31 : ((cbWidth <= 8 && cbHeight <= 8) ? 15 : 7)
	...		

[0414]

[0415] 표 9-15 - 컨텍스트 코딩된 bin들로 선택스 요소들에 대해 ctxInc의 할당

선택스 요소	binIdx					
	0	1	2	3	4	>= 5
...	종료	na	na	na	na	na
intra_lwip_flag[][]	(Abs(Log2(cbWidth) - Log2(cbHeight)) > 1) ? 3 : (0, 1, 2 (조항 9.5.4.2.2))	na	na	na	na	na
intra_lwip_mpm_flag[][]	0	na	na	na	na	na
intra_lwip_mpm_idx[][]	바이패스	바이패스	na	na	na	na
intra_lwip_mpm_remainder[][]	바이패스	바이패스	바이패스	바이패스	바이패스	na

[0416]

[0417] 표 9-16 - 좌측 및 상부 선택스 요소들을 이용한 ctxInc의 사양

선택스 요소	condL	condA	ctxSetIdx
...			
intra_lwip_flag[x0][y0]	intra_lwip_flag[xNbL][yNbL]	intra_lwip_flag[xNbA][yNbA]	0
...			

[0418]

[0419] <end>

[0420] ALWIP의 요약

[0421] 폭 W 및 높이 H의 직사각형 블록의 샘플들을 예측하기 위해, 아핀 선형 가중 인트라 예측(ALWIP)은 블록의 좌측에 있는 H개의 재구성된 이웃 경계 샘플의 하나의 라인 및 블록 상부의 W개의 재구성된 이웃 경계 샘플의 하나의 라인을 입력으로서 취한다. 재구성된 샘플들이 이용가능하지 않은 경우, 이들은 종래의 인트라 예측에서 행

해지는 바와 같이 생성된다. ALWIP는 루마 인트라 블록에만 적용된다. 크로마 인트라 블록의 경우, 종래의 인트라 코딩 모드들이 적용된다.

[0422] 예측 신호의 생성은 다음의 3개의 단계에 기초한다:

[0423] 1. 경계 샘플들 중에서, $W=H=4$ 의 경우에 4개의 샘플 및 모든 다른 경우에 8개의 샘플이 평균화에 의해 추출된다.

[0424] 2. 오프셋의 가산이 뒤따르는 행렬 벡터 곱셈은 입력으로서 평균화된 샘플들을 이용하여 수행된다. 그 결과는 원래의 블록에서의 샘플들의 서브샘플링된 세트에 대한 감소된 예측 신호이다.

[0425] 3. 나머지 위치들에서의 예측 신호는 각각의 방향에서의 단일 단계 선형 보간인 선형 보간에 의해 서브샘플링된 세트 상의 예측 신호로부터 생성된다.

[0426] ALWIP 모드가 적용되는 경우, ALWIP 모드의 인덱스 $predmode$ 는 3 MPMS를 갖는 MPM-리스트를 사용하여 시그널링된다. 여기서, MPM들의 도출은 다음과 같이 상부 및 좌측 PU의 인트라 모드들을 사용하여 수행된다. ALWIP 모드를 각각의 종래의 인트라 예측 모드 $predmode_{Angular}$ 에 할당하는 3개의 고정된 테이블 $map_angular_to_alwip_{idx}$, $idx \in \{0,1,2\}$ 가 존재하며,

[0427] $predmode_{ALWIP} = map_angular_to_alwip_{idx}[predmode_{Angular}]$ 이다.

[0428] 폭 W 및 높이 H 의 각각의 PU에 대해, 인덱스

[0429] $idx(PU) = idx(W, H) \in \{0,1,2\}$

[0430] 를 정의하고, 이는 ALWIP-파라미터들이 3세트 중 어느 것으로부터 취해져야 하는지를 표시한다.

[0431] 상부 예측 유닛 PU_{above} 이 이용가능하고, 현재 PU와 동일한 CTU에 속하며, 인트라 모드에 있는 경우, $idx(PU)=idx(PU_{above})$ 인 경우, 및 ALWIP가 ALWIP-모드 $predmode_{ALWIP}^{above}$ 로 PU_{above} 에 적용되는 경우,

[0432] $mode_{ALWIP}^{above} = predmode_{ALWIP}^{above}$ 로 표현한다.

[0433] 상부 PU가 이용가능하고, 현재 PU와 동일한 CTU에 속하고 인트라 모드에 있는 경우, 그리고 종래의 인트라 예측 모드 $predmode_{Angular}^{above}$ 가 상부 PU에 적용되는 경우,

[0434] $mode_{ALWIP}^{above} = map_angular_to_alwip_{idx(PU_{above})}[predmode_{Angular}^{above}]$ 로 표현한다.

[0435] 모든 다른 경우들에서,

[0436] $mode_{ALWIP}^{above} = -1$ 로 표현하고,

[0437] 이는 이 모드가 이용가능하지 않다는 것을 의미한다. 동일한 방식으로 그러나 좌측 PU가 현재 PU와 동일한 CTU에 속할 필요가 있다는 제한 없이, 모드 $mode_{ALWIP}^{left}$ 를 도출한다.

[0438] 마지막으로, 3개의 고정된 디폴트 리스트 $list_{idx}$, $idx \in \{0,1,2\}$ 가 제공되고, 이들 각각은 3개의 별개의 ALWIP 모드를 포함한다. 디폴트 리스트 $list_{idx(PU)}$ 및 모드들 $mode_{ALWIP}^{above}$ 및 $mode_{ALWIP}^{left}$ 중에서, 반복들을 제거할 뿐만 아니라 디폴트 값들에 의해 -1을 치환함으로써 3개의 별개의 MPM을 구성한다.

[0439] 루마 MPM-리스트 도출을 위해, ALWIP-모드 $predmode_{ALWIP}$ 를 사용하는 이웃 루마 블록이 직면될 때마다, 이 블록은 종래의 인트라-예측 모드 $predmode_{Angular}$ 을 사용하고 있었던 것처럼 취급된다.

[0440] $predmode_{Angular} = map_alwip_to_angular_{idx(PU)}[predmode_{ALWIP}]$

[0441] 3 VVC에서의 변환

- [0442] 3.1 다중 변환 선택(MTS)
- [0443] HEVC에서 채용된 DCT-II에 더하여, 인터 및 인트라 코딩된 블록들 둘 다를 잔차 코딩하기 위해 다중 변환 선택(MTS) 스킴이 사용된다. 그것은 DCT8/DST7로부터의 다중 선택된 변환들을 사용한다. 새롭게 도입된 변환 행렬들은 DST-VII 및 DCT-VIII이다.
- [0444] 3.2 JVET-N0193에서 제안된 감소된 2차 변환(RST)
- [0445] 감소된 2차 변환(RST)은 4x4 및 8x8 블록들에 대해 각각 16x16 및 16x64 비분리형 변환을 적용한다. 1차 순방향 및 역방향 변환들은 2개의 1-D 수평/수직 변환 패스들과 동일한 방식으로 여전히 수행된다. 2차 순방향 및 역방향 변환은 1차 변환들과는 별개의 프로세스 단계이다. 인코더에 대해, 1차 순방향 변환이 먼저 수행되고, 그 후 2차 순방향 변환 및 양자화, 및 CABAC 비트 인코딩이 후속된다. 디코더에 대해, CABAC 비트 디코딩 및 역양자화, 그 후 2차 역변환이 먼저 수행되고, 그 후 1차 역변환이 후속된다. RST는 인트라 슬라이스 및 인터 슬라이스들 둘 다에서 인트라 코딩된 TU들에만 적용된다.
- [0446] 3.3 JVET-N0185에서의 인트라 모드 코딩을 위한 통합된 MPM 리스트
- [0447] 통합된 6-MPM 리스트는 다중 참조 라인(MRL) 및 인트라 서브-파티션(ISP) 코딩 툴들이 적용되는지 여부에 관계없이 인트라 블록들에 대해 제안된다. MPM 리스트는 VTM4.0에서와 같이 좌측 및 상부 이웃 블록의 인트라 모드들에 기초하여 구성된다. 좌측의 모드가 Left로 표시되고 상부 블록의 모드가 Above로 표시된다고 가정하면, 통합된 MPM 리스트는 다음과 같이 구성된다:
- [0448] ● 이웃 블록이 이용가능하지 않을 때, 그 인트라 모드는 디폴트로 Planar(평면)으로 설정된다.
- [0449] ● Left 및 Above 모드들 둘 다가 비-각도 모드들인 경우:
- [0450] a. MPM 리스트 $\rightarrow \{\text{Planar}, \text{DC}, \text{V}, \text{H}, \text{V}-4, \text{V}+4\}$
- [0451] ● 모드들 Left 및 Above 중 하나가 각도 모드이고, 다른 하나가 비-각도인 경우:
- [0452] a. 모드 Max를 Left 및 Above에서 더 큰 모드로서 설정한다.
- [0453] b. MPM 리스트 $\rightarrow \{\text{Planar}, \text{Max}, \text{DC}, \text{Max}-1, \text{Max}+1, \text{Max}-2\}$
- [0454] ● Left 및 Above가 둘 다 각도이고 이들이 상이한 경우:
- [0455] a. 모드 Max를 Left 및 Above에서 더 큰 모드로서 설정한다.
- [0456] b. 모드 Left와 Above의 차이가 2 내지 62(경계 포함)의 범위에 있는 경우,
- [0457] i. MPM 리스트 $\rightarrow \{\text{Planar}, \text{Left}, \text{Above}, \text{DC}, \text{Max}-1, \text{Max}+1\}$
- [0458] c. 그렇지 않으면
- [0459] i. MPM 리스트 $\rightarrow \{\text{Planar}, \text{Left}, \text{Above}, \text{DC}, \text{Max}-2, \text{Max}+2\}$
- [0460] ● Left 및 Above가 둘 다 각도이고 이들이 동일한 경우:
- [0461] a. MPM 리스트 $\rightarrow \{\text{Planar}, \text{Left}, \text{Left}-1, \text{Left}+1, \text{DC}, \text{Left}-2\}$
- [0462] 게다가, MPM 인덱스 코드워드의 제1 빈은 CABAC 컨텍스트 코딩된다. 현재 인트라 블록이 MRL 인에이블형(enabled)인지, ISP 인에이블형인지, 또는 정상(normal) 인트라 블록인지에 대응하는 총 3개의 컨텍스트가 사용된다.
- [0463] 통합된 MPM 리스트 구성에서 사용되는 좌측 이웃 블록 및 상부 이웃 블록은 도 10에 도시된 바와 같이 A2 및 B2이다.
- [0464] 하나의 MPM 플래그가 먼저 코딩된다. 블록이 MPM 리스트에서 모드 중 하나로 코딩되는 경우에, MPM 인덱스가 추가로 코딩된다. 그렇지 않으면, (MPM들을 제외한) 나머지 모드들에 대한 인덱스가 코딩된다.
- [0465] 4 기존의 구현들에서의 단점들의 예들
- [0466] JVET-N0217에서의 ALWIP의 설계는 다음의 문제점들을 갖는다:
- [0467] 1) 2019년 3월 JVET 회의에서, 통합된 6-MPM 리스트 생성이 MRL 모드, ISP 모드, 및 정상 인트라 모드(normal

intra mode)에 대해 채택되었다. 그러나, 아핀 선형 가중 예측 모드는 MPM 리스트 구성을 복잡하게 하는 상이한 3-MPM 리스트 구성을 사용한다. 복잡한 MPM 리스트 구성은, 특히 4x4 샘플들과 같은 작은 블록들에 대해, 디코더의 스루풋을 손상시킬 수 있다.

2) ALWIP는 블록의 루마 컴포넌트에만 적용된다. ALWP 코딩된 블록의 크로마 컴포넌트의 경우, 크로마 모드 인덱스가 코딩되고 디코더로 전송되며, 이는 불필요한 시그널링을 초래할 수 있었다.

3) ALWIP와 다른 코딩 톨의 상호작용이 고려되어야 한다.

$$\text{upsBdryX}[x] = (\sum_{i=0}^{\text{uDwn}-1} \text{refX}[x * \text{uDwn} + 4i] + (1 \ll (\text{Log2}(\text{uDwn}) - 1))) \gg \text{Log2}(\text{uDwn}) \quad (8\text{-X31})$$
에서의 upsBdryX를 계산할 때, $\text{Log2}(\text{uDwn}) - 1$ 은 -1과 동일한 한편, -1로 좌측 시프트되는 것은 정의되지 않는 것이 가능하다.

5) 예측 샘플들을 업샘플링할 때, 라운딩이 적용되지 않는다.

6) 더블록킹 프로세스에서, ALWIP 코딩된 블록들은 정상 인트라-블록들로서 취급된다.

5 행렬-기반 인트라 코딩을 위한 예시적인 방법들

현재 개시된 기술의 실시예들은 기존의 구현들의 단점들을 극복하고, 그에 의해 더 높은 코딩 효율들을 갖지만 더 낮은 계산 복잡도를 갖는 비디오 코딩을 제공한다. 본 문서에 설명된 바와 같은, 비디오 코딩을 위한 행렬 기반 인트라 예측 방법들은 기존 및 장래의 비디오 코딩 표준들 둘 다를 향상시킬 수 있고, 다양한 구현들에 대해 설명된 다음의 예들에서 설명된다. 아래에 제공되는 개시된 기술의 예들은 일반적인 개념들을 설명하고, 제한적인 것으로 해석되도록 의도되지 않는다. 일 예에서, 명시적으로 반대로 표시되지 않는 한, 이러한 예들에서 설명된 다양한 특징들이 조합될 수 있다.

이하의 논의에서, 인트라 예측 모드는 각도 인트라 예측 모드(DC, 평면, CCLM 및 다른 가능한 인트라 예측 모드들을 포함함)를 지칭한다; 한편 인트라 모드는 정상 인트라 모드, 또는 MRL, 또는 ISP 또는 ALWIP를 지칭한다.

다음의 논의에서, "다른 인트라 모드들"은 정상 인트라 모드, 또는 MRL, 또는 ISP와 같은 ALWIP를 제외한 하나 또는 다수의 인트라 모드를 지칭할 수 있다.

이하의 논의에서, $\text{SatShift}(x, n)$ 는 다음과 같이 정의된다:

$$\text{SatShift}(x, n) = \begin{cases} (x + \text{offset0}) \gg n & x \geq 0 \text{ 인 경우} \\ -((-x + \text{offset1}) \gg n) & x < 0 \text{ 인 경우} \end{cases}$$

$\text{Shift}(x, n)$ 는 $\text{Shift}(x, n) = (x + \text{offset0}) \gg n$ 으로서 정의된다.

일 예에서, offset0 및/또는 offset1은 $(1 \ll n) \gg 1$ 또는 $(1 \ll (n-1))$ 로 설정된다. 다른 예에서, offset0 및/또는 offset1은 0으로 설정된다.

다른 예에서, $\text{offset0} = \text{offset1} = ((1 \ll n) \gg 1) - 1$ 또는 $((1 \ll (n-1))) - 1$ 이다.

$\text{Clip3}(\text{min}, \text{max}, x)$ 은 다음과 같이 정의된다:

$$\text{Clip3}(\text{Min}, \text{Max}, x) = \begin{cases} \text{Min} & x < \text{Min} \text{ 인 경우} \\ \text{Max} & x > \text{Max} \text{ 인 경우} \\ x & \text{그 외} \end{cases}$$

ALWIP에 대한 MPM 리스트 구성

1. ALWIP에 대한 MPM 리스트의 전체 또는 부분이 (정상 인트라 모드, MRL, 또는 ISP와 같은) 비-ALWIP 인트라 모드에 대한 MPM 리스트를 구성하기 위해 전체 또는 부분 절차에 따라 구성될 수 있는 것이 제안된다.

a. 일 예에서, ALWIP에 대한 MPM 리스트의 크기는 비-ALWIP 인트라 모드에 대한 MPM 리스트의 크기와 동일할 수 있다.

i. 예를 들어, MPM 리스트의 크기는 ALWIP 및 비-ALWIP 인트라 모드들 둘 다에 대해 6이

다.

[0488] b. 일 예에서, ALWIP에 대한 MPM 리스트는 비-ALWIP 인트라 모드에 대한 MPM 리스트로부터 도출될 수 있다.

[0489] i. 일 예에서, 비-ALWIP 인트라 모드에 대한 MPM 리스트가 먼저 구성될 수 있다. 그 후, 이들 중 일부 또는 전부는, ALWIP 코딩된 블록들에 대한 MPM 리스트에 더 추가될 수 있는 MPM들로 전환될 수 있다.

[0490] 1) 대안적으로, 또한, 전환된 MPM을 ALWIP 코딩된 블록들에 대한 MPM 리스트에 추가할 때, 프루닝(pruning)이 적용될 수 있다.

[0491] 2) 디폴트 모드들은 ALWIP 코딩된 블록들에 대한 MPM 리스트에 추가될 수 있다.

[0492] a. 일 예에서, 디폴트 모드들은 비-ALWIP 인트라 모드의 MPM 리스트로부터 전환된 것들 전에 추가될 수 있다.

[0493] b. 대안적으로, 디폴트 모드들은 비-ALWIP 인트라 모드의 MPM 리스트로부터 전환된 것들 이후에 추가될 수 있다.

[0494] c. 대안적으로, 디폴트 모드들은 비-ALWIP 인트라 모드의 MPM 리스트로부터 전환된 것들과 인터리빙된 방식으로 추가될 수 있다.

[0495] d. 일 예에서, 디폴트 모드들은 모든 종류의 블록들에 대해 동일하도록 고정될 수 있다.

[0496] e. 대안적으로, 디폴트 모드들은 이웃 블록들의 이용가능성, 이웃 블록들의 모드 정보, 블록 치수와 같은 코딩된 정보에 따라 결정될 수 있다.

[0497] ii. 일 예에서, 비-ALWIP 인트라 모드에 대한 MPM 리스트 내의 하나의 인트라-예측 모드는, 그것이 ALWIP에 대한 MPM 리스트에 넣어질 때, 그 대응하는 ALWIP 인트라-예측 모드로 전환될 수 있다.

[0498] 1) 대안적으로, 비-ALWIP 인트라 모드들에 대한 MPM 리스트 내의 모든 인트라-예측 모드들은 ALWIP에 대한 MPM 리스트를 구성하기 위해 사용되기 전에 대응하는 ALWIP 인트라-예측 모드들로 전환될 수 있다.

[0499] 2) 대안적으로, 모든 후보 인트라-예측 모드들(이웃 블록들로부터의 인트라-예측 모드들 및 Planar 및 DC와 같은 디폴트 인트라-예측 모드들을 포함할 수 있음)은, 비-ALWIP 인트라 모드들에 대한 MPM 리스트가 ALWIP에 대한 MPM 리스트를 도출하기 위해 추가로 사용될 수 있는 경우, 비-ALWIP 인트라 모드들에 대한 MPM 리스트를 구성하기 위해 사용되기 전에 대응하는 ALWIP 인트라-예측 모드들로 전환될 수 있다.

[0500] 3) 일 예에서, 2개의 전환된 ALWIP 인트라 예측 모드가 비교될 수 있다.

[0501] a. 일 예에서, 이들이 동일한 경우에, 이들 중 하나만이 ALWIP에 대한 MPM 리스트에 넣어질 수 있다.

