



공개특허 10-2021-0142642



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0142642
(43) 공개일자 2021년11월25일

- (51) 국제특허분류(Int. Cl.)
H04N 19/61 (2014.01) *H04N 19/122* (2014.01)
H04N 19/159 (2014.01) *H04N 19/176* (2014.01)
H04N 19/625 (2014.01) *H04N 19/70* (2014.01)
H04N 19/91 (2014.01)
- (52) CPC특허분류
H04N 19/61 (2015.01)
H04N 19/122 (2015.01)
- (21) 출원번호 10-2021-7030566
- (22) 출원일자(국제) 2020년04월03일
심사청구일자 없음
- (85) 번역문제출일자 2021년09월23일
- (86) 국제출원번호 PCT/US2020/026620
- (87) 국제공개번호 WO 2020/206286
국제공개일자 2020년10월08일
- (30) 우선권주장
62/830,125 2019년04월05일 미국(US)
(뒷면에 계속)

- (71) 출원인
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 발명자
에길메즈 힐미 에네스
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
세레진 바딤
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(뒷면에 계속)
- (74) 대리인
특허법인코리아나

전체 청구항 수 : 총 39 항

(54) 발명의 명칭 **비디오 코딩을 위한 확장된 다중 변환 선택****(57) 요 약**

비디오 데이터를 코딩하기 위한 예시의 디바이스는 비디오 데이터를 저장하도록 구성된 메모리; 및 회로부에서 구현되는 하나 이상의 프로세서들을 포함하고, 하나 이상의 프로세서들은 비디오 데이터의 현재 블록에 대한 다중 변환 선택(MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 것으로서, 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하고, 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하며; 그리고 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하도록 구성된다. 제 2 코드워드는 저 주파수 분리불가능한 변환(LFNST) 신택스 엘리먼트에 대한 값일 수도 있다.

대 표 도 - 도9

인코더 측



디코더 측

(52) CPC특허분류

HO4N 19/159 (2015.01)

HO4N 19/176 (2015.01)

HO4N 19/625 (2015.01)

HO4N 19/70 (2015.01)

HO4N 19/91 (2015.01)

(72) 발명자

사이드 아미르

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

카르체비츠 마르타

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

(30) 우선권주장

62/855,398 2019년05월31일 미국(US)

16/838,553 2020년04월02일 미국(US)

명세서

청구범위

청구항 1

비디오 데이터를 코딩하는 방법으로서,

비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 단계로서, 상기 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 단계;

상기 가용 2차 변환들의 세트로부터의 상기 2차 변환을 나타내는 제 2 코드워드를 코딩하는 단계; 및

상기 현재 블록에 대한 잔차 데이터의 코딩 동안 상기 1차 변환 및 상기 2차 변환을 적용하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 변환 후보들의 세트는,

- 1) 수평 아이덴티티 변환 및 수직 아이덴티티 변환;
- 2) 이산 코사인 변환 (DCT)-2 수평 변환 및 DCT-2 수직 변환;
- 3) 이산 사인 변환 (DST)-7 수평 변환 및 DST-7 수직 변환;
- 4) DCT-8 수평 변환 및 상기 DST-7 수직 변환;
- 5) 상기 DST-7 수평 변환 및 DST-8 수직 변환;
- 6) 제 1 분리불가능한 변환; 및
- 7) 제 2 분리불가능한 변환

을 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 3

제 1 항에 있어서,

상기 변환 후보들의 세트는,

- 1) 수평 아이덴티티 변환 및 수직 아이덴티티 변환;
- 2) 이산 코사인 변환 (DCT)-2 수평 변환 및 DCT-2 수직 변환;
- 3) 이산 사인 변환 (DST)-7 수평 변환 및 DST-7 수직 변환;
- 4) DCT-8 수평 변환 및 상기 DST-7 수직 변환;
- 5) 상기 DST-7 수평 변환 및 DST-8 수직 변환; 및
- 6) 상기 가용 2차 변환들의 세트의 상기 2차 변환

을 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 4

제 1 항에 있어서,

상기 변환 후보들의 세트는 이산 코사인 변환 (DCT)-8 수평 변환 및 DCT-8 수직 변환의 조합을 제외하는, 비디

오 데이터를 코딩하는 방법.