[0502] b. 일 예에서, 이들이 동일한 경우에, 이들 중 하나만이 비-ALWIP 인트라 모드들에 대한 MPM 리스트에 넣어질 수 있다.

[0503] iii. 일 예에서, 비-ALWIP 인트라 모드들에 대한 MPM 리스트 내의 S개의 인트라 예측 모드 중 K개가 ALWIP 모드에 대한 MPM 리스트로서 선택될 수 있다. 예를 들어, K는 3과 동일하고 S는 6과 동일하다.

[0504] 1) 일 예에서, 비-ALWIP 인트라 모드들에 대한 MPM 리스트 내의 제1 K개의 인트라 예측 모드는 ALWIP 모드에 대한 MPM 리스트로서 선택될 수 있다.

[0505] 2. ALWIP에 대한 MPM 리스트를 도출하기 위해 사용되는 하나 또는 다수의 이웃 블록은 또한 (정상 인트라 모드, MRL, 또는 ISP와 같은) 비-ALWIP 인트라 모드들에 대한 MPM 리스트를 도출하기 위해 사용될 수 있는 것이 제안된다.

[0506] a. 일 예에서, ALWIP에 대한 MPM 리스트를 도출하기 위해 사용되는 현재 블록에 남겨진 이웃 블록은 비-ALWIP 인트라 모드들에 대한 MPM 리스트를 도출하기 위해 사용되는 것과 동일해야 한다.

- [0507] i. 현재 블록의 상단 좌측 코너가 (x_{Cb}, y_{Cb}) 이고, 현재 블록의 폭 및 높이가 W 및 H 라고 가정하면, 일 예에서, ALWIP 및 비-ALWIP 인트라 모드들 둘 다에 대한 MPM 리스트를 도출하기 위해 사용되는 좌측 이웃 블록은 위치 $(x_{Cb}-1, y_{Cb})$ 를 커버할 수 있다. 대안적인 예에서, ALWIP 및 비-ALWIP 인트라 모드들 둘 다에 대한 MPM 리스트를 도출하기 위해 사용되는 좌측 이웃 블록은 위치 $(x_{Cb}-1, y_{Cb}+H-1)$ 를 커버할 수 있다.
- [0508] ii. 예를 들어, 통합된 MPM 리스트 구성에서 사용되는 좌측 이웃 블록 및 상부 이웃 블록은 도 10에 도시된 바와 같이 A2 및 B2이다.
- [0509] b. 일 예에서, ALWIP에 대한 MPM 리스트를 도출하기 위해 사용되는 현재 블록 상부의 이웃 블록은 비-ALWIP 인트라 모드들에 대한 MPM 리스트를 도출하기 위해 사용되는 것과 동일해야 한다.
- [0510] i. 현재 블록의 상단 좌측 코너가 (x_{Cb}, y_{Cb}) 이고, 현재 블록의 폭 및 높이가 W 및 H 라고 가정하면, 일 예에서, ALWIP 및 비-ALWIP 인트라 모드들 둘 다에 대한 MPM 리스트를 도출하기 위해 사용되는 상부 이웃 블록은 위치 $(x_{Cb}, y_{Cb}-1)$ 를 커버할 수 있다. 대안적인 예에서, ALWIP 및 비-ALWIP 인트라 모드들 둘 다에 대한 MPM 리스트를 도출하기 위해 사용되는 상부 이웃 블록은 위치 $(x_{Cb}+W-1, y_{Cb}-1)$ 를 커버할 수 있다.
- [0511] ii. 예를 들어, 통합된 MPM 리스트 구성에서 사용되는 좌측 이웃 블록 및 상부 이웃 블록은 도 10에 도시된 바와 같이 A1 및 B1이다.
- [0512] 3. ALWIP에 대한 MPM 리스트는 현재 블록의 폭 및/또는 높이에 따라 상이한 방식으로 구성될 수 있는 것이 제안된다.
- [0513] a. 일 예에서, 상이한 블록 치수들에 대해 상이한 이웃 블록들이 액세스될 수 있다.
- [0514] 4. ALWIP에 대한 MPM 리스트 및 비-ALWIP 인트라 모드들에 대한 MPM 리스트는 동일한 절차로 그러나 상이한 파라미터들로 구성될 수 있는 것이 제안된다.
- [0515] a. 일 예에서, 비-ALWIP 인트라 모드들의 MPM 리스트 구성 절차에서 S 개의 인트라 예측 모드 중 K 개는 ALWIP 모드에서 사용되는 MPM 리스트에 대해 도출될 수 있다. 예를 들어, K 는 3과 동일하고 S 는 6과 동일하다.
- [0516] i. 일 예에서, MPM 리스트 구성 절차에서의 제1 K 개의 인트라 예측 모드는 ALWIP 모드에서 사용되는 MPM 리스트에 대해 도출될 수 있다.
- [0517] b. 일 예에서, MPM 리스트 내의 제1 모드는 상이할 수 있다.
- [0518] i. 예를 들어, 비-ALWIP 인트라 모드들에 대한 MPM 리스트 내의 제1 모드는 Planar일 수 있지만, 그것은 ALWIP에 대한 MPM 리스트 내의 모드 $X0$ 일 수 있다.
- [0519] 1) 일 예에서, $X0$ 은 Planar로부터 전환된 ALWIP 인트라 예측 모드일 수 있다.
- [0520] c. 일 예에서, MPM 리스트 내의 스템핑(stuffing) 모드들은 상이할 수 있다.
- [0521] i. 예를 들어, 비-ALWIP 인트라 모드들에 대한 MPM 리스트 내의 처음 3개의 스템핑 모드들은 DC, Vertical(수직) 및 (Horizontal)수평일 수 있지만, 이들은 ALWIP에 대한 MPM 리스트 내의 모드 $X1$, $X2$, $X3$ 일 수 있다.
- [0522] 1) 일 예에서, $X1$, $X2$, $X3$ 은 상이한 $sizeId$ 에 대해 상이할 수 있다.
- [0523] ii. 일 예에서, 스템핑 모드의 수는 상이할 수 있다.
- [0524] d. 일 예에서, MPM 리스트 내의 이웃 모드들은 상이할 수 있다.
- [0525] i. 예를 들어, 이웃 블록들의 정상 인트라 예측 모드들은 비-ALWIP 인트라 모드들에 대한 MPM 리스트를 구성하기 위해 사용된다. 그리고 이들은 ALWIP 인트라-예측 모드들로 전환되어 ALWIP 모드에 대한 MPM 리스트를 구성한다.
- [0526] e. 일 예에서, MPM 리스트 내의 시프트된 모드들은 상이할 수 있다.
- [0527] i. 예를 들어, $X+K0$ (여기서, X 가 정상 인트라 예측 모드이고 $K0$ 이 정수임)이 비-ALWIP 인트라 모드들에 대한 MPM 리스트에 넣어질 수 있다. 그리고 $Y+K1$ (여기서, Y 는 ALWIP 인트라 예측 모드이고, $K1$ 은 정수임)은 ALWIP에 대한 MPM 리스트에 넣어질 수 있고, 여기서 $K0$ 은 $K1$ 과 상이할 수 있다.

- [0528] 1) 일 예에서, K1은 폭 및 높이에 의존할 수 있다.
- [0529] 5. 이웃 블록은 비-ALWIP 인트라 모드들로 현재 블록에 대한 MPM 리스트를 구성할 때 이웃 블록이 ALWIP로 코딩되는 경우 이용불가능한 것으로 취급되는 것이 제안된다.
- [0530] a. 대안적으로, 이웃 블록은 비-ALWIP 인트라 모드들로 현재 블록에 대한 MPM 리스트를 구성할 때 이웃 블록이 ALWIP로 코딩되는 경우에 (Planar과 같은) 미리 정의된 인트라-예측 모드로 코딩되는 것으로서 취급된다.
- [0531] 6. 이웃 블록은 ALWIP 모드로 현재 블록에 대한 MPM 리스트를 구성할 때 이웃 블록이 비-ALWIP 인트라 모드들로 코딩되는 경우 이용불가능한 것으로 취급되는 것이 제안된다.
- [0532] a. 대안적으로, 이웃 블록은 ALWIP 모드로 현재 블록에 대한 MPM 리스트를 구성할 때 이웃 블록이 비-ALWIP 인트라 모드들로 코딩되는 경우에 미리 정의된 ALWIP 인트라 예측 모드 X로 코딩되는 것으로서 취급된다.
- [0533] i. 일 예에서, X는 폭 및/또는 높이와 같은 블록 치수들에 의존할 수 있다.
- [0534] 7. 라인 버퍼로부터 ALWIP 플래그의 저장을 제거하는 것이 제안된다.
- [0535] a. 일 예에서, 액세스될 제2 블록이 현재 블록에 비해 상이한 LCU/CTU 행/영역에 위치할 때, 제2 블록이 ALWIP로 코딩되는지의 조건부 검사는 스킵된다.
- [0536] b. 일 예에서, 액세스될 제2 블록이 현재 블록에 비해 상이한 LCU/CTU 행/영역에 위치할 때, 제2 블록은 정상 인트라 코딩된 블록으로서 취급되는 것과 같은, 비-ALWIP 모드와 동일한 방식으로 취급된다.
- [0537] 8. ALWIP 플래그를 인코딩할 때, $K(K \geq 0)$ 개 이하의 컨텍스트가 사용될 수 있다.
- [0538] a. 일 예에서, $K=1$ 이다.
- [0539] 9. ALWIP 모드와 연관된 모드 인덱스를 직접 저장하는 대신에 ALWIP 코딩된 블록들의 전환된 인트라 예측 모드를 저장하는 것이 제안된다.
- [0540] a. 일 예에서, 하나의 ALWIP 코딩된 블록과 연관된 디코딩된 모드 인덱스는, 섹션 2.5.7에서 설명된 바와 같은 `map_alwip_to_angular`에 따른 것과 같이, 정상 인트라 모드에 매핑된다.
- [0541] b. 대안적으로, 또한, ALWIP 플래그의 저장은 완전히 제거된다.
- [0542] c. 대안적으로, 또한, ALWIP 모드의 저장은 완전히 제거된다.
- [0543] d. 대안적으로, 또한, 하나의 이웃/현재 블록이 ALWIP 플래그로 코딩되는지의 조건 검사는 스킵될 수 있다.
- [0544] e. 대안적으로, 또한, 하나의 액세스된 블록과 연관된 정상 인트라 예측들 및 ALWIP 코딩된 블록들에 대해 할당된 모드들의 전환이 스킵될 수 있다.
- [0545] 상이한 컬러 컴포넌트들에 대한 ALWIP
- [0546] 10. 대응하는 루마 블록이 ALWIP 모드로 코딩되는 경우, 추론된 크로마 인트라 모드(예를 들어, DM 모드)가 항상 적용될 수 있는 것이 제안된다.
- [0547] a. 일 예에서, 크로마 인트라 모드는 대응하는 루마 블록이 ALWIP 모드로 코딩되는 경우에 시그널링 없이 DM 모드인 것으로 추론된다.
- [0548] b. 일 예에서, 대응하는 루마 블록은 주어진 위치(예를 들어, 현재 크로마 블록의 상단 좌측, 현재 크로마 블록의 중심)에 위치한 크로마 샘플의 대응하는 샘플을 커버하는 것일 수 있다.
- [0549] c. 일 예에서, DM 모드는 (ALWIP) 모드를 정상 인트라 모드 중 하나에 매핑하는 것 등을 통해, 대응하는 루마 블록의 인트라 예측 모드에 따라 도출될 수 있다.
- [0550] 11. 크로마 블록들의 대응하는 루마 블록이 ALWIP 모드로 코딩될 때, 몇몇 DM 모드들이 도출될 수 있다.
- [0551] 12. 하나의 대응하는 루마 블록이 ALWIP 모드로 코딩되는 경우 크로마 블록들에 특수 모드가 할당되는 것이 제안된다.

- [0552] a. 일 예에서, 특수 모드는 ALWIP 코딩된 블록들과 연관된 인트라 예측 모드에 관계없이 주어진 정상 인트라 예측 모드로 정의된다.
- [0553] b. 일 예에서, 인트라 예측의 상이한 방식들이 이 특수 모드에 할당될 수 있다.
- [0554] 13. ALWIP가 또한 크로마 컴포넌트들에 적용될 수 있는 것이 제안된다.
- [0555] a. 일 예에서, 행렬 및/또는 바이어스 벡터는 상이한 컬러 컴포넌트들에 대해 상이할 수 있다.
- [0556] b. 일 예에서, 행렬 및/또는 바이어스 벡터는 Cb 및 Cr에 대해 공동으로 미리 정의될 수 있다.
- [0557] i. 일 예에서, Cb 및 Cr 컴포넌트는 연결될 수 있다.
- [0558] ii. 일 예에서, Cb 및 Cr 컴포넌트는 인터리빙될 수 있다.
- [0559] c. 일 예에서, 크로마 컴포넌트는 대응하는 루마 블록과 동일한 ALWIP 인트라 예측 모드를 공유할 수 있다.
- [0560] i. 일 예에서, 대응하는 루마 블록이 ALWIP 모드를 적용하고 크로마 블록이 DM 모드로 코딩되는 경우, 동일한 ALWIP 인트라 예측 모드가 크로마 컴포넌트에 적용된다.
- [0561] ii. 일 예에서, 동일한 ALWIP 인트라-예측 모드가 크로마 컴포넌트에 적용되고 그 후 선형 보간이 스킵될 수 있다.
- [0562] iii. 일 예에서, 동일한 ALWIP 인트라 예측 모드가 서브샘플링된 행렬 및/또는 바이어스 벡터로 크로마 컴포넌트에 적용된다.
- [0563] d. 일 예에서, 상이한 컴포넌트에 대한 ALWIP 인트라 예측 모드들의 수는 상이할 수 있다.
- [0564] i. 예를 들어, 크로마 컴포넌트들에 대한 ALWIP 인트라 예측 모드들의 수는 동일한 블록 폭 및 높이에 대한 루마 컴포넌트에 대한 것보다 적을 수 있다.
- [0565] ALWIP의 적용가능성
- [0566] 14. ALWIP가 적용될 수 있는지가 시그널링될 수 있는 것이 제안된다.
- [0567] a. 예를 들어, 그것은 시퀀스 레벨에서(예를 들어, SPS에서), 픽처 레벨에서(예를 들어, PPS 또는 픽처 헤더에서), 슬라이스 레벨에서(예를 들어, 슬라이스 헤더에서), 타일 그룹 레벨에서(예를 들어, 타일 그룹 헤더에서), 타일 레벨에서, CTU 행 레벨에서, 또는 CTU 레벨에서 시그널링될 수 있다.
- [0568] b. 예를 들어, `intra_lwip_flag`는 ALWIP가 적용될 수 없는 경우 시그널링되지 않고 0인 것으로 추론될 수 있다.
- [0569] 15. ALWIP가 적용될 수 있는지가 블록 폭(W) 및/또는 높이(H)에 의존할 수 있는 것이 제안된다.
- [0570] c. 예를 들어, ALWIP는 $W \geq T1$ (또는 $W > T1$)이고 $H \geq T2$ (또는 $H > T2$)인 경우 적용되지 않을 수 있다. 예를 들어, $T1=T2=32$;
- [0571] i. 예를 들어, ALWIP는 $W \leq T1$ (또는 $W < T1$)이고 $H \leq T2$ (또는 $H < T2$)인 경우 적용되지 않을 수 있다. 예를 들어, $T1=T2=32$;
- [0572] d. 예를 들어, ALWIP는 $W \geq T1$ (또는 $W > T1$)이거나 $H \geq T2$ (또는 $H > T2$)인 경우 적용되지 않을 수 있다. 예를 들어, $T1=T2=32$;
- [0573] i. 예를 들어, ALWIP는 $W \leq T1$ (또는 $W < T1$)이거나 $H \leq T2$ (또는 $H < T2$)인 경우 적용되지 않을 수 있다. 예를 들어, $T1=T2=32$;
- [0574] e. 예를 들어, ALWIP는 $W + H \geq T$ (또는 $W * H \geq T$)인 경우 적용되지 않을 수 있다. 예를 들어, $T=256$;
- [0575] i. 예를 들어, ALWIP는 $W + H \leq T$ (또는 $W + H < T$)인 경우 적용되지 않을 수 있다. 예를 들어, $T=256$;
- [0576] f. 예를 들어, ALWIP는 $W * H \geq T$ (또는 $W * H > T$)인 경우 적용되지 않을 수 있다. 예를 들어,

T=256;

- [0577] i. 예를 들어, ALWIP는 $W * H \leq T$ (또는 $W * H < T$)인 경우 적용되지 않을 수 있다. 예를 들어, T=256;
- [0578] g. 예를 들어, intra_lwip_flag는 ALWIP가 적용될 수 없는 경우 시그널링되지 않고 0인 것으로 추론될 수 있다.
- [0579] ALWIP에서의 계산 문제들
- [0580] 16. ALWIP에 수반되는 임의의 시프트 연산은 단지 S만큼 좌측 시프트 또는 우측 시프트만 할 수 있는 것으로 제안되며, 여기서 S는 0 이상이어야 한다.
- [0581] a. 일 예에서, 우측 시프트 연산은 S가 0과 동일하거나 0보다 클 때 상이할 수 있다.
- [0582] i. 일 예에서, upsBdryX[x]는
- [0583] $uDwn > 1$ 일 때 $upsBdryX[x] = (\sum_{i=0}^{uDwn-1} refX[x * uDwn + i] + (1 \ll (\text{Log2}(uDwn) - 1))) \gg \text{Log2}(uDwn)$ 와 같이 계산되어야 하고, uDwn이 1과 동일할 때 $upsBdryX[x] = \sum_{i=0}^{uDwn-1} refX[x * uDwn + i]$ 와 같이 계산되어야 한다.
- [0584] b. 일 예에서, upsBdryX[x]는
- [0585] $upsBdryX[x] = (\sum_{i=0}^{uDwn-1} refX[x * uDwn + i] + (1 \ll \text{Log2}(uDwn) \gg 1)) \gg \text{Log2}(uDwn)$ 과 같이 계산되어야 한다.
- [0586] 17. ALWIP의 업샘플링 프로세스에서 결과들이 제로 쪽으로 또는 제로로부터 멀어지게 라운딩(round)되어야 한다는 것이 제안된다.
- [0587] a. 일 예에서,
- [0588] $predSamples[xHor + dX][yHor] = ((upHor - dX) * predSamples[xHor][yHor] + dX * predSamples[xHor + upHor][yHor] + offsetHor) / upHor$ (8-X39)
- [0589] 및
- [0590] $predSamples[xVer][yVer + dY] = ((upVer - dY) * predSamples[xVer][yVer] + dY * predSamples[xVer][yVer + upVer] + offsetVer) / upVer$ (8-X40)
- [0591] 여기서 offsetHor 및 offsetVer은 정수들이다. 예를 들어, offsetHor = upHor/2 및 offsetVer=upVer/2이다.
- [0592] 다른 코딩 툴들과의 상호작용
- [0593] 18. ALWIP가 CIIP-코딩된 블록에 사용될 수 있는 것이 제안된다.
- [0594] a. 일 예에서, CIIP-코딩된 블록에서, 인트라 예측 신호를 생성하기 위해 ALWIP 인트라-예측 모드가 사용되는지 또는 Planar과 같은 정상 인트라 예측 모드(normal intra prediction mode)가 사용되는지가 명시적으로 시그널링될 수 있다.
- [0595] b. 일 예에서, 인트라 예측 신호를 생성하기 위해 ALWIP 인트라 예측 모드가 사용될 수 있는지 또는 Planar과 같은 정상 인트라 예측 모드가 사용될 수 있는지가 암시적으로 추론될 수 있다.
- [0596] i. 일 예에서, ALWIP 인트라-예측 모드는 CIIP 코딩된 블록에서 결코 사용되지 않을 수 있다.
- [0597] 1) 대안적으로, 정상 인트라 예측은 CIIP 코딩된 블록에서 결코 사용되지 않을 수 있다.
- [0598] ii. 일 예에서, 인트라 예측 신호를 생성하기 위해 ALWIP 인트라 예측 모드가 사용되는지 또는 Planar과 같은 정상 인트라 예측 모드가 사용되는지가 이웃 블록들의 정보로부터 추론될 수 있다.
- [0599] 19. CCLM 모드에서 이웃 루마 샘플들을 다운샘플링하기 위해 사용되는 절차의 전체 또는 부분이 ALWIP 모드

에서 이웃 샘플들을 다운샘플링하기 위해 사용될 수 있는 것이 제안된다.

[0600] a. 대안적으로, ALWIP 모드에서 이웃 루마 샘플들을 다운샘플링하기 위해 사용되는 절차의 전체 또는 부분이 CCLM 모드에서 이웃 샘플들을 다운샘플링하기 위해 사용될 수 있다.

[0601] b. 다운-샘플링 절차는 CCLM 프로세스 및 ALWIP 프로세스에서 사용될 때 상이한 파라미터들/인수들로 호출될 수 있다.

[0602] c. 일 예에서, CCLM 프로세스에서의 (이웃하는 루마 위치들, 다운샘플링 필터들의 선택과 같은) 다운샘플링 방법은 ALWIP 프로세스에서 활용될 수 있다.

[0603] d. 이웃하는 루마 샘플들을 다운샘플링하기 위해 사용되는 절차는 다운샘플링된 위치들, 다운샘플링 필터들, 라운딩 및 클리핑 연산들의 선택을 적어도 포함한다.

[0604] 20. ALWIP 모드로 코딩된 블록이 RST 또는/및 2차 변환 또는/및 회전 변환 또는/및 비분리형 2차 변환(NSST)을 적용할 수 없는 것이 제안된다.

[0605] a. 일 예에서, 이러한 제약이 적용될 수 있는지의 여부는, 예를 들어, (15)에서 설명된 조건들과 동일하게, 블록의 치수 정보에 의존할 수 있다.

[0606] b. 대안적으로, ALWIP 모드는 RST 또는/및 2차 변환 또는/및 회전 변환 또는/및 NSST가 적용될 때 허용되지 않을 수 있다.

[0607] c. 대안적으로, ALWIP 모드로 코딩된 블록은 RST 또는/및 2차 변환 또는/및 회전 변환 또는/및 비분리형 2차 변환(NSST)을 적용할 수 있다.

[0608] i. 일 예에서, 변환 행렬의 선택은 ALWIP 인트라 예측 모드에 의존할 수 있다.

[0609] ii. 일 예에서, 변환 행렬의 선택은 ALWIP 인트라 예측 모드로부터 전환되는 정상 인트라 예측 모드에 의존할 수 있다.

[0610] iii. 일 예에서, 변환 행렬의 선택은 ALWIP 인트라 예측 모드로부터 전환되는 정상 인트라 예측 모드에 대한 분류에 의존할 수 있다.

[0611] 21. ALWIP 모드로 코딩된 블록은 블록 기반 DPCM(BDPCM) 또는 잔차 RDPCM을 적용할 수 없는 것이 제안된다.

[0612] a. 대안적으로, ALWIP 모드는 BDPCM 또는 RDPCM이 적용될 때 허용되지 않을 수 있다.

[0613] 22. ALWIP 모드로 코딩된 블록이 변환으로서 DCT-II만을 사용할 수 있는 것이 제안된다.

[0614] a. 일 예에서, 변환 행렬 인덱스들의 시그널링은 항상 스킵된다.

[0615] b. 대안적으로, ALWIP 모드로 코딩된 블록에 사용되는 변환이 명시적으로 시그널링되는 대신에 암시적으로 도출될 수 있는 것이 제안된다. 예를 들어, 변환은 JVET-M0303에서 제안된 방식에 따라 선택될 수 있다.

[0616] c. 대안적으로, ALWIP 모드로 코딩된 블록이 변환 스킵만을 사용할 수 있는 것이 제안된다.

[0617] i. 대안적으로, 또한, ALWIP가 사용될 때, 변환 스킵의 사용의 표시의 시그널링은 스킵된다.

[0618] d. 일 예에서, ALWIP 모드 정보(예컨대, 인에이블형/디스에이블형, 예측 모드 인덱스)는 변환 행렬의 표시들 이후에 조건부로 시그널링될 수 있다.

[0619] i. 일 예에서, (변환 스킵 또는 DCT-II와 같은) 주어진 변환 행렬에 대해, ALWIP 모드 정보의 표시들이 시그널링될 수 있다.

[0620] ii. 대안적으로, 또한, ALWIP 모드 정보의 표시들은 일부 미리 정의된 변환 행렬들에 대해 스킵될 수 있다.

[0621] 23. ALWIP 모드로 코딩된 블록은 선택된 변환이 모드-의존적일 때 ALWIP 인트라-예측 모드로부터 전환된 정상 인트라-예측으로 코딩되는 것으로 간주되는 것이 제안된다.

[0622] 24. ALWIP 모드는 변환 스킵을 사용하지 않을 수 있다.

- [0623] a. 예를 들어, 이 경우에 변환 스킵의 사용의 표시를 추가로 시그널링할 필요가 없다.
- [0624] b. 대안적으로, ALWIP 모드는 변환 스킵이 적용될 때 허용되지 않을 수 있다.
- [0625] i. 예를 들어, 이 경우에 변환 스킵이 적용될 때 ALWIP 모드 정보를 시그널링할 필요가 없다.
- [0626] 25. 디블록킹 필터, SAO(sample adaptive offset), ALF(adaptive loop filter)와 같은 필터링 프로세스에서, 필터들을 선택하는 방법 및/또는 샘플들을 필터링할지는 ALWIP의 사용에 의해 결정될 수 있다.
- [0627] 26. 필터링되지 않은 이웃 샘플들은 ALWIP 모드에서 사용될 수 있다.
- [0628] a. 대안적으로, 필터링된 이웃 샘플들은 ALWIP 모드에서 사용될 수 있다.
- [0629] b. 일 예에서, 필터링된 이웃 샘플들은 다운 샘플링에 사용될 수 있고 필터링되지 않은 이웃 샘플들은 업 샘플링에 사용될 수 있다.
- [0630] c. 일 예에서, 필터링되지 않은 이웃 샘플들은 다운 샘플링에 사용될 수 있고 필터링된 이웃 샘플들은 업 샘플링에 사용될 수 있다.
- [0631] d. 일 예에서, 필터링된 좌측 이웃 샘플들은 업 샘플링에서 사용될 수 있고 필터링되지 않은 상부 이웃 샘플들은 업 샘플링에서 사용될 수 있다.
- [0632] e. 일 예에서, 필터링되지 않은 좌측 이웃 샘플들은 업 샘플링에서 사용될 수 있고, 필터링된 상부 이웃 샘플들은 업 샘플링에서 사용될 수 있다.
- [0633] f. 일 예에서, 필터링된 이웃 샘플들이 사용되는지 또는 필터링되지 않은 이웃 샘플들이 사용되는지는 ALWIP 모드에 의존할 수 있다.
- [0634] i. 일 예에서, ALWIP 모드는 전통적인 인트라 예측 모드로 전환될 수 있고, 필터링된 이웃 샘플들이 사용되는지 또는 필터링되지 않은 이웃 샘플들이 사용되는지가 전환된 전통적인 인트라 예측 모드에 의존할 수 있다. 예를 들어, 그러한 결정은 전통적인 인트라 예측 모드들과 동일하다.
- [0635] ii. 대안적으로, ALWIP 모드에 필터링된 이웃 샘플들이 사용되는지 또는 필터링되지 않은 이웃 샘플들이 사용되는지가 시그널링될 수 있다.
- [0636] g. 일 예에서, 필터링된 샘플들은 전통적인 인트라 예측 모드들과 동일하게 생성될 수 있다.
- [0637] 27. 어느 행렬들 또는/및 오프셋 벡터들이 사용되는지는 재정형(reshaping)(일명, LMCS(luma mapping with chroma scaling)) 정보에 의존할 수 있다.
- [0638] a. 일 예에서, 재정형이 온 및 오프일 때 상이한 행렬들 또는/및 오프셋 벡터들이 사용될 수 있다.
- [0639] b. 일 예에서, 상이한 재정형 파라미터들에 대해 상이한 행렬들 또는/및 오프셋 벡터들이 사용될 수 있다.
- [0640] c. 일 예에서, ALWIP는 항상 원래의 도메인에서 수행될 수 있다.
- [0641] i. 예를 들어, 이웃 샘플은 ALWIP에서 사용되기 전에 (재정형이 적용되는 경우) 원래의 도메인에 매핑된다.
- [0642] 28. ALWIP는 재정형이 적용될 때 디스에이블될 수 있다.
- [0643] a. 대안적으로, ALWIP가 인에이블될 때 재정형이 디스에이블될 수 있다.
- [0644] b. 일 예에서, ALWIP는 재정형이 적용될 때 HDR(high dynamic range) 콘텐츠에 대해 디스에이블될 수 있다.
- [0645] 29. ALWIP에서 사용되는 행렬들은 샘플 비트-심도에 의존할 수 있다.
- [0646] a. 대안적으로, 또한, ALWIP에서 사용되는 오프셋 값들은 샘플 비트-심도에 의존할 수 있다.
- [0647] b. 대안적으로, 행렬 파라미터들 및 오프셋 값들은 N-비트 샘플들에 대해 M-비트 정밀도로 저장될 수 있고($M \leq N$), 예를 들어, 행렬 파라미터들 및 오프셋 값들은 10-비트 샘플에 대해 8-비트 정밀도로 저장될 수

있다.

- [0648] c. 샘플 비트-심도는 루마와 같은 컬러 컴포넌트에 대한 입력 어레이의 비트-심도일 수 있다.
- [0649] d. 샘플 비트-심도는 루마와 같은 컬러 컴포넌트에 대한 내부 어레이/재구성된 샘플의 비트-심도일 수 있다.
- [0650] 30. 특정된 블록 크기에 대한 행렬 파라미터들 및/또는 오프셋 값들은 다른 블록 크기들에 대한 행렬 파라미터들 및/또는 오프셋 값들로부터 도출될 수 있다.
- [0651] 31. 일 예에서, 8x8 블록의 16x8 행렬은 4x4 블록의 16x4 행렬로부터 도출될 수 있다.
- [0652] 32. ALWIP에 의해 생성된 예측은 추가로 사용될 예측 신호를 획득하기 위해 처리될 중간 신호로서 취급될 수 있는 것이 제안된다.
- [0653] a. 일 예에서, 위치 의존적 인트라 예측 조합(PDPC)은 ALWIP에 의해 생성된 예측에 적용되어 추가로 사용될 예측 신호를 생성할 수 있다.
- [0654] i. 일 예에서, PDPC는 ALWIP 코딩된 블록에 대해, 그 블록이 Planar 또는 DC와 같은 특정 정상 인트라 예측 모드로 코딩되는 것과 동일한 방식으로 행해진다.
- [0655] ii. 일 예에서, PDPC는 ALWIP 코딩된 블록에 대해, 그 블록이 ALWIP 인트라 예측 모드로부터 전환되는 정상 인트라 예측 모드로 코딩되는 것과 동일한 방식으로 행해진다.
- [0656] iii. 일 예에서, PDPC는 ALWIP 코딩된 블록에 조건부로 적용된다.
- [0657] 1) 예를 들어, PDPC는 PDPC가 ALWIP 인트라 예측 모드로부터 전환되는 정상 인트라 예측 모드에 적용될 때에만 ALWIP 코딩된 블록에 적용된다.
- [0658] b. 일 예에서, ALWIP에 의해 생성된 경계 샘플들 예측은 이웃 샘플들로 필터링되어 추가로 사용될 예측 신호를 생성할 수 있다.
- [0659] i. 일 예에서, 경계 샘플들에 대한 필터링은 ALWIP 코딩된 블록에 대해, 그 블록이 Planar 또는 DC와 같은 특정 정상 인트라-예측 모드로 코딩되는 것과 동일한 방식으로 행해진다.
- [0660] ii. 일 예에서, 경계 샘플들에 대한 필터링은 ALWIP 코딩된 블록에 대해, 그 블록이 ALWIP 인트라 예측 모드로부터 전환되는 정상 인트라-예측 모드로 코딩되는 것과 동일한 방식으로 행해진다.
- [0661] iii. 일 예에서, 경계 샘플들에 대한 필터링은 ALWIP 코딩된 블록에 조건부로 적용된다.
- [0662] 1) 예를 들어, 경계 샘플들에 대한 필터링은 경계 샘플들에 대한 필터링이 ALWIP 인트라-예측 모드로부터 전환되는 정상 인트라-예측 모드에 적용될 때에만 ALWIP 코딩된 블록에 적용된다.
- [0663] 33. 쌍선형 보간 필터 이외의 보간 필터들이 ALWIP의 업-샘플링 프로세스에서 사용될 수 있는 것이 제안된다.
- [0664] a. 일 예에서, 4-탭 보간 필터가 ALWIP의 업-샘플링 프로세스에서 사용될 수 있다.
- [0665] i. 예를 들어, 크로마 컴포넌트들에 대한 모션 보상을 행하기 위해 사용되는 VVC에서의 4-탭 보간 필터가 ALWIP의 업-샘플링 프로세스에서 사용될 수 있다.
- [0666] ii. 예를 들어, 각도 인트라-예측을 행하기 위해 사용되는 VVC에서의 4-탭 보간 필터는 ALWIP의 업-샘플링 프로세스에서 사용될 수 있다.
- [0667] iii. 예를 들어, 루마 컴포넌트에 대한 모션 보상을 수행하기 위해 사용되는 VVC에서의 8-탭 보간 필터가 ALWIP의 업-샘플링 프로세스에서 사용될 수 있다.
- [0668] 34. ALWIP 모드에서 코딩된 블록 내의 샘플들은 상이한 방식들로 예측될 수 있다.
- [0669] a. 일 예에서, W*H 블록에 대해, 그 내부의 sW*sH 서브-블록의 예측은 그것에 sW*sH ALWIP를 적용함으로써 생성될 수 있다.
- [0670] i. 일 예에서, W*H 블록에 대해, 그의 상단 좌측 W/2*H/2 블록의 예측은 그것에 W/2*H/2 ALWIP를 적용함으로써 생성될 수 있다.

- [0671] ii. 일 예에서, $W \times H$ 블록에 대해, 그의 좌측 $W/2 \times H$ 블록의 예측은 그것에 $W/2 \times H$ ALWIP를 적용함으로써 생성될 수 있다.
- [0672] iii. 일 예에서, $W \times H$ 블록에 대해, 그의 상단 $W \times H/2$ 블록의 예측은 그것에 $W \times H/2$ ALWIP를 적용함으로써 생성될 수 있다.
- [0673] iv. 일 예에서, $sW \times sH$ 서브-블록은 이용가능한 좌측 또는/및 상부 이웃 샘플들을 가질 수 있다.
- [0674] b. 일 예에서, 서브-블록의 위치를 결정하는 방법은 블록의 치수들에 의존할 수 있다.
- [0675] i. 예를 들어, $W \geq H$ 일 때, 그의 좌측 $W/2 \times H$ 블록의 예측은 그것에 $W/2 \times H$ ALWIP를 적용함으로써 생성될 수 있다.
- [0676] ii. 예를 들어, $H \geq W$ 일 때, 그의 상단 $W \times H/2$ 블록의 예측은 그것에 $W \times H/2$ ALWIP를 적용함으로써 생성될 수 있다.
- [0677] iii. 예를 들어, W 가 H 와 동일할 때, 그의 상단 좌측 $W/2 \times H/2$ 블록의 예측은 그것에 $W/2 \times H/2$ ALWIP를 적용함으로써 생성될 수 있다.
- [0678] c. 일 예에서, 또한, 나머지 샘플들(예를 들어, 샘플들은 $sW \times sH$ 서브-블록에 속하지 않음)의 예측은 $W \times H$ ALWIP를 적용함으로써 생성될 수 있다.
- [0679] i. 대안적으로, 나머지 샘플들의 예측은 종래의 인트라 예측을 적용함으로써(예를 들어, 인트라 모드로서 전환된 인트라 예측 모드를 사용하여) 생성될 수 있다.
- [0680] ii. 또한, $sW \times sH$ 서브-블록 내의 샘플들에 대해 계산이 스킵될 수 있다.
- [0681] 35. ALWIP 모드에서 코딩된 블록 내의 샘플들은(예를 들어, 크기 $sW \times sH$ 를 갖는) 서브-블록 레벨에서 예측될 수 있다.
- [0682] a. 일 예에서, $sW \times sH$ ALWIP는 이웃하는 재구성된 샘플들(예를 들어, 경계 서브-블록들에 대해) 및/또는 이웃하는 예측된 샘플들(예를 들어, 내부 서브-블록들에 대해)을 사용하여 각각의 서브-블록에 적용될 수 있다.
- [0683] b. 일 예에서, 서브-블록들은 래스터-스캔 순서로 예측될 수 있다.
- [0684] c. 일 예에서, 서브-블록들은 지그재그 순서로 예측될 수 있다.
- [0685] d. 일 예에서, 서브-블록들의 폭(높이)은 sW_{Max} (sH_{Max}) 이하일 수 있다.
- [0686] e. 일 예에서, 폭 또는 높이 중 어느 하나 또는 폭과 높이 둘 다를 갖는 블록이 둘 다 임계값 L 보다 클 때(또는 동일할 때), 블록은 다수의 서브-블록으로 분할될 수 있다.
- [0687] f. 임계값 L 은 미리 정의되거나 SPS/PPS/픽처/슬라이스/타일 그룹/타일 레벨에서 시그널링될 수 있다.
- [0688] i. 대안적으로, 임계값들은 블록 크기, 픽처 타입, 시간 레이어 인덱스 등과 같은 특정 코딩된 정보에 의존할 수 있다.
- [0689] 36. 이웃 샘플들(인접한 또는 비-인접한)이 ALWIP에서 사용되기 전에 필터링되는 것이 제안된다.
- [0690] a. 대안적으로, 이웃 샘플들은 ALWIP에서 사용되기 전에 필터링되지 않는다.
- [0691] b. 대안적으로, 이웃 샘플들은 ALWIP에서 사용되기 전에 조건부로 필터링된다.
- [0692] i. 예를 들어, 이웃 샘플들은 ALWIP 인트라 예측 모드가 하나 또는 일부 특정 값과 동일할 때에만 ALWIP에서 사용되기 전에 필터링된다.
- [0693] 전술된 예들은, 비디오 인코더 및/또는 디코더에서 구현될 수 있는, 후술되는 방법들, 예를 들어, 방법들(1100-1400 및 2000-2300)의 컨텍스트에 포함될 수 있다.
- [0694] 도 11은 비디오 처리를 위한 예시적인 방법의 흐름도를 도시한다. 방법(1100)은, 단계 1110에서, 현재 비디오 블록이 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계를 포함한다.