청구항 5

제 1 항에 있어서,

상기 변환 후보들과 연관된 코드워드들은 허프만 (Huffman) 코드들을 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 6

제 1 항에 있어서,

상기 제 1 코드워드를 코딩하는 단계는 예측 모드 또는 상기 현재 블록의 사이즈 중 적어도 하나에 따라 상기 제 1 코드워드를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 7

제 6 항에 있어서,

상기 예측 모드 또는 상기 현재 블록의 사이즈 중 적어도 하나에 따라 변환 후보들의 상이한 세트들 또는 상기 변환 후보들에 대한 상이한 이진화들 중 적어도 하나를 각각 포함하는 MTS 스Kim들의 세트로부터 상기 MTS 스Kim을 선택하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 8

제 6 항에 있어서,

상기 예측 모드가 인트라-예측 모드인지 인터-예측 모드 인지에 따라 상기 MTS 스Kim을 선택하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 9

제 6 항에 있어서,

상기 현재 블록의 형상 또는 상기 현재 블록의 사이즈 중 적어도 하나에 따라 MTS 스Kim들의 세트로부터 상기 MTS 스Kim을 선택하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 10

제 9 항에 있어서,

상기 MTS 스Kim들의 세트는 제 1 사이즈의 블록들에 대한 제 1 수의 변환 후보들을 갖는 제 1 MTS 스Kim 및 제 2 사이즈의 블록들에 대한 제 2 수의 변환 후보들을 갖는 제 2 MTS 스Kim을 포함하고, 상기 제 2 수는 상기 제 1 수보다 더 크고 상기 제 2 사이즈는 상기 제 1 사이즈보다 더 큰, 비디오 데이터를 코딩하는 방법.

청구항 11

제 6 항에 있어서,

상기 현재 블록의 사이즈는 상기 현재 블록의 폭 또는 상기 현재 블록의 높이 중 적어도 하나에 따라 정의되는, 비디오 데이터를 코딩하는 방법.

청구항 12

제 6 항에 있어서,

상기 현재 블록의 형상은 정사각형 또는 직사각형으로서 정의되는, 비디오 데이터를 코딩하는 방법.

청구항 13

제 1 항에 있어서,

상기 현재 블록의 사이즈, 상기 현재 블록의 형상, 또는 상기 현재 블록에 대한 예측 모드 중 적어도 하나에 따

라 상기 제 1 코드워드를 코딩하기 위한 컨텍스트를 결정하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 14

제 1 항에 있어서,

상기 변환 후보들의 세트는 하나 이상의 분리가능한 변환 후보들 및 하나 이상의 분리불가능한 변환 후보들을 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 15

제 1 항에 있어서,

상기 제 2 코드워드는 저 주파수 분리불가능한 변환 (LFNST) 선택스 엘리먼트에 대한 값을 포함하고, 상기 방법은,

상기 LFNST 선택스 엘리먼트에 대한 값이 상기 선택된 변환 스킴에 따라 코딩된다고 결정하는 단계를 더 포함하고,

상기 LFNST 선택스 엘리먼트에 대한 값을 코딩하는 단계는 상기 LFNST 선택스 엘리먼트에 대한 값이 코딩된다고 결정하는 것에 응답하여 상기 LFNST 선택스 엘리먼트에 대한 값을 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 16

제 15 항에 있어서,

상기 제 1 코드워드는 MTS 선택스 엘리먼트를 포함하고, 상기 LFNST 선택스 엘리먼트에 대한 값이 코딩된다고 결정하는 단계는 상기 LFNST 선택스 엘리먼트에 대한 값이 상기 제 1 코드워드에 따라 코딩된다고 결정하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 17

제 15 항에 있어서,

상기 LFNST 선택스 엘리먼트에 대한 값이 코딩된다고 결정하는 단계는, 상기 선택된 변환 스킴이 이산 코사인 변환 (DCT)-2 수평 변환 및 DCT-2 수직 변환을 포함하는지 여부에 따라 상기 LFNST 선택스 엘리먼트에 대한 값이 코딩된다고 결정하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 18