- [0695] 방법(1100)은, 단계 1120에서, 이 결정에 기초하여, 비-ALWIP 인트라 모드에 대한 MPM 리스트의 적어도 부분에 기초하여 ALWIP 모드에 대한 최고 확률 모드(MPM) 리스트의 적어도 부분을 구성하는 단계를 포함한다.
- [0696] 방법(1100)은, 단계 1130에서, ALWIP 모드에 대한 MPM 리스트에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0697] 일부 실시예들에서, ALWIP 모드의 MPM 리스트의 크기는 비-ALWIP 인트라 모드에 대한 MPM 리스트의 크기와 동일하다. 일 예에서, ALWIP 모드의 MPM 리스트의 크기는 6이다.
- [0698] 일부 실시예들에서, 방법(1100)은 ALWIP 모드에 대한 MPM 리스트에 디폴트 모드들을 삽입하는 단계를 추가로 포함한다. 일 예에서, 디폴트 모드들은 비-ALWIP 인트라 모드에 대한 MPM 리스트에 기초하는 ALWIP 모드에 대한 MPM 리스트의 부분 이전에 삽입된다. 다른 예에서, 디폴트 모드들은 비-ALWIP 인트라 모드에 대한 MPM 리스트에 기초하는 ALWIP 모드에 대한 MPM 리스트의 부분에 후속하여 삽입된다. 또 다른 예에서, 디폴트 모드들은 비-ALWIP 인트라 모드에 대한 MPM 리스트에 기초하는 ALWIP 모드에 대한 MPM 리스트의 부분과 인터리빙된 방식으로 삽입된다.
- [0699] 일부 실시예들에서, ALWIP 모드에 대한 MPM 리스트 및 비-ALWIP 인트라 모드에 대한 MPM 리스트를 구성하는 것은 하나 이상의 이웃 블록에 기초한다.
- [0700] 일부 실시예들에서, ALWIP 모드에 대한 MPM 리스트 및 비-ALWIP 인트라 모드에 대한 MPM 리스트를 구성하는 것은 현재 비디오 블록의 높이 또는 폭에 기초한다.
- [0701] 일부 실시예들에서, ALWIP 모드에 대한 MPM 리스트를 구성하는 것은 비-ALWIP 인트라 모드에 대한 MPM 리스트를 구성하기 위해 사용되는 제2 파라미터 세트와 상이한 제1 파라미터 세트에 기초한다.
- [0702] 일부 실시예들에서, 방법(1100)은 현재 비디오 블록의 이웃 블록이 ALWIP 모드로 코딩되었다고 결정하고, 비-ALWIP 인트라 모드에 대한 MPM 리스트를 구성할 때, 이웃 블록을 이용불가능한 것으로서 지정하는 단계를 추가로 포함한다.
- [0703] 일부 실시예들에서, 방법(1100)은 현재 비디오 블록의 이웃 블록이 비-ALWIP 인트라 모드로 코딩되었다고 결정하고, ALWIP 모드에 대한 MPM 리스트를 구성할 때, 이웃 블록을 이용불가능한 것으로서 지정하는 단계를 추가로 포함한다.
- [0704] 일부 실시예들에서, 비-ALWIP 인트라 모드는 정상 인트라 모드, 다중 참조 라인(MRL) 인트라 예측 모드 또는 인트라 서브-파티션(ISP) 틀에 기초한다.
- [0705] 도 12는 비디오 처리를 위한 예시적인 방법의 흐름도를 도시한다. 방법(1200)은, 단계 1210에서, 현재 비디오 블록의 루마 컴포넌트가 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계를 포함한다.
- [0706] 방법(1200)은, 단계 1220에서, 결정에 기초하여, 크로마 인트라 모드를 추천하는 단계를 포함한다.
- [0707] 방법(1200)은, 단계 1230에서, 크로마 인트라 모드에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0708] 일부 실시예들에서, 루마 컴포넌트는 크로마 컴포넌트의 미리 결정된 크로마 샘플을 커버한다. 일 예에서, 미리 결정된 크로마 샘플은 크로마 컴포넌트의 상단 좌측 샘플 또는 중심 샘플이다.
- [0709] 일부 실시예들에서, 추천된 크로마 인트라 모드는 DM 모드이다.
- [0710] 일부 실시예들에서, 추천된 크로마 인트라 모드는 ALWIP 모드이다.
- [0711] 일부 실시예들에서, ALWIP 모드는 현재 비디오 블록의 하나 이상의 크로마 컴포넌트에 적용된다.
- [0712] 일부 실시예들에서, ALWIP 모드의 상이한 행렬 또는 바이어스 벡터들이 현재 비디오 블록의 상이한 컬러 컴포넌트들에 적용된다. 일 예에서, 상이한 행렬 또는 바이어스 벡터들은 Cb 및 Cr 컴포넌트들에 대해 공동으로 미리 정의된다. 다른 예에서, Cb 및 Cr 컴포넌트들은 연결된다. 또 다른 예에서, Cb 및 Cr 컴포넌트들은 인터리빙된다.
- [0713] 도 13은 비디오 처리를 위한 예시적인 방법의 흐름도를 도시한다. 방법(1300)은, 단계 1310에서, 현재 비디오 블록이 아핀 선형 가중 인트라 예측(ALWIP) 모드를 사용하여 코딩되는 것을 결정하는 단계를 포함한다.

- [0714] 방법(1300)은, 단계 1320에서, 이 결정에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0715] 일부 실시예들에서, 이 결정은 시퀀스 파라미터 세트(SPS), 픽처 파라미터 세트(PPS), 슬라이스 헤더, 타일 그룹 헤더, 타일 헤더, 코딩 트리 유닛(CTU) 행 또는 CTU 영역에서의 시그널링에 기초한다.
- [0716] 일부 실시예들에서, 이 결정은 현재 비디오 블록의 높이(H) 또는 폭(W)에 기초한다. 일 예에서, $W > T1$ 이거나 $H > T2$ 이다. 다른 예에서, $W \geq T1$ 이거나 $H \geq T2$ 이다. 또 다른 예에서, $W < T1$ 이거나 $H < T2$ 이다. 또 다른 예에서, $W \leq T1$ 이거나 $H \leq T2$ 이다. 또 다른 예에서, $T1 = 32$ 이고 $T2 = 32$ 이다.
- [0717] 일부 실시예들에서, 이 결정은 현재 비디오 블록의 높이(H) 또는 폭(W)에 기초한다. 일 예에서, $W + H \leq T$ 이다. 다른 예에서, $W + H \geq T$ 이다. 또 다른 예에서, $W \times H \leq T$ 이다. 또 다른 예에서, $W \times H \geq T$ 이다. 또 다른 예에서, $T = 256$ 이다.
- [0718] 도 14는 비디오 처리를 위한 예시적인 방법의 흐름도를 도시한다. 방법(1400)은, 단계 1410에서, 현재 비디오 블록이 아핀 선형 가중 인트라 예측(ALWIP) 모드와 상이한 코딩 모드를 사용하여 코딩되는 것을 결정하는 단계를 포함한다.
- [0719] 방법(1400)은, 단계 1420에서, 결정에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 단계를 포함한다.
- [0720] 일부 실시예들에서, 코딩 모드는 조합된 인트라 및 인터 예측(CIIP) 모드이고, 방법(1400)은 ALWIP 모드와 정상 인트라 예측 모드 사이의 선택을 수행하는 단계를 추가로 포함한다. 일 예에서, 선택을 수행하는 것은 현재 비디오 블록의 비트스트림 표현에서의 명시적 시그널링에 기초한다. 다른 예에서, 선택을 수행하는 것은 미리 결정된 규칙에 기초한다. 또 다른 예에서, 미리 결정된 규칙은 현재 비디오 블록이 CIIP 모드를 사용하여 코딩될 때 항상 ALWIP 모드를 선택한다. 또 다른 예에서, 미리 결정된 규칙은 현재 비디오 블록이 CIIP 모드를 사용하여 코딩될 때 정상 인트라 예측 모드를 항상 선택한다.
- [0721] 일부 실시예들에서, 코딩 모드는 크로스 컴포넌트 선형 모델(CCLM) 예측 모드이다. 일 예에서, ALWIP 모드에 대한 다운샘플링 절차는 CCLM 예측 모드에 대한 다운샘플링 절차에 기초한다. 다른 예에서, ALWIP 모드에 대한 다운샘플링 절차는 제1 파라미터 세트에 기초하고, CCLM 예측 모드에 대한 다운샘플링 절차는 제1 파라미터 세트와 상이한 제2 파라미터 세트에 기초한다. 또 다른 예에서, ALWIP 모드 또는 CCLM 예측 모드에 대한 다운샘플링 절차는 라운딩 연산 또는 클리핑 연산, 다운샘플링 필터들의 선택, 다운샘플링된 위치들의 선택 중 적어도 하나를 포함한다.
- [0722] 일부 실시예들에서, 방법(1400)은 감소된 2차 변환(RST), 2차 변환, 회전 변환 또는 비분리형 2차 변환(NSST) 중 하나 이상을 적용하는 단계를 추가로 포함한다.
- [0723] 일부 실시예들에서, 방법(1400)은 DPCM(block-based differential pulse coded modulation) 또는 잔차 DPCM을 적용하는 단계를 추가로 포함한다.
- [0724] 6 개시된 기술의 예시적인 구현들
- [0725] 도 15는 비디오 처리 장치(1500)의 블록도이다. 장치(1500)는 본 명세서에서 설명된 방법들 중 하나 이상을 구현하기 위해 사용될 수 있다. 장치(1500)는 스마트폰, 태블릿, 컴퓨터, IoT(Internet of Things) 수신기 등으로 구체화될 수 있다. 장치(1500)는 하나 이상의 프로세서(1502), 하나 이상의 메모리(1504) 및 비디오 처리 하드웨어(1506)를 포함할 수 있다. 프로세서(들)(1502)는 본 문서에 설명된 하나 이상의 방법(방법들(1100 내지 1400 및 2000 내지 2300)을 포함하지만, 이들로 제한되지 않음)을 구현하도록 구성될 수 있다. 메모리(메모리들)(1504)는 본 명세서에서 설명된 방법들 및 기법들을 구현하기 위해 사용되는 코드 및 데이터를 저장하기 위해 사용될 수 있다. 비디오 처리 하드웨어(1506)는 본 문서에서 설명된 일부 기법들을 하드웨어 회로로 구현하기 위해 사용될 수 있다.
- [0726] 일부 실시예들에서, 비디오 코딩 방법들은 도 15와 관련하여 설명된 바와 같이 하드웨어 플랫폼 상에 구현되는 장치를 사용하여 구현될 수 있다.
- [0727] 도 16은 본 명세서에 개시된 다양한 기법들이 구현될 수 있는 예시적인 비디오 처리 시스템(1600)을 도시하는 블록도이다. 다양한 구현들은 시스템(1600)의 컴포넌트들의 일부 또는 전부를 포함할 수 있다. 시스템(1600)은 비디오 콘텐츠를 수신하기 위한 입력(1602)을 포함할 수 있다. 비디오 콘텐츠는 미가공(raw) 또는 압축되지

많은 포맷, 예를 들어, 8 또는 10비트 멀티 컴포넌트 픽셀 값들로 수신될 수 있거나, 압축된 또는 인코딩된 포맷일 수 있다. 입력(1602)은 네트워크 인터페이스, 주변기기 버스 인터페이스, 또는 저장 인터페이스를 나타낼 수 있다. 네트워크 인터페이스의 예는 이더넷, PON(passive optical network) 등과 같은 유선 인터페이스들 및 Wi-Fi 또는 셀룰러 인터페이스들과 같은 무선 인터페이스들을 포함한다.

[0728] 시스템(1600)은 본 문서에서 설명된 다양한 코딩 또는 인코딩 방법들을 구현할 수 있는 코딩 컴포넌트(1604)를 포함할 수 있다. 코딩 컴포넌트(1604)는 비디오의 평균 비트레이트를 입력(1602)으로부터 코딩 컴포넌트(1604)의 출력으로 감소시켜 비디오의 코딩된 표현을 생성할 수 있다. 따라서 코딩 기법들은 때때로 비디오 압축 또는 비디오 트랜스코딩 기법들이라고 불린다. 코딩 컴포넌트(1604)의 출력은, 컴포넌트(1606)로 나타낸 바와 같이, 저장되거나 연결된 통신을 통해 송신될 수 있다. 입력(1602)에서 수신된 비디오의 저장된 또는 통신된 비트스트림(또는 코딩된) 표현은, 디스플레이 인터페이스(1610)에 전송되는 픽셀 값들 또는 디스플레이가능한 비디오를 생성하기 위해 컴포넌트(1608)에 의해 사용될 수 있다. 비트스트림 표현으로부터 사용자-시청가능 비디오를 생성하는 프로세스는 때때로 비디오 압축해제라고 불린다. 게다가, 특정 비디오 처리 동작이 "코딩" 동작들 또는 툴들이라고 지칭되지만, 코딩 툴들 또는 동작들은 인코더에서 사용되고 코딩의 결과들을 반대로 행하는 대응하는 디코딩 툴들 또는 동작들은 디코더에 의해 수행될 것임이 이해될 것이다.

[0729] 주변기기 버스 인터페이스 또는 디스플레이 인터페이스의 예들은 USB(universal serial bus) 또는 HDMI(high definition multimedia interface) 또는 Displayport 등을 포함할 수 있다. 저장 인터페이스들의 예들은 SATA(serial advanced technology attachment), PCI, IDE 인터페이스 등을 포함한다. 본 문서에서 설명된 기법들은, 디지털 데이터 처리 및/또는 비디오 디스플레이를 수행할 수 있는 모바일 폰들, 랩톱들, 스마트폰들 또는 다른 디바이스들과 같은 다양한 전자 디바이스들에서 구체화될 수 있다.

[0730] 개시된 기술의 일부 실시예들은 비디오 처리 툴 또는 모드를 인에이블하라는 판정 또는 결정을 하는 것을 포함한다. 일 예에서, 비디오 처리 툴 또는 모드가 인에이블될 때, 인코더는 비디오의 블록의 처리에서 툴 또는 모드를 사용하거나 구현할 것이지만, 툴 또는 모드의 사용에 기초하여 결과적인 비트스트림을 반드시 수정할 필요는 없다. 즉, 비디오의 블록으로부터 비디오의 비트스트림 표현으로의 전환은 이 판정 또는 결정에 기초하여 비디오 처리 툴 또는 모드가 인에이블될 때 그것을 사용할 것이다. 다른 예에서, 비디오 처리 툴 또는 모드가 인에이블될 때, 디코더는 비트스트림이 비디오 처리 툴 또는 모드에 기초하여 수정되었다는 것을 알고서 비트스트림을 처리할 것이다. 즉, 비디오의 비트스트림 표현으로부터 비디오의 블록으로의 전환은 이 판정 또는 결정에 기초하여 인에이블되었던 비디오 처리 툴 또는 모드를 사용하여 수행될 것이다.

[0731] 개시된 기술의 일부 실시예들은 비디오 처리 툴 또는 모드를 디스에이블하라는 판정 또는 결정을 하는 것을 포함한다. 일 예에서, 비디오 처리 툴 또는 모드가 디스에이블될 때, 인코더는 비디오의 블록을 비디오의 비트스트림 표현으로 전환할 때 툴 또는 모드를 사용하지 않을 것이다. 다른 예에서, 비디오 처리 툴 또는 모드가 디스에이블될 때, 디코더는 이 판정 또는 결정에 기초하여 디스에이블되었던 비디오 처리 툴 또는 모드를 사용하여 비트스트림이 수정되지 않았다는 것을 알고서 비트스트림을 처리할 것이다.

[0732] 도 17은 본 개시의 기술들을 활용할 수 있는 예시적인 비디오 코딩 시스템(100)을 예시하는 블록도이다. 도 17에 도시된 바와 같이, 비디오 코딩 시스템(100)은 소스 디바이스(110) 및 목적지 디바이스(120)를 포함할 수 있다. 비디오 인코딩 디바이스라고 지칭될 수 있는 소스 디바이스(110)는 인코딩된 비디오 데이터를 생성한다. 비디오 디코딩 디바이스라고 지칭될 수 있는 목적지 디바이스(120)는 소스 디바이스(110)에 의해 생성되는 인코딩된 비디오 데이터를 디코딩할 수 있다. 소스 디바이스(110)는 비디오 소스(112), 비디오 인코더(114), 및 입출력(I/O) 인터페이스(116)를 포함할 수 있다.

[0733] 비디오 소스(112)는 비디오 캡처 디바이스와 같은 소스, 비디오 콘텐츠 제공자로부터 비디오 데이터를 수신하기 위한 인터페이스, 및/또는 비디오 데이터를 생성하기 위한 컴퓨터 그래픽 시스템, 또는 그러한 소스들의 조합을 포함할 수 있다. 비디오 데이터는 하나 이상의 픽처를 포함할 수 있다. 비디오 인코더(114)는 비디오 소스(112)로부터의 비디오 데이터를 인코딩하여 비트스트림을 생성한다. 비트스트림은 비디오 데이터의 코딩된 표현을 형성하는 비트 시퀀스를 포함할 수 있다. 비트스트림은 코딩된 픽처들 및 연관된 데이터를 포함할 수 있다. 코딩된 픽처는 픽처의 코딩된 표현이다. 연관된 데이터는 시퀀스 파라미터 세트들, 픽처 파라미터 세트들, 및 다른 신택스 구조들을 포함할 수 있다. I/O 인터페이스(116)는 변조기/복조기(모뎀) 및/또는 송신기를 포함할 수 있다. 인코딩된 비디오 데이터는 I/O 인터페이스(116)를 통해 네트워크(130a)를 거쳐 목적지 디바이스(120)로 직접 전송될 수 있다. 인코딩된 비디오 데이터는 또한 목적지 디바이스(120)에 의한 액세스를 위해 저장 매체/서버(130b)에 저장될 수 있다.

- [0734] 목적지 디바이스(120)는 I/O 인터페이스(126), 비디오 디코더(124), 및 디스플레이 디바이스(122)를 포함할 수 있다.
- [0735] 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 취득할 수 있다. 비디오 디코더(124)는 인코딩된 비디오 데이터를 디코딩할 수 있다. 디스플레이 디바이스(122)는 디코딩된 비디오 데이터를 사용자에게 디스플레이할 수 있다. 디스플레이 디바이스(122)는 목적지 디바이스(120)와 통합될 수 있거나, 또는 외부 디스플레이 디바이스와 인터페이싱하도록 구성된 목적지 디바이스(120)의 외부에 있을 수 있다.
- [0736] 비디오 인코더(114) 및 비디오 디코더(124)는, HEVC(High Efficiency Video Coding) 표준, VVM(Versatile Video Coding) 표준 및 다른 현재 및/또는 추가 표준들과 같은, 비디오 압축 표준에 따라 동작할 수 있다.
- [0737] 도 18은 도 17에 예시된 시스템(100) 내의 비디오 인코더(114)일 수 있는, 비디오 인코더(200)의 예를 예시하는 블록도이다.
- [0738] 비디오 인코더(200)는 본 개시의 기술들의 일부 또는 전부를 수행하도록 구성될 수 있다. 도 18의 예에서, 비디오 인코더(200)는 복수의 기능 컴포넌트를 포함한다. 본 개시에 설명된 기술들은 비디오 인코더(200)의 다양한 컴포넌트들 사이에서 공유될 수 있다. 일부 예들에서, 프로세서는 본 개시에 설명된 기술들의 일부 또는 전부를 수행하도록 구성될 수 있다.
- [0739] 비디오 인코더(200)의 기능 컴포넌트들은 파티션 유닛(201), 모드 선택 유닛(203), 모션 추정 유닛(204), 모션 보상 유닛(205) 및 인트라 예측 유닛(206)을 포함할 수 있는 예측 유닛(202), 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)을 포함할 수 있다.
- [0740] 다른 예들에서, 비디오 인코더(200)는 보다 많은, 보다 적은 또는 상이한 기능 컴포넌트들을 포함할 수 있다. 일 예에서, 예측 유닛(202)은 인트라 블록 카피(intra block copy, IBC) 유닛을 포함할 수 있다. IBC 유닛은 적어도 하나의 참조 픽처가 현재 비디오 블록이 위치하는 픽처인 IBC 모드에서 예측을 수행할 수 있다.
- [0741] 또한, 모션 추정 유닛(204) 및 모션 보상 유닛(205)과 같은 일부 컴포넌트들은 고도로 통합될 수 있지만, 설명을 위해 도 18의 예에서 개별적으로 표현된다.
- [0742] 파티션 유닛(201)은 픽처를 하나 이상의 비디오 블록으로 파티셔닝할 수 있다. 비디오 인코더(200) 및 비디오 디코더(300)는 다양한 비디오 블록 크기들을 지원할 수 있다.
- [0743] 모드 선택 유닛(203)은, 예를 들어, 에러 결과들에 기초하여, 코딩 모드들, 인트라 또는 인터 중 하나를 선택하고, 결과적인 인트라-코딩된 또는 인터-코딩된 블록을 잔차 생성 유닛(207)에 제공하여 잔차 블록 데이터를 생성하고 재구성 유닛(212)에 제공하여 참조 픽처로서 사용하기 위한 인코딩된 블록을 재구성할 수 있다. 일부 예에서, 모드 선택 유닛(203)은 예측이 인터 예측 신호 및 인트라 예측 신호에 기초하는 CIIP(combination of intra and inter predication) 모드를 선택할 수 있다. 모드 선택 유닛(203)은 또한 인터 예측의 경우에 블록에 대한 모션 벡터의 분해능(예를 들면, 서브 픽셀 또는 정수 픽셀 정밀도)을 선택할 수 있다.
- [0744] 현재 비디오 블록에 대한 인터 예측을 수행하기 위해, 모션 추정 유닛(204)은 버퍼(213)로부터의 하나 이상의 참조 프레임을 현재 비디오 블록과 비교하는 것에 의해 현재 비디오 블록에 대한 모션 정보를 생성할 수 있다. 모션 보상 유닛(205)은 현재 비디오 블록과 연관된 픽처 이외의 버퍼(213)로부터의 픽처들의 디코딩된 샘플들 및 모션 정보에 기초하여 현재 비디오 블록에 대한 예측된 비디오 블록을 결정할 수 있다.
- [0745] 모션 추정 유닛(204) 및 모션 보상 유닛(205)은, 예를 들어, 현재 비디오 블록이 I 슬라이스, P 슬라이스, 또는 B 슬라이스에 있는지에 따라, 현재 비디오 블록에 대해 상이한 동작들을 수행할 수 있다.
- [0746] 일부 예들에서, 모션 추정 유닛(204)은 현재 비디오 블록에 대한 단방향 예측을 수행할 수 있고, 모션 추정 유닛(204)은 현재 비디오 블록에 대한 참조 비디오 블록에 대해 리스트 0 또는 리스트 1의 참조 픽처들을 탐색할 수 있다. 모션 추정 유닛(204)은 이어서 참조 비디오 블록을 포함하는 리스트 0 또는 리스트 1 내의 참조 픽처를 표시하는 참조 인덱스 및 현재 비디오 블록과 참조 비디오 블록 사이의 공간적 변위를 표시하는 모션 벡터를 생성할 수 있다. 모션 추정 유닛(204)은 참조 인덱스, 예측 방향 표시자, 및 모션 벡터를 현재 비디오 블록의 모션 정보로서 출력할 수 있다. 모션 보상 유닛(205)은 현재 비디오 블록의 모션 정보가 표시하는 참조 비디오 블록에 기초하여 현재 블록의 예측된 비디오 블록을 생성할 수 있다.

- [0747] 다른 예들에서, 모션 추정 유닛(204)은 현재 비디오 블록에 대해 양방향 예측을 수행할 수 있고, 모션 추정 유닛(204)은 현재 비디오 블록에 대한 참조 비디오 블록에 대해 리스트 0 내의 참조 픽처들을 탐색할 수 있고, 또한 현재 비디오 블록에 대한 다른 참조 비디오 블록에 대해 리스트 1 내의 참조 픽처들을 탐색할 수 있다. 모션 추정 유닛(204)은 이어서 참조 비디오 블록들을 포함하는 리스트 0 및 리스트 1 내의 참조 픽처들을 표시하는 참조 인덱스들 및 참조 비디오 블록들과 현재 비디오 블록 사이의 공간적 변위들을 표시하는 모션 벡터들을 생성할 수 있다. 모션 추정 유닛(204)은 현재 비디오 블록의 모션 정보로서 참조 인덱스들 및 현재 비디오 블록의 모션 벡터들을 출력할 수 있다. 모션 보상 유닛(205)은 현재 비디오 블록의 모션 정보가 표시하는 참조 비디오 블록들에 기초하여 현재 비디오 블록의 예측된 비디오 블록을 생성할 수 있다.
- [0748] 일부 예들에서, 모션 추정 유닛(204)은 디코더의 디코딩 처리를 위한 모션 정보의 전체 세트를 출력할 수 있다.
- [0749] 일부 예들에서, 모션 추정 유닛(204)은 현재 비디오에 대한 모션 정보의 전체 세트를 출력하지 않을 수 있다. 오히려, 모션 추정 유닛(204)은 다른 비디오 블록의 모션 정보를 참조하여 현재 비디오 블록의 모션 정보를 시그널링할 수 있다. 예를 들어, 모션 추정 유닛(204)은 현재 비디오 블록의 모션 정보가 이웃 비디오 블록의 모션 정보와 충분히 유사하다고 결정할 수 있다.
- [0750] 일 예에서, 모션 추정 유닛(204)은, 현재 비디오 블록과 연관된 선택스 구조에서, 현재 비디오 블록이 다른 비디오 블록과 동일한 모션 정보를 갖는 것을 비디오 디코더(300)에 표시하는 값을 표시할 수 있다.
- [0751] 다른 예에서, 모션 추정 유닛(204)은, 현재 비디오 블록과 연관된 선택스 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 표시한다. 비디오 디코더(300)는 표시된 비디오 블록의 모션 벡터 및 모션 벡터 차이를 사용하여 현재 비디오 블록의 모션 벡터를 결정할 수 있다.
- [0752] 위에서 논의된 바와 같이, 비디오 인코더(200)는 모션 벡터를 예측적으로 시그널링할 수 있다. 비디오 인코더(200)에 의해 구현될 수 있는 예측적 시그널링 기술들의 두 가지 예는 AMVP(advanced motion vector predication) 및 병합 모드 시그널링을 포함한다.
- [0753] 인트라 예측 유닛(206)은 현재 비디오 블록에 대해 인트라 예측을 수행할 수 있다. 인트라 예측 유닛(206)이 현재 비디오 블록에 대해 인트라 예측을 수행할 때, 인트라 예측 유닛(206)은 동일한 픽처 내의 다른 비디오 블록들의 디코딩된 샘플들에 기초하여 현재 비디오 블록에 대한 예측 데이터를 생성할 수 있다. 현재 비디오 블록에 대한 예측 데이터는 예측된 비디오 블록 및 다양한 선택스 요소들을 포함할 수 있다.
- [0754] 잔차 생성 유닛(207)은 현재 비디오 블록으로부터 현재 비디오 블록의 예측된 비디오 블록(들)을 감산(예를 들면, 마이너스 부호로 표시됨)하는 것에 의해 현재 비디오 블록에 대한 잔차 데이터를 생성할 수 있다. 현재 비디오 블록의 잔차 데이터는 현재 비디오 블록 내의 샘플들의 상이한 샘플 컴포넌트들에 대응하는 잔차 비디오 블록들을 포함할 수 있다.
- [0755] 다른 예들에서, 예를 들어, 스킵 모드에서 현재 비디오 블록에 대한 현재 비디오 블록의 잔차 데이터가 없을 수 있고, 잔차 생성 유닛(207)은 감산 동작을 수행하지 않을 수 있다.
- [0756] 변환 처리 유닛(208)은 현재 비디오 블록과 연관된 잔차 비디오 블록에 하나 이상의 변환을 적용하는 것에 의해 현재 비디오 블록에 대한 하나 이상의 변환 계수 비디오 블록을 생성할 수 있다.
- [0757] 변환 처리 유닛(208)이 현재 비디오 블록과 연관된 변환 계수 비디오 블록을 생성한 후에, 양자화 유닛(209)은 현재 비디오 블록과 연관된 하나 이상의 양자화 파라미터(QP) 값에 기초하여 현재 비디오 블록과 연관된 변환 계수 비디오 블록을 양자화할 수 있다.
- [0758] 역양자화 유닛(210) 및 역변환 유닛(211)은, 제각기, 변환 계수 비디오 블록에 역양자화 및 역변환을 적용하여 변환 계수 비디오 블록으로부터 잔차 비디오 블록을 재구성할 수 있다. 재구성 유닛(212)은 버퍼(213)에 저장할 현재 블록과 연관된 재구성된 비디오 블록을 생성하기 위해 예측 유닛(202)에 의해 생성되는 하나 이상의 예측된 비디오 블록으로부터의 대응하는 샘플들에 재구성된 잔차 비디오 블록을 가산할 수 있다.
- [0759] 재구성 유닛(212)이 비디오 블록을 재구성한 후에, 비디오 블록에서의 비디오 블로킹 아티팩트들을 감소시키기 위해 루프 필터링 동작이 수행될 수 있다.
- [0760] 엔트로피 인코딩 유닛(214)은 비디오 인코더(200)의 다른 기능 컴포넌트들로부터 데이터를 수신할 수 있다.

엔트로피 인코딩 유닛(214)이 데이터를 수신할 때, 엔트로피 인코딩 유닛(214)은 엔트로피 인코딩된 데이터를 생성하고 엔트로피 인코딩된 데이터를 포함하는 비트스트림을 출력하기 위해 하나 이상의 엔트로피 인코딩 동작을 수행할 수 있다.

- [0761] 도 19는 도 17에 예시된 시스템(100) 내의 비디오 디코더(114)일 수 있는 비디오 디코더(300)의 예를 예시하는 블록도이다.
- [0762] 비디오 디코더(300)는 본 개시의 기술들의 일부 또는 전부를 수행하도록 구성될 수 있다. 도 19의 예에서, 비디오 디코더(300)는 복수의 기능 컴포넌트를 포함한다. 본 개시에 설명된 기술들은 비디오 디코더(300)의 다양한 컴포넌트들 사이에서 공유될 수 있다. 일부 예들에서, 프로세서는 본 개시에 설명된 기술들의 일부 또는 전부를 수행하도록 구성될 수 있다.
- [0763] 도 19의 예에서, 비디오 디코더(300)는 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)를 포함한다. 비디오 디코더(300)는, 일부 예들에서, 비디오 인코더(200)(도 18)와 관련하여 설명된 인코딩 패스(encoding pass)와 일반적으로 반대인 디코딩 패스(decoding pass)를 수행할 수 있다.
- [0764] 엔트로피 디코딩 유닛(301)은 인코딩된 비트스트림을 검색할 수 있다. 인코딩된 비트스트림은 엔트로피 코딩된 비디오 데이터(예를 들면, 비디오 데이터의 인코딩된 블록들)를 포함할 수 있다. 엔트로피 디코딩 유닛(301)은 엔트로피 코딩된 비디오 데이터를 디코딩할 수 있고, 엔트로피 디코딩된 비디오 데이터로부터, 모션 보상 유닛(302)은 모션 벡터들, 모션 벡터 정밀도, 참조 픽처 리스트 인덱스들, 및 다른 모션 정보를 포함하는 모션 정보를 결정할 수 있다. 모션 보상 유닛(302)은, 예를 들어, AMVP 및 병합 모드를 수행하는 것에 의해 그러한 정보를 결정할 수 있다.
- [0765] 모션 보상 유닛(302)은 모션 보상된 블록들을 생성할 수 있으며, 가능하게는 보간 필터들에 기초하여 보간을 수행할 수 있다. 사용될 보간 필터들에 대한 식별자들이 서브-픽셀 정밀도와 함께 신택스 요소들에 포함될 수 있다.
- [0766] 모션 보상 유닛(302)은 참조 블록의 서브-정수 픽셀들에 대한 보간된 값들을 계산하기 위해 비디오 블록의 인코딩 동안 비디오 인코더(20)에 의해 사용되는 보간 필터들을 사용할 수 있다. 모션 보상 유닛(302)은 수신된 신택스 정보에 따라 비디오 인코더(200)에 의해 사용되는 보간 필터들을 결정할 수 있고 예측 블록들을 생성하기 위해 보간 필터들을 사용할 수 있다.
- [0767] 모션 보상 유닛(302)은 인코딩된 비디오 시퀀스의 프레임(들) 및/또는 슬라이스(들)를 인코딩하기 위해 사용되는 블록들의 크기들, 인코딩된 비디오 시퀀스의 픽처의 각각의 매크로블록이 어떻게 파티셔닝되는지를 설명하는 파티션 정보, 각각의 파티션이 어떻게 인코딩되는지를 표시하는 모드들, 각각의 인트라-인코딩된 블록에 대한 하나 이상의 참조 프레임(및 참조 프레임 리스트들), 및 인코딩된 비디오 시퀀스를 디코딩하기 위한 다른 정보를 결정하기 위해 신택스 정보의 일부를 사용할 수 있다.
- [0768] 인트라 예측 유닛(303)은 공간적으로 인접한 블록들로부터 예측 블록을 형성하기 위해, 예를 들어, 비트스트림에서 수신되는 인트라 예측 모드들을 사용할 수 있다. 역양자화 유닛(303)은 비트스트림에서 제공되고 엔트로피 디코딩 유닛(301)에 의해 디코딩되는 양자화된 비디오 블록 계수들을 역양자화(inverse quantize), 즉 탈양자화(de-quantize)한다. 역변환 유닛(303)은 역변환을 적용한다.
- [0769] 재구성 유닛(306)은 디코딩된 블록들을 형성하기 위해 모션 보상 유닛(302) 또는 인트라 예측 유닛(303)에 의해 생성되는 대응하는 예측 블록들과 잔차 블록들을 합산할 수 있다. 원하는 경우, 디코딩된 블록들을 필터링하여 블록성 아티팩트(blockiness artifact)를 제거하기 위해 디블록킹 필터가 또한 적용될 수 있다. 디코딩된 비디오 블록들은 이어서 버퍼(307)에 저장되고, 버퍼는 후속 모션 보상/인트라 예측을 위한 참조 블록들을 제공하고 또한 디스플레이 디바이스 상에 제시하기 위한 디코딩된 비디오를 생성한다.
- [0770] 일부 실시예들에서, ALWIP 모드 또는 MIP 모드에서, 현재 비디오 블록에 대한 예측 블록은 예측 블록을 결정하기 위한 행 및 열 방향 평균화, 이어서 행렬 곱셈, 이어서 보간에 의해 결정된다.
- [0771] 도 20은 행렬-기반 인트라 예측을 위한 예시적인 방법(2000)의 예시적인 흐름도를 도시한다. 동작(2002)은 규칙에 따라 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 것을 포함하고, 규칙은 전환 동안 행렬 가중 인트라 예측(MIP) 모드에서 적용되는 행렬들 또는 오프셋 값들과 현재 비디오 블록의 샘플들 사이의 관계를 특정하고, MIP 모드는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘

플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 예측 블록을 결정하는 것을 포함한다.

[0772] 방법(2000)에 대한 일부 실시예들에서, 규칙은 MIP 모드에서 적용되는 행렬들의 요소들이 샘플들의 비트-심도에 의존하는 것을 특징한다. 방법(2000)에 대한 일부 실시예들에서, 규칙은 MIP 모드에서 적용되는 오프셋 값들이 샘플들의 비트-심도에 의존하는 것을 특징한다. 방법(2000)에 대한 일부 실시예들에서, 규칙은 행렬들의 요소들 및 오프셋 값들이 N-비트 정밀도를 갖는 샘플들에 대해 M-비트 정밀도를 갖는 것을 특징하고, 여기서 M은 N 이하이다. 방법(2000)에 대한 일부 실시예들에서, M은 8이고 N은 10이다. 방법(2000)에 대한 일부 실시예들에서, 샘플들의 비트-심도는 컬러 컴포넌트에 대한 입력 어레이의 제2 비트-심도와 동일하다. 방법(2000)에 대한 일부 실시예들에서, 샘플들의 비트-심도는 컬러 컴포넌트에 대한 재구성된 샘플 또는 내부 어레이의 제2 비트-심도와 동일하다. 방법(2000)에 대한 일부 실시예들에서, 컬러 컴포넌트는 루마 컴포넌트를 포함한다. 방법(2000)에 대한 일부 실시예들에서, 현재 비디오 블록에 대한 오프셋 값들 및/또는 행렬들에 대한 제1 파라미터 세트는 다른 비디오 블록의 제2 오프셋 값 세트 및/또는 제2 행렬 세트에 대한 제2 파라미터 세트로부터 도출된다. 방법(2000)에 대한 일부 실시예들에서, 현재 비디오 블록은 8x8 비디오 블록을 포함하고, 다른 비디오 블록은 4x4 비디오 블록을 포함하며, 16x8 행렬에 대한 제1 파라미터 세트는 16x4 행렬에 대한 제2 파라미터 세트로부터 도출된다.

[0773] 도 21은 행렬-기반 인트라 예측을 위한 예시적인 방법(2100)의 예시적인 흐름도를 도시한다. 동작(2102)은, 현재 비디오 블록에 대해, 행렬 가중 인트라 예측(MIP) 모드를 사용하여 중간 예측 블록을 생성하는 것- MIP 모드에서는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 중간 예측 블록이 결정됨 -을 포함한다. 동작(2104)은, 중간 예측 블록에 기초하여, 추가 동작에 기초하여 최종 예측 블록을 생성하는 것을 포함한다. 동작(2106)은, 최종 예측 신호에 기초하여, 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 것을 포함한다.

[0774] 방법(2100)에 대한 일부 실시예들에서, 추가 동작은 위치 의존적 인트라 예측 조합(PDPC)이다. 방법(2100)에 대한 일부 실시예들에서, PDPC를 사용하여 최종 예측 신호를 생성하는 것을 포함하는 제1 동작은 PDPC를 인트라 예측 모드를 사용하여 생성된 예측 신호에 적용하는 것을 포함하는 제2 동작과 동일하다. 방법(2100)에 대한 일부 실시예들에서, 인트라 예측 모드는 평면 모드 또는 DC 모드를 포함한다. 방법(2100)에 대한 일부 실시예들에서, PDPC를 사용하여 최종 예측 신호를 생성하는 것을 포함하는 제1 동작은 PDPC를 인트라 예측 모드를 사용하여 생성된 예측 신호에 적용하는 것을 포함하는 제2 동작과 동일하고, 인트라 예측 모드는 MIP 모드로부터 전환된다.