제 1 항에 있어서,

상기 제 1 코드워드를 코딩하는 단계는 상기 제 1 코드워드를 디코딩하는 단계를 포함하고;

상기 제 2 코드워드를 코딩하는 단계는 상기 제 2 코드워드를 디코딩하는 단계를 포함하고; 그리고

상기 1차 변환 및 상기 2차 변환을 적용하는 단계는,

중간 변환 계수들을 생성하기 위해 디코딩된 변환 계수들에 상기 2차 변환을 적용하는 단계; 및

상기 현재 블록에 대한 잔차 블록을 생성하기 위해 상기 중간 변환 계수들에 상기 1차 변환을 적용하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 19

제 1 항에 있어서,

상기 제 1 코드워드를 코딩하는 단계는 상기 제 1 코드워드를 인코딩하는 단계를 포함하고;

상기 제 2 코드워드를 코딩하는 단계는 상기 제 2 코드워드를 인코딩하는 단계를 포함하고; 그리고

상기 1차 변환 및 상기 2차 변환을 적용하는 단계는,

중간 변환 계수들을 생성하기 위해 상기 현재 블록에 대한 잔차 블록에 상기 1차 변환을 적용하는 단계; 및
상기 중간 변환 계수들에 상기 2차 변환을 적용하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 20

비디오 데이터를 코딩하기 위한 디바이스로서,
비디오 데이터를 저장하도록 구성된 메모리; 및
회로부에서 구현되는 하나 이상의 프로세서들을 포함하고,
상기 하나 이상의 프로세서들은,
비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 것으로서, 상기 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하고;
상기 가용 2차 변환들의 세트로부터의 상기 2차 변환을 나타내는 제 2 코드워드를 코딩하며; 그리고
상기 현재 블록에 대한 잔차 데이터의 코딩 동안 상기 1차 변환 및 상기 2차 변환을 적용하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 21

제 20 항에 있어서,
상기 변환 후보들의 세트는,
 1) 수평 아이덴티티 변환 및 수직 아이덴티티 변환;
 2) 이산 코사인 변환 (DCT)-2 수평 변환 및 DCT-2 수직 변환;
 3) 이산 사인 변환 (DST)-7 수평 변환 및 DST-7 수직 변환;
 4) DCT-8 수평 변환 및 상기 DST-7 수직 변환;
 5) 상기 DST-7 수평 변환 및 DST-8 수직 변환;
 6) 제 1 분리불가능한 변환; 및
 7) 제 2 분리불가능한 변환
을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 22

제 20 항에 있어서,
상기 변환 후보들의 세트는,
 1) 수평 아이덴티티 변환 및 수직 아이덴티티 변환;
 2) 이산 코사인 변환 (DCT)-2 수평 변환 및 DCT-2 수직 변환;
 3) 이산 사인 변환 (DST)-7 수평 변환 및 DST-7 수직 변환;
 4) DCT-8 수평 변환 및 상기 DST-7 수직 변환;
 5) 상기 DST-7 수평 변환 및 DST-8 수직 변환; 및
 6) 상기 가용 2차 변환들의 세트의 상기 2차 변환
을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 23

제 20 항에 있어서,

상기 변환 후보들의 세트는 이산 코사인 변환 (DCT)-8 수평 변환 및 DCT-8 수평 변환의 조합을 제외하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 24

제 20 항에 있어서,

상기 하나 이상의 프로세서들은 예측 모드 또는 상기 현재 블록의 사이즈 중 적어도 하나에 따라 상기 제 1 코드워드를 코딩하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 25

제 20 항에 있어서,

상기 하나 이상의 프로세서들은 또한 상기 현재 블록의 사이즈, 상기 현재 블록의 형상, 또는 상기 현재 블록에 대한 예측 모드 중 적어도 하나에 따라 상기 제 1 코드워드를 코딩하기 위한 컨텍스트를 결정하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 26