[0775] 방법(2100)에 대한 일부 실시예들에서, PDPC는 규칙에 기초하여 현재 비디오 블록의 중간 예측 블록에 적용된다. 방법(2100)에 대한 일부 실시예들에서, 규칙은, PDPC가 MIP 모드로부터 전환되는 인트라-예측 모드에 의해 생성되는 예측 신호에 적용되는 것에 응답하여, PDPC가 현재 비디오 블록의 필드중간 예측 블록에 적용될 것이라는 것을 표시한다. 방법(2100)에 대한 일부 실시예들에서, 추가 동작은 현재 비디오 블록의 경계 샘플들이 현재 비디오 블록의 이웃 샘플들로 필터링되는 필터링 동작이다. 방법(2100)에 대한 일부 실시예들에서, MIP 모드로 코딩된 현재 비디오 블록의 경계 샘플들을 필터링하기 위한 필터링 동작은 인트라-예측 모드를 사용하여 경계 샘플들을 필터링하기 위한 다른 필터링 동작과 동일하다.

[0776] 방법(2100)에 대한 일부 실시예들에서, 인트라-예측 모드는 평면 모드 또는 DC(direct current) 모드를 포함한다. 방법(2100)에 대한 일부 실시예들에서, MIP 모드로 코딩된 현재 비디오 블록의 경계 샘플들을 필터링하기 위한 필터링 동작은 인트라-예측 모드를 사용하여 경계 샘플들을 필터링하기 위한 다른 필터링 동작과 동일하고, 인트라-예측 모드는 MIP 모드로부터 전환된다. 방법(2100)에 대한 일부 실시예들에서, 필터링 동작은 규칙에 기초하여 적용된다. 방법(2100)에 대한 일부 실시예들에서, 규칙은, 경계 샘플들이 MIP 모드로부터 전환되는 인트라-예측 모드로 필터링되는 것에 응답하여 경계 샘플들을 필터링하기 위해 필터링 동작이 적용되는 것을 표시한다.

[0777] 도 22는 행렬-기반 인트라 예측을 위한 예시적인 방법(2200)의 예시적인 흐름도를 도시한다. 동작(2202)은 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 것을 포함하고, 전환은 행렬 가중 인트라 예측(MIP) 모드에서 현재 비디오 블록의 적어도 부분의 복수의 샘플을 예측하는 것을 포함하는데, MIP 모드에서는 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 부분의 예측 블록이 결정된다.

- [0778] 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 현재 비디오 블록은 폭(W) 및 높이(H)를 갖고, 서브-블록은 폭(sW) 및 높이(sH)를 갖고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다. 방법(2200)에 대한 일부 실시예들에서, 폭(sW) 및 높이(sH)를 갖는 서브-블록의 복수의 샘플은 현재 비디오 블록의 좌측 이웃 샘플들 또는 현재 비디오 블록의 상부 이웃 샘플들을 포함한다. 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 현재 비디오 블록은 폭(W) 및 높이(H)를 갖고, 서브-블록은 현재 비디오 블록의 상단 좌측 $W/2 \times H/2$ 블록이고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다. 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 현재 비디오 블록은 폭(W) 및 높이(H)를 갖고, 서브-블록은 현재 비디오 블록의 좌측 $W/2 \times H$ 블록이고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다.
- [0779] 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 현재 비디오 블록은 폭(W) 및 높이(H)를 갖고, 서브-블록은 현재 비디오 블록의 상단 $W \times H/2$ 블록이고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다. 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 현재 비디오 블록은 폭(W) 및 높이(H)를 갖고, 서브-블록은 폭(sW) 및 높이(sH)를 갖고, 서브-블록에 대한 복수의 샘플은 현재 비디오 블록의 좌측 이웃 샘플들을 사용함으로써 또는 현재 비디오 블록의 상부 이웃 샘플들을 사용함으로써 MIP를 서브-블록에 적용함으로써 예측된다.
- [0780] 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 서브-블록의 위치는 현재 비디오 블록의 폭(W)과 높이(H) 사이의 관계에 기초한다. 방법(2200)에 대한 일부 실시예들에서, 서브-블록은 $W \geq H$ 에 응답하여 현재 비디오 블록의 좌측 $W/2 \times H$ 블록이고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다. 방법(2200)에 대한 일부 실시예들에서, 서브-블록은 $H \geq W$ 에 응답하여 현재 비디오 블록의 상단 $W \times H/2$ 블록이고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다.
- [0781] 방법(2200)에 대한 일부 실시예들에서, 서브-블록은 $W = H$ 에 응답하여 현재 비디오 블록의 상단 좌측 $W/2 \times H/2$ 블록이고, 서브-블록에 대한 복수의 샘플은 MIP를 서브-블록에 적용함으로써 예측된다. 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 이 방법은: 현재 비디오 블록의 제2 샘플 세트를 예측하는 단계를 추가로 포함하고, 제2 샘플 세트는 서브-블록 외부에 위치하고, 제2 샘플 세트는 MIP를 현재 비디오 블록에 적용함으로써 예측된다.
- [0782] 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 방법은: 현재 비디오 블록의 제2 샘플 세트를 예측하는 단계를 추가로 포함하고, 제2 샘플 세트는 서브-블록의 외부에 위치하고, 제2 샘플 세트는 현재 비디오 블록에 인트라 예측 모드를 적용함으로써 예측되고, 인트라 예측 모드는 MIP 모드로부터 전환된다. 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 서브-블록에 속하고, 방법은: 현재 비디오 블록의 제2 샘플 세트를 예측하는 단계를 추가로 포함하고, 제2 샘플 세트는 서브-블록 외부에 위치하고, 제2 샘플 세트는 서브-블록을 제외한 현재 비디오 블록의 영역에 MIP를 적용함으로써 예측된다.
- [0783] 방법(2200)에 대한 일부 실시예들에서, 복수의 샘플은 현재 비디오 블록의 적어도 하나의 서브-블록에 속한다. 방법(2200)에 대한 일부 실시예들에서, 각각의 서브-블록에 대해, MIP를 서브-블록에 적용함으로써 복수의 샘플이 예측되고, 각각의 서브-블록에 대해, MIP는 서브-블록에 대한 이웃하는 재구성된 샘플들을 사용함으로써 그리고/또는 서브-블록에 대한 이웃하는 예측된 샘플들을 사용함으로써 서브-블록에 적용된다. 방법(2200)에 대한 일부 실시예들에서, 이웃하는 재구성된 샘플들은 현재 비디오 블록의 경계에 위치한 서브-블록에 대해 사용된다. 방법(2200)에 대한 일부 실시예들에서, 이웃하는 재구성된 샘플들은, 서브-블록의 경계의 부분이 현재 비디오 블록의 경계의 부분과 동일한 공간에 걸쳐 있지 않도록, 현재 비디오 블록 내에 위치한 서브-블록에 대해 사용된다. 방법(2200)에 대한 일부 실시예들에서, 복수의 서브-블록은 래스터-스캔 순서로 예측된다. 방법(2200)에 대한 일부 실시예들에서, 복수의 서브-블록은 지그재그 순서로 예측된다.
- [0784] 방법(2200)에 대한 일부 실시예들에서, 적어도 하나의 서브-블록의 폭 및 높이는 각각 최대 폭 및 최대 높이보다 크지 않다. 일부 실시예들에서, 방법(2200)은, 현재 비디오 블록의 폭 및 높이 중 임의의 하나 이상이 임계값 이상인 것에 응답하여, 현재 비디오 블록을 다수의 서브-블록으로 분할하는 단계를 추가로 포함한다. 방법(2200)에 대한 일부 실시예들에서, 임계값은 미리 정의된다. 방법(2200)에 대한 일부 실시예들에서, 임계값은 시퀀스 파라미터 세트(PS), 픽처 파라미터 세트(PPS), 픽처 헤더, 슬라이스 헤더, 타일 그룹 헤더 또는 타일

헤더에서 시그널링된다. 방법(2200)에 대한 일부 실시예들에서, 임계값은 현재 비디오 블록과 연관된 코딩된 정보에 기초한다. 방법(2200)에 대한 일부 실시예들에서, 코딩된 정보는 현재 비디오 블록의 블록 크기, 현재 비디오 블록의 픽처 타입, 또는 현재 비디오 블록의 시간 레이어 인덱스를 포함한다.

[0785] 도 23은 행렬-기반 인트라 예측을 위한 예시적인 방법(2300)의 예시적인 흐름도를 도시한다. 동작(2302)은 비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행하는 것을 포함하고, 전환은 전환 동안 행렬 가중 인트라 예측(MIP) 모드를 적용하기 전에 현재 비디오 블록의 이웃 샘플들을 필터링할지를 표시하는 규칙에 기초하고, MIP 모드는, 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 예측 블록을 결정하는 것을 포함한다.

[0786] 방법(2300)에 대한 일부 실시예들에서, 규칙은 이웃 샘플들이 MIP 모드에서 사용되기 전에 필터링되는 것을 표시한다. 방법(2300)에 대한 일부 실시예들에서, 규칙은 이웃 샘플들이 MIP 모드에서 사용되기 전에 필터링되지 않는 것을 표시한다. 방법(2300)에 대한 일부 실시예들에서, 규칙은, MIP 모드가 특정 값과 동일한 것에 응답하여 이웃 샘플들이 MIP 모드에서 사용되기 전에 필터링되는 것을 표시한다.

[0787] 본 문서에서, 용어 "비디오 처리" 또는 "변환"은 비디오 인코딩, 비디오 디코딩, 비디오 압축 또는 비디오 압축 해제에 지칭할 수 있다. 예를 들어, 비디오의 픽셀 표현으로부터 대응하는 비트스트림 표현으로 또는 그 반대로 전환하는 동안 비디오 압축 알고리즘들이 적용될 수 있다. 현재 비디오 블록의 비트스트림 표현은, 예를 들어, 선택스에 의해 정의된 바와 같이, 비트스트림 내의 상이한 위치들에 병치(co-locate)되거나 분산되는 비트들에 대응할 수 있다. 예를 들어, 매크로블록은 변환되고 코딩된 여러 잔차 값들의 관점에서 그리고 또한 헤더들 내의 비트들 및 비트스트림 내의 다른 필드들을 사용하여 인코딩될 수 있다. 게다가, 전환 동안, 디코더는, 위의 해결책들에 설명된 바와 같이, 결정에 기초하여, 일부 필드들이 존재하거나 존재하지 않을 수 있다는 것을 알고서 비트스트림을 파싱할 수 있다. 유사하게, 인코더는 특정 선택스 필드들이 포함되어야 하는지 여부를 결정할 수 있고, 그에 따라 코딩된 표현으로부터 선택스 필드들을 포함하거나 제외하는 것에 의해 코딩된 표현을 생성할 수 있다.

[0788] 진술한 것으로부터, 본 개시된 기술의 특정 실시예들은 예시를 위해 본 명세서에서 설명되었지만, 다양한 변경들이 본 발명의 범위로부터 벗어나지 않고 이루어질 수 있는 것이 이해될 것이다. 따라서, 본 개시된 기술은, 첨부된 청구범위에 의한 것을 제외하고는 제한되지 않는다.

[0789] 본 특허 문서에 설명된 기능적 동작들 및 주제의 구현들은 본 명세서에 개시된 구조들과 그 구조적 균등물들을 포함하는 다양한 시스템들, 디지털 전자 회로로, 또는 컴퓨터 소프트웨어, 펌웨어 또는 하드웨어로, 또는 그들 중 하나 이상의 조합으로 구현될 수 있다. 본 명세서에 설명된 주제의 구현들은 하나 이상의 컴퓨터 프로그램 제품으로서, 즉, 데이터 처리 장치에 의한 실행을 위해, 또는 데이터 처리 장치의 동작을 제어하기 위해 유형의 그리고 비-일시적 컴퓨터 판독가능 매체 상에 인코딩된 컴퓨터 프로그램 명령어들의 하나 이상의 모듈로서 구현될 수 있다. 컴퓨터 판독가능 매체는 머신 판독가능 저장 디바이스, 머신 판독가능 저장 기관, 메모리 디바이스, 머신 판독가능 전파 신호를 실현하는 조성물(composition of matter), 또는 이들 중 하나 이상의 조합일 수 있다. "데이터 처리 유닛" 또는 "데이터 처리 장치"라는 용어는 예로서, 프로그램가능 프로세서, 컴퓨터, 또는 다중 프로세서 또는 컴퓨터를 포함하는 데이터를 처리하기 위한 모든 장치들, 디바이스들 및 머신들을 포괄한다. 장치는 하드웨어 이외에도 대상(in question) 컴퓨터 프로그램용 실행 환경을 생성하는 코드, 예를 들어, 프로세서 펌웨어를 구성하는 코드, 프로토콜 스택, 데이터베이스 관리 시스템, 운영 체제, 또는 이들 중 하나 이상의 조합을 포함할 수 있다.

[0790] (프로그램, 소프트웨어, 소프트웨어 애플리케이션, 스크립트 또는 코드로도 알려진) 컴퓨터 프로그램은 컴파일되거나 해석된 언어를 포함하는 임의의 형태의 프로그래밍 언어로 기재될 수 있고, 이는 독립형(stand-alone) 프로그램 또는 모듈, 컴포넌트, 서브루틴 또는 컴퓨팅 환경의 사용에 적절한 기타 유닛을 포함하는 임의의 형태로 사용될 수 있다. 컴퓨터 프로그램이 파일 시스템에서의 파일에 반드시 대응하는 것은 아니다. 프로그램은 다른 프로그램들 또는 데이터(예를 들어, 마크업 언어 문서에 저장된 하나 이상의 스크립트)를 보유하는 파일의 일부분에, 대상(in question) 프로그램 전용인 단일 파일에, 또는 다수의 통합 파일(coordinated file)(예를 들어, 하나 이상의 모듈, 서브 프로그램(sub program)들, 또는 코드 부분(portion of code)들을 저장하는 파일들)에 저장될 수 있다. 컴퓨터 프로그램은 하나의 컴퓨터상에서, 또는 하나의 위치에 배치되어 있거나, 다수의 위치에 걸쳐 분산되어 통신 네트워크에 의해 상호접속되는 다수의 컴퓨터 상에서 실행되도록 배치될 수 있다.

[0791] 본 명세서에 설명된 프로세스들 및 로직 흐름들은, 입력 데이터에 대해 동작하고 출력을 생성함으로써 기능들을

수행하도록 하나 이상의 컴퓨터 프로그램을 실행하는 하나 이상의 프로그램가능한 프로세서에 의해 수행될 수 있다. 프로세스들 및 로직 흐름들은 또한, 특수 목적 로직 회로, 예컨대 FPGA(field programmable gate array) 또는 ASIC(application specific integrated circuit)에 의해 수행될 수 있으며, 장치는 또한, 특수 목적 로직 회로, 예컨대 FPGA 또는 ASIC로서 구현될 수 있다.

[0792] 컴퓨터 프로그램을 실행하는데 적합한 프로세서들은, 예를 들어, 범용 및 특수 목적 마이크로프로세서들 둘 다, 및 임의의 종류의 디지털 컴퓨터 중 임의의 하나 이상의 프로세서를 포함한다. 일반적으로, 프로세서는 ROM(read only memory) 또는 RAM(random access memory) 또는 둘 다로부터 명령어들과 데이터를 수신할 것이다. 컴퓨터의 필수 요소들은, 명령어들을 실행하기 위한 프로세서 및 명령어들과 데이터를 저장하기 위한 하나 이상의 메모리 디바이스이다. 일반적으로, 컴퓨터는 또한 데이터를 저장하기 위한 하나 이상의 대용량 저장 디바이스, 예를 들면, 자기, 자기 광학 디스크, 또는 광학 디스크를 포함할 것이거나, 또는 이들로부터 데이터를 수신하거나 이들로 데이터를 전송하도록 동작 가능하게 결합될 수 있거나, 또는 둘 모두일 것이다. 그러나, 컴퓨터가 이러한 디바이스들을 가질 필요는 없다. 컴퓨터 프로그램 명령어들과 데이터를 저장하기에 적합한 컴퓨터 판독가능 매체는 EPROM, EEPROM 및 플래시 메모리 디바이스들 등의 반도체 메모리 디바이스들을 예로서 포함하는 모든 형태의 비휘발성 메모리, 매체 및 메모리 디바이스들을 포함한다. 프로세서 및 메모리는 특수 목적 논리 회로에 의해 보완되거나 그에 포함될 수 있다.

[0793] 본 명세서는 도면과 함께 단지 예시적인 것으로 간주되며, 예시라는 말은 일 예를 의미하는 것으로 의도된다. 본 명세서에 사용된 바와 같이, "또는"의 사용은, 문맥이 명백하게 달리 나타내지 않는 한, "및/또는"을 포함하는 것으로 의도된다.

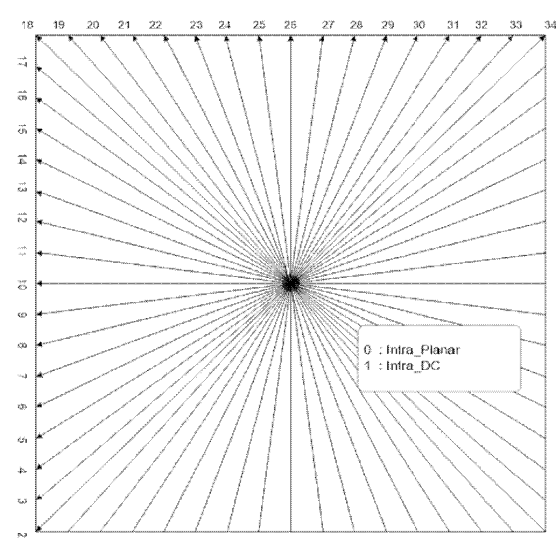
[0794] 본 특허 문서가 많은 특정한 사항들을 포함하지만, 이들은 본 발명의 범위의 또는 청구 범위에 관한 제한으로서 해석되어서는 안 되며, 오히려 특정 발명들의 특정 실시예들에 대해 특정적일 수 있는 특징들의 설명들로서 해석되어야 한다. 별개의 실시예들의 문맥에서 본 특허 문서에 설명되는 특정한 특징들이 또한 단일 실시예에서 조합하여 구현될 수 있다. 역으로, 단일 실시예의 문맥에서 설명된 여러 가지 특징이 또한 복수의 실시예에서 별개로 또는 임의의 적합한 하위 조합으로 실시될 수 있다. 또한, 앞서 특징들이 특정 조합들로 동작하는 것으로서 설명되고 심지어는 처음에는 이렇게 청구될 수 있지만, 어떤 경우들에서는 청구된 조합으로부터의 하나 이상의 특징이 조합으로부터 제외될(excised) 수 있으며, 청구된 조합은 부분조합 또는 부분조합의 변형에 대한 것일 수 있다.

[0795] 유사하게, 도면들에는 동작들이 특정 순서로 도시되지만, 이것은 바람직한 결과들을 달성하기 위해 그러한 동작들이 도시된 특정 순서로 또는 순차적으로 수행되어야 하거나, 모든 예시된 동작들이 수행되어야 하는 것을 요구하는 것으로 이해되어서는 안 된다. 또한, 본 특허 문서에서 설명된 실시예들에서의 다양한 시스템 컴포넌트들의 분리는 모든 실시예들에서 이러한 분리를 요구하는 것으로 이해되어서는 안 된다.

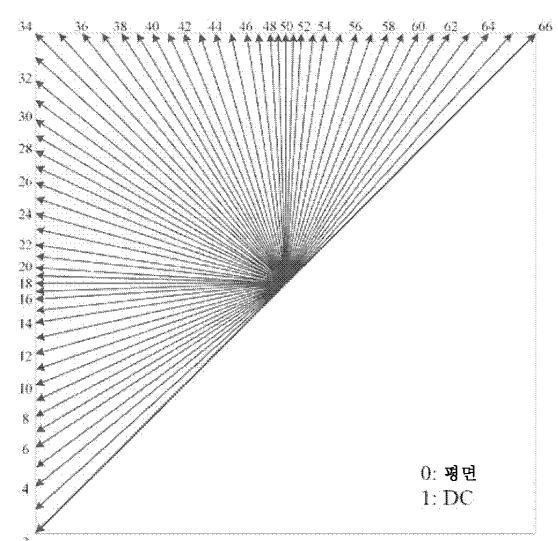
[0796] 단지 몇 개의 구현 및 예들만이 설명되었지만, 본 특허 문서에서 설명되고 예시된 내용에 기초하여 다른 구현들, 향상들 및 변형들이 이루어질 수 있다.

도면

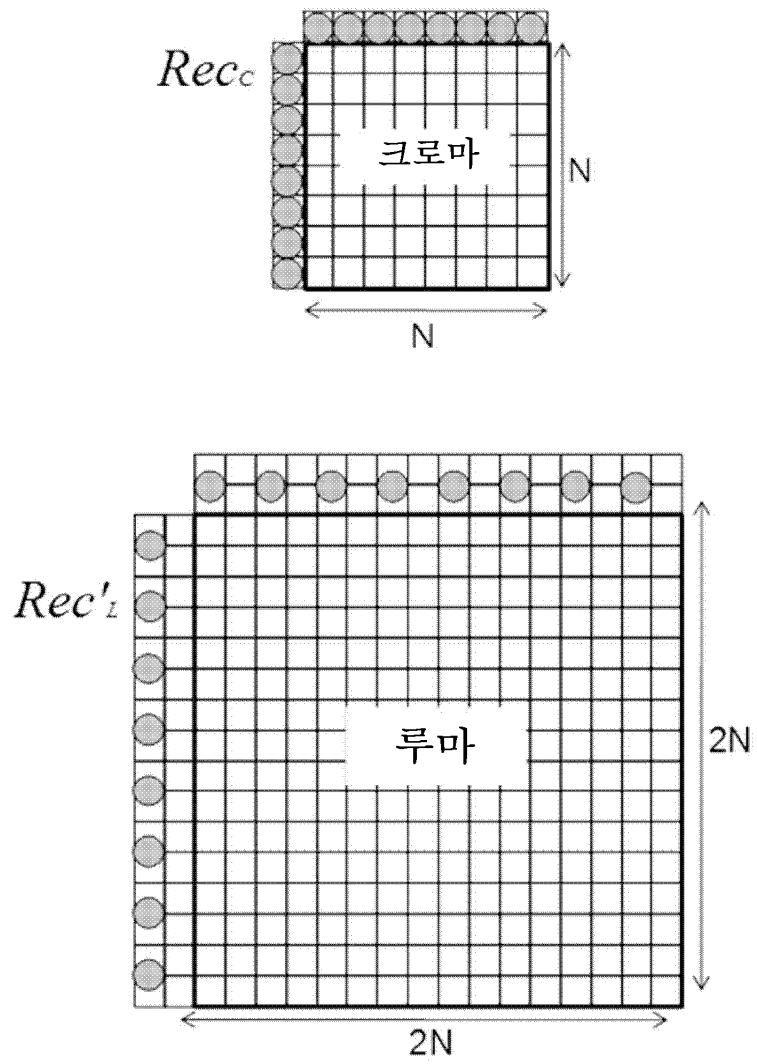
도면1



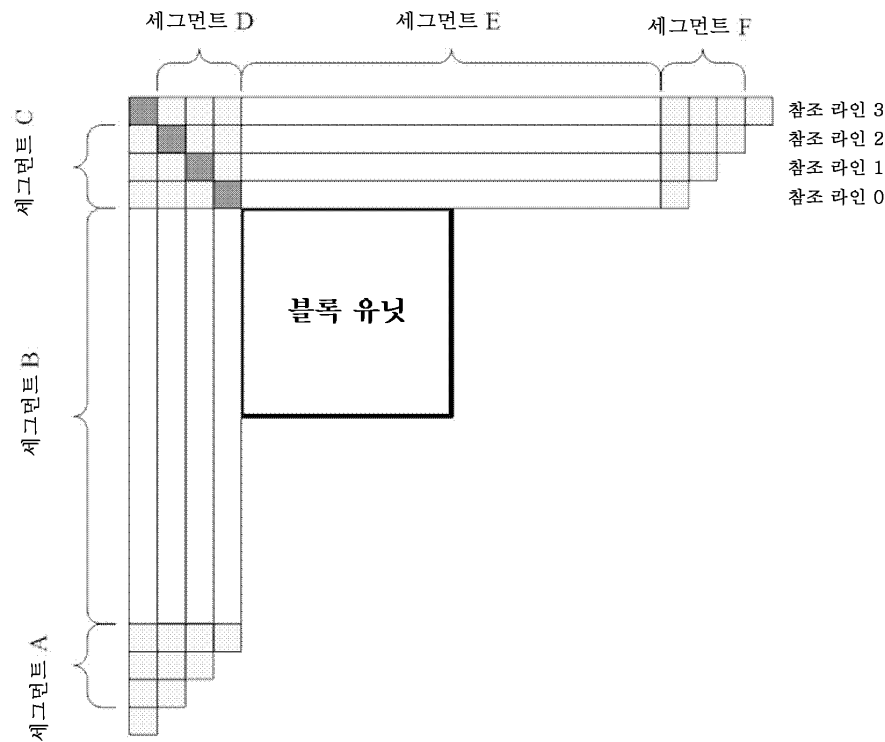
도면2



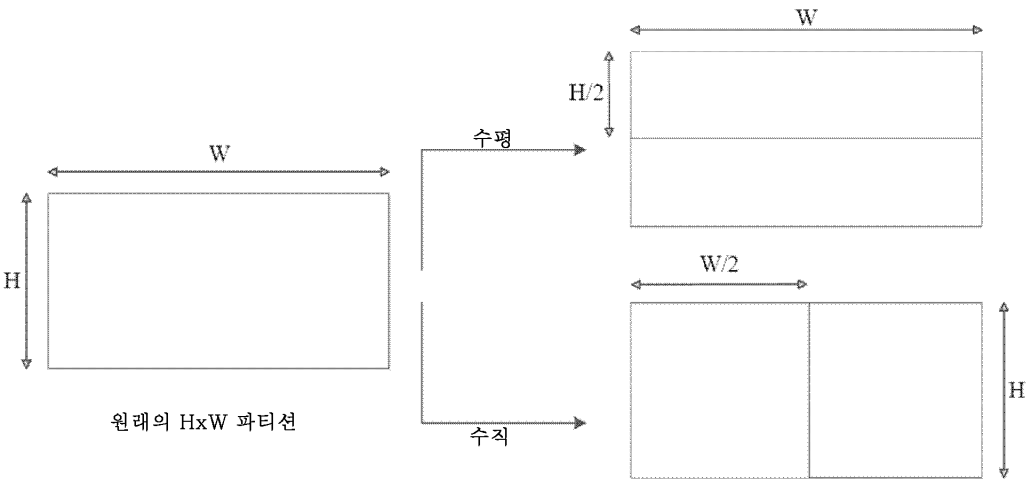
도면3



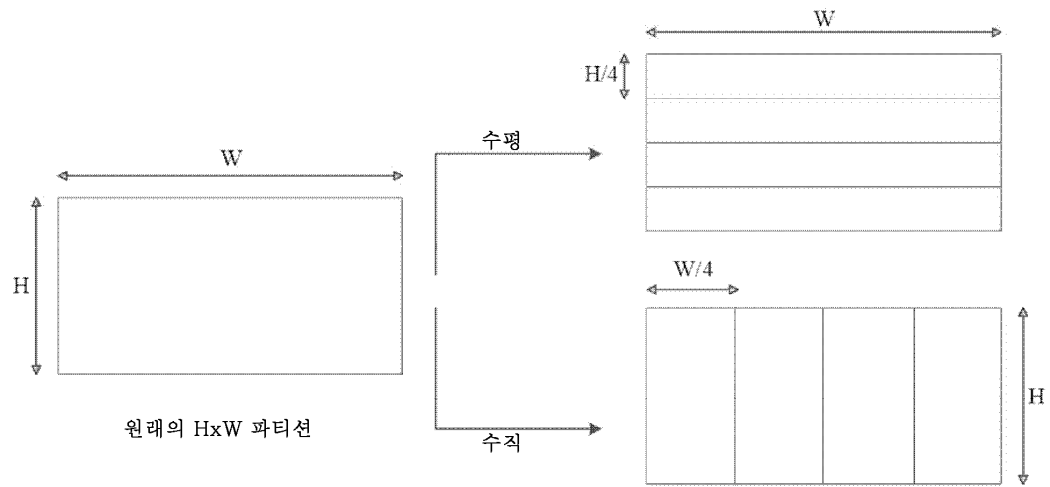
도면4



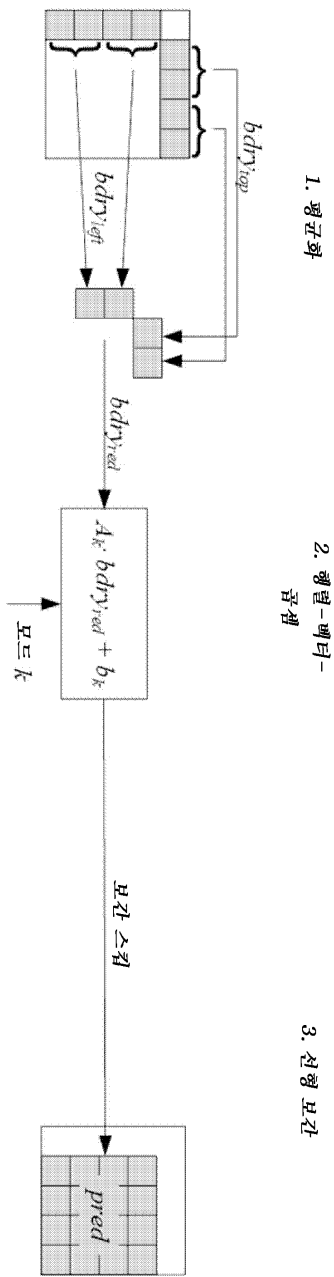
도면5a



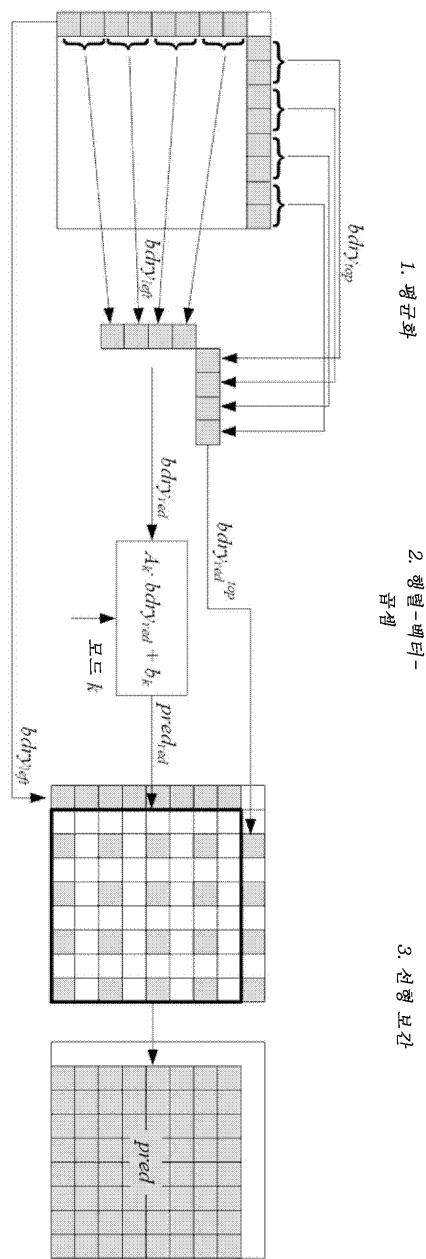
도면5b



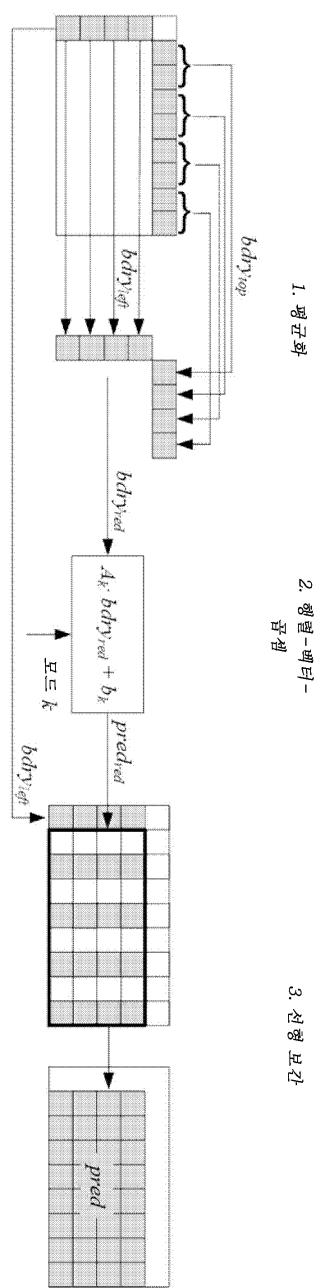
도면6



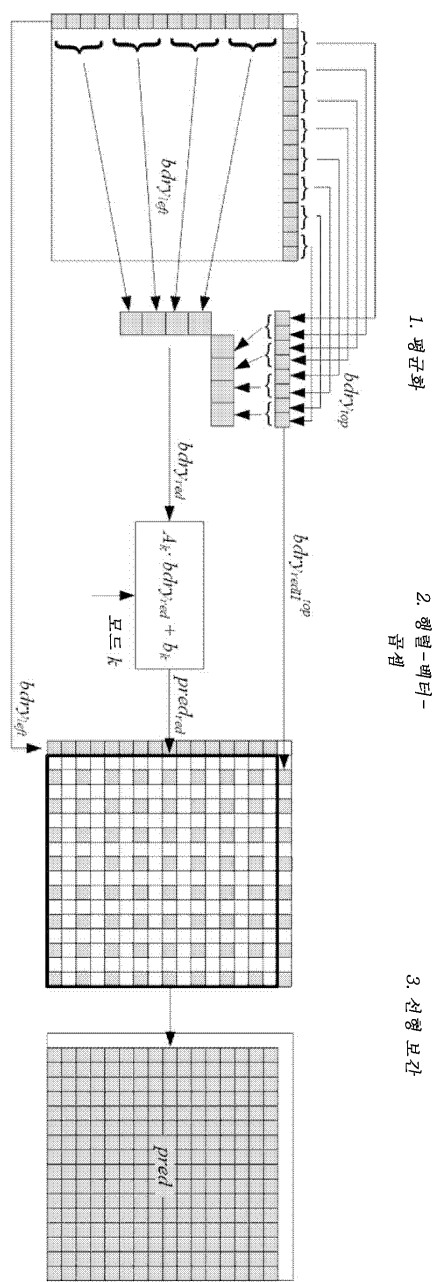
도면7



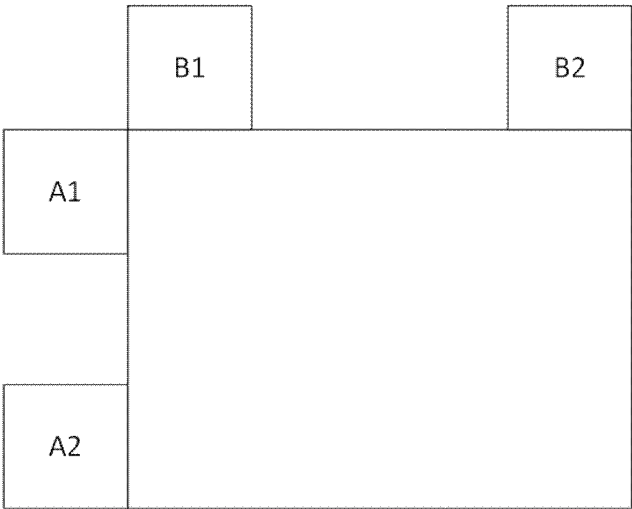
도면8



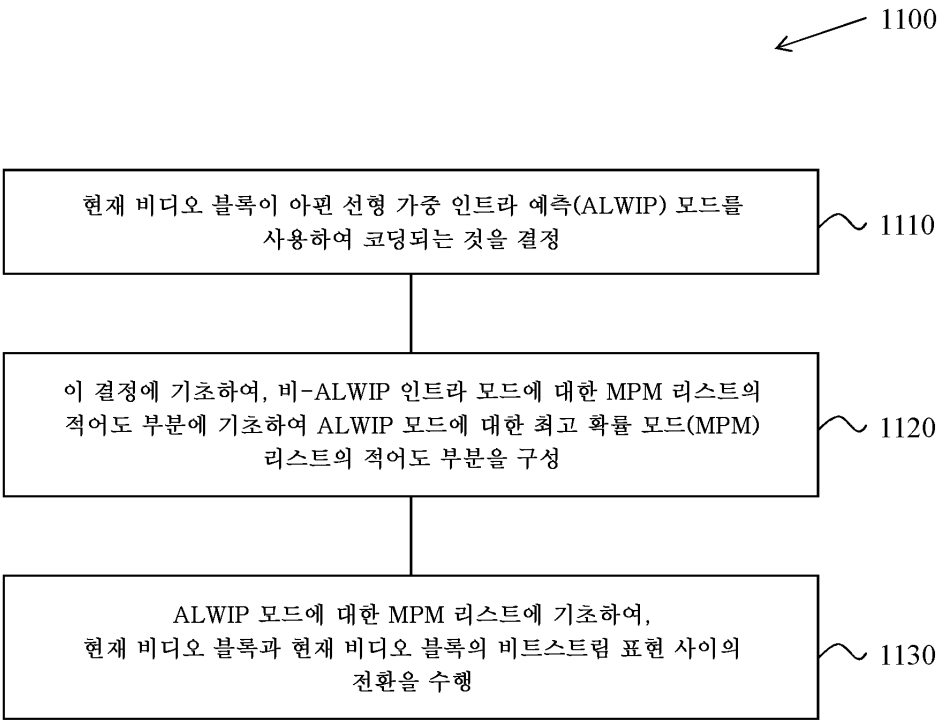
도면9



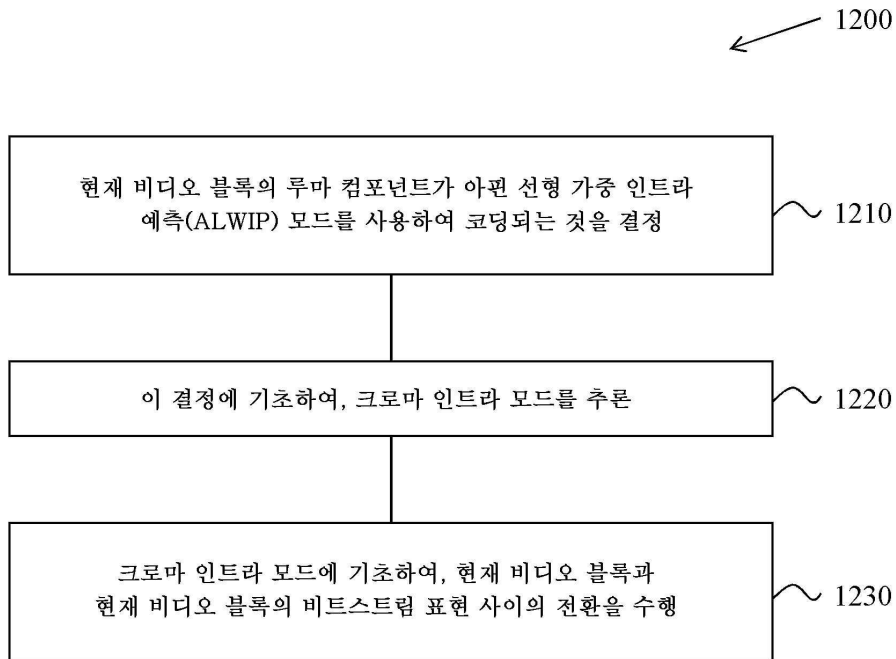
도면10



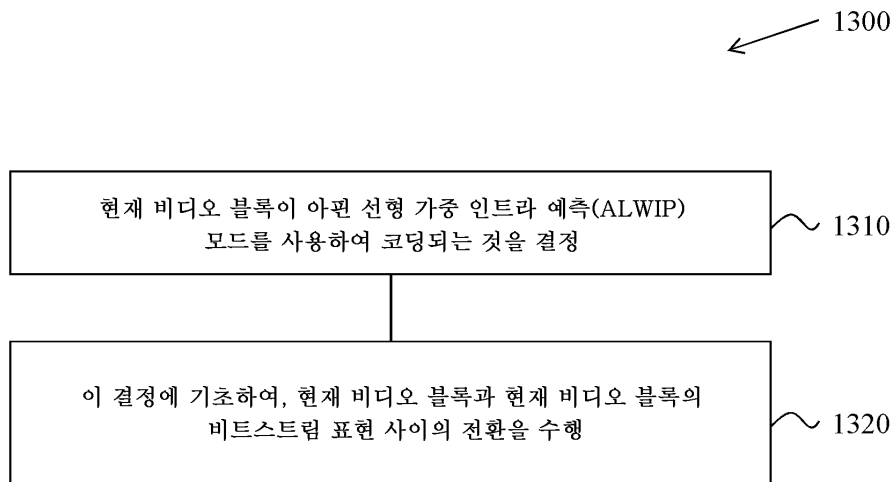
도면11



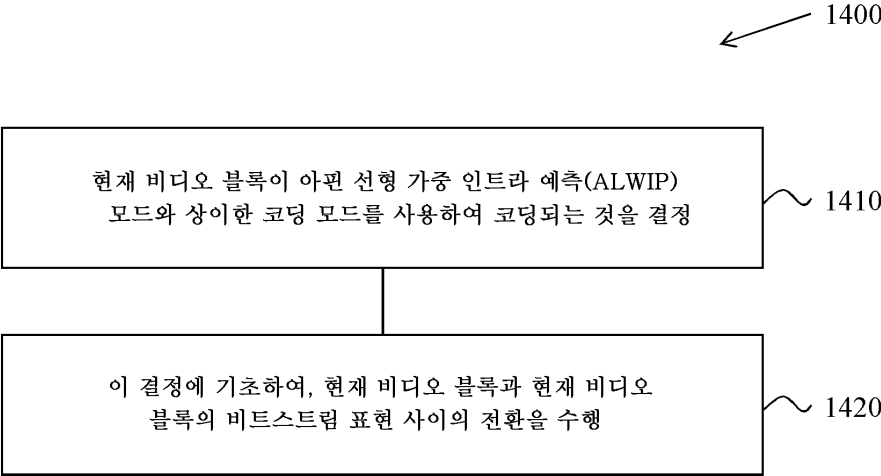
도면12



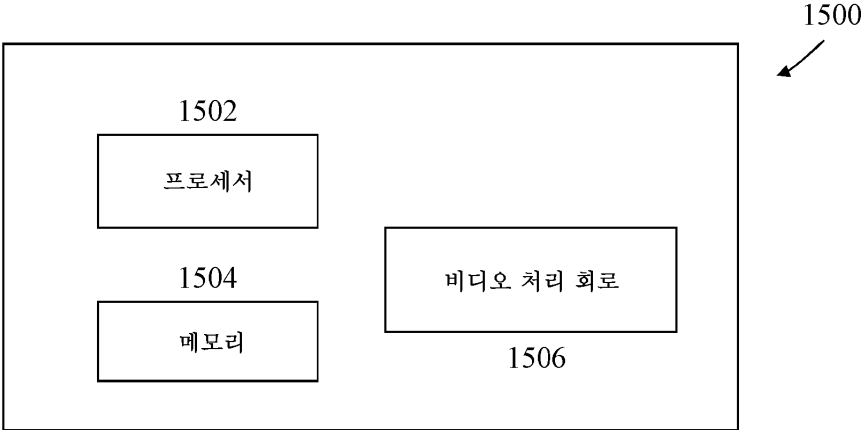
도면13



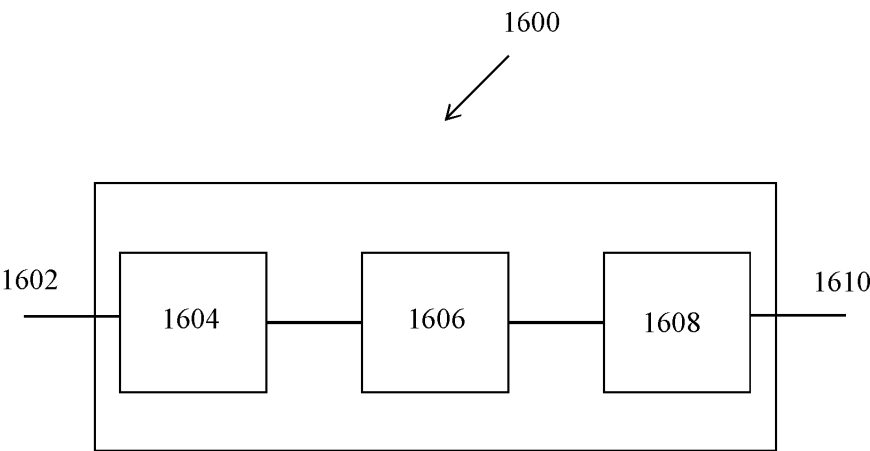
도면14



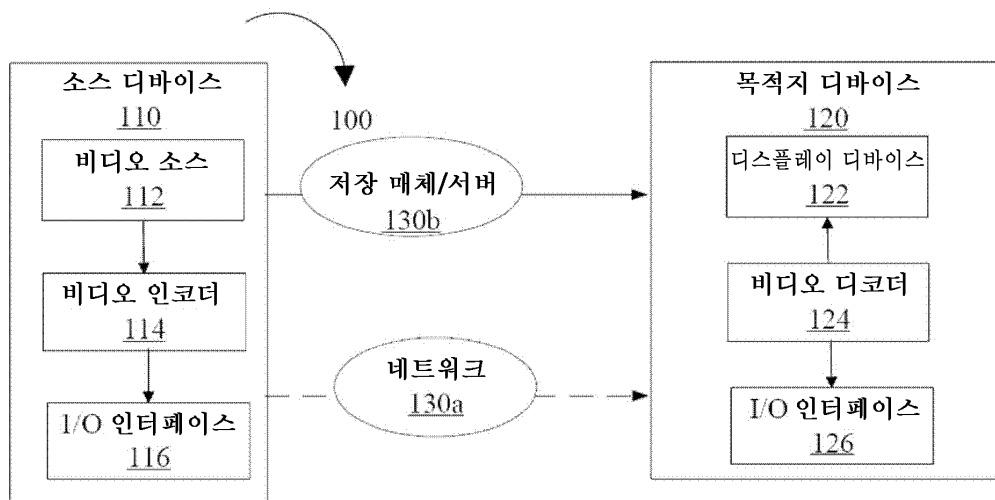
도면15



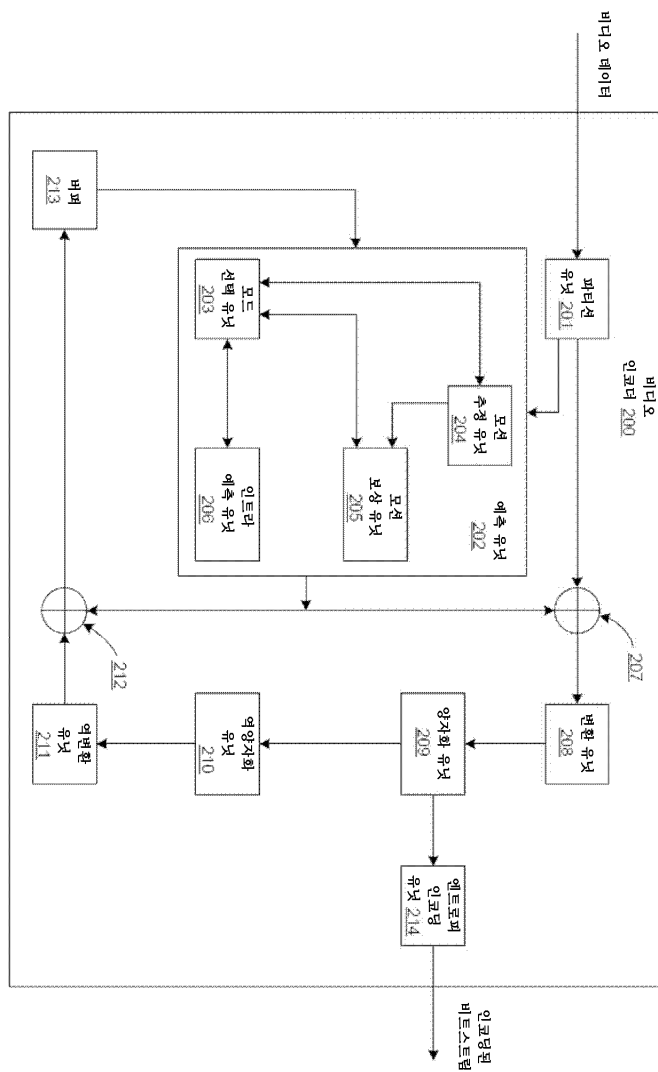
도면16



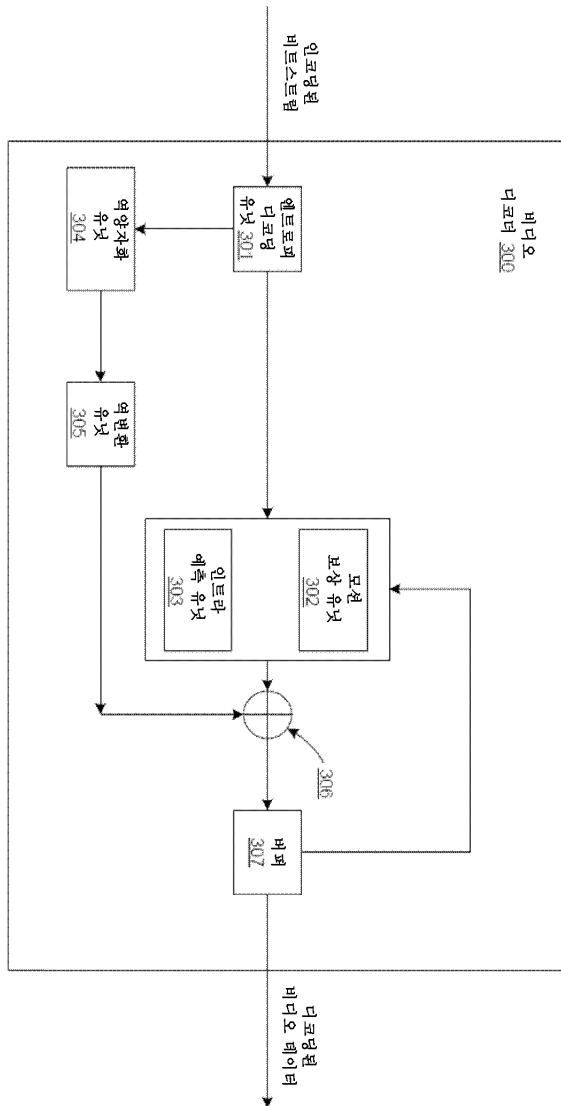
도면17



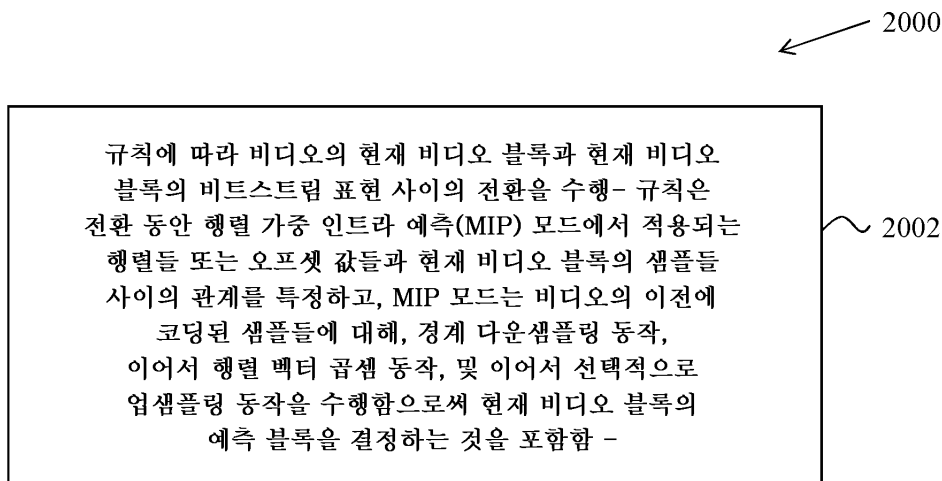
도면 18



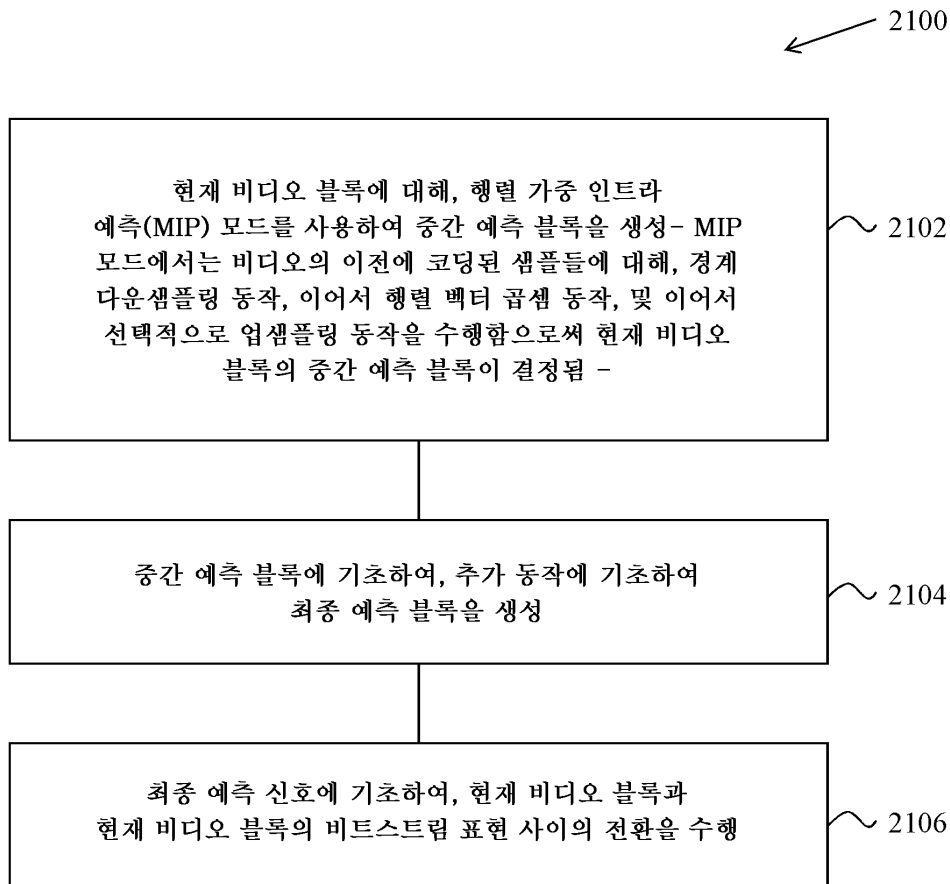
도면19



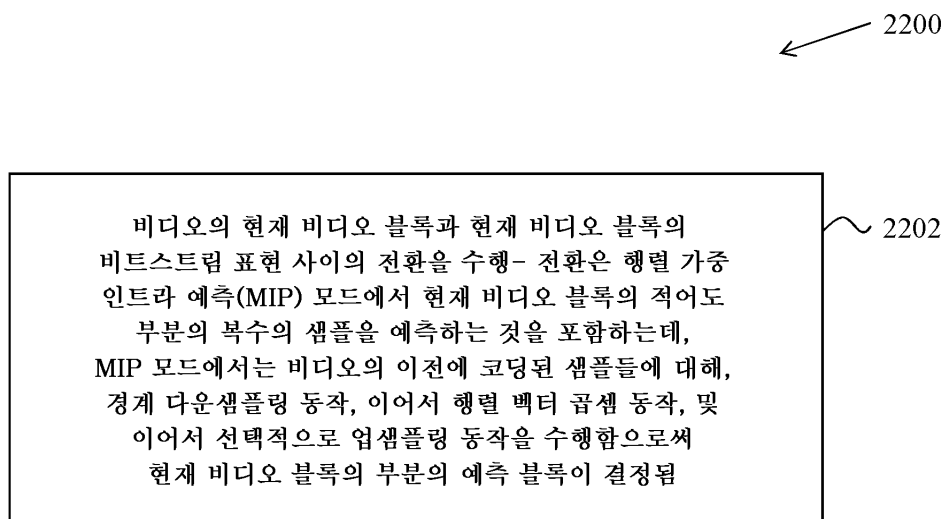
도면20



도면21



도면22



도면23

← 2300

2302

비디오의 현재 비디오 블록과 현재 비디오 블록의 비트스트림 표현 사이의 전환을 수행- 전환은 전환 동안 행렬 가중 인트라 예측(MIP) 모드를 적용하기 전에 현재 비디오 블록의 이웃 샘플들을 필터링할지를 표시하는 규칙에 기초하고, MIP 모드는, 비디오의 이전에 코딩된 샘플들에 대해, 경계 다운샘플링 동작, 이어서 행렬 벡터 곱셈 동작, 및 이어서 선택적으로 업샘플링 동작을 수행함으로써 현재 비디오 블록의 예측 블록을 결정하는 것을 포함함 -