제 20 항에 있어서,

상기 제 2 코드워드는 저 주파수 분리불가능한 변환 (LFNST) 선택스 엘리먼트에 대한 값을 포함하고, 상기 하나 이상의 프로세서들은 또한,

상기 LFNST 선택스 엘리먼트에 대한 값이 상기 선택된 변환 스킵에 따라 코딩된다고 결정하도록 구성되고,

상기 하나 이상의 프로세서들은 상기 LFNST 선택스 엘리먼트에 대한 값이 코딩된다고 결정하는 것에 응답하여 상기 LFNST 선택스 엘리먼트에 대한 값을 코딩하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 27

제 26 항에 있어서,

상기 제 1 코드워드는 MTS 선택스 엘리먼트를 포함하고, 상기 하나 이상의 프로세서들은 상기 LFNST 선택스 엘리먼트에 대한 값이 상기 제 1 코드워드에 따라 코딩되는지 여부를 결정하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 28

제 26 항에 있어서,

상기 하나 이상의 프로세서들은 상기 선택된 변환 스킵이 이산 코사인 변환 (DCT)-2 수평 변환 및 DCT-2 수직 변환을 포함하는지 여부에 따라 상기 LFNST 선택스 엘리먼트에 대한 값이 코딩된다고 결정하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 29

제 20 항에 있어서,

상기 디바이스는 비디오 디코더를 포함하고, 상기 하나 이상의 프로세서들은

상기 제 1 코드워드를 디코딩하고;

상기 제 2 코드워드를 디코딩하고;

중간 변환 계수들을 생성하기 위해 디코딩된 변환 계수들에 상기 2차 변환을 적용하며; 그리고

상기 현재 블록에 대한 잔차 블록을 생성하기 위해 상기 중간 변환 계수들에 상기 1차 변환을 적용하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 30

제 20 항에 있어서,

상기 디바이스는 비디오 인코더를 포함하고, 상기 하나 이상의 프로세서들은

상기 제 1 코드워드를 인코딩하고;

상기 제 2 코드워드를 인코딩하고;

중간 변환 계수들을 생성하기 위해 상기 현재 블록에 대한 잔차 블록에 상기 1차 변환을 적용하며; 그리고

상기 중간 변환 계수들에 상기 2차 변환을 적용하도록 구성되는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 31

제 20 항에 있어서,

디코딩된 비디오 데이터를 디스플레이하도록 구성된 디스플레이를 더 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 32

제 20 항에 있어서,

상기 디바이스는 카메라, 컴퓨터, 모바일 디바이스, 브로드캐스트 수신기 디바이스 또는 셋톱 박스 중 하나 이상을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 33

제 20 항에 있어서,

상기 디바이스는,

집적 회로;

마이크로프로세서; 또는

무선 통신 디바이스

중 적어도 하나를 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 34

비디오 데이터를 코딩하기 위한 디바이스로서,

비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 수단으로서, 상기 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 수단;

상기 가용 2차 변환들의 세트로부터의 상기 2차 변환을 나타내는 제 2 코드워드를 코딩하는 수단; 및

상기 현재 블록에 대한 잔차 데이터의 코딩 동안 상기 1차 변환 및 상기 2차 변환을 적용하는 수단을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 35

제 34 항에 있어서,

상기 제 2 코드워드는 저 주파수 분리불가능한 변환 (LFNST) 싱택스 엘리먼트에 대한 값을 포함하고,

상기 LFNST 싱택스 엘리먼트에 대한 값이 상기 선택된 변환 스킵에 따라 코딩된다고 결정하는 수단을 더 포함하고,

상기 LFNST 싱택스 엘리먼트에 대한 값을 코딩하는 수단은 상기 LFNST 싱택스 엘리먼트에 대한 값이 코딩된다고 결정하는 것에 응답하여 상기 LFNST 싱택스 엘리먼트에 대한 값을 코딩하는 수단을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 36

제 34 항에 있어서,

상기 디바이스는 비디오 디코더를 포함하고;

상기 제 1 코드워드를 코딩하는 수단은 상기 제 1 코드워드를 디코딩하는 수단을 포함하고;

상기 제 2 코드워드를 코딩하는 수단은 상기 제 2 코드워드를 디코딩하는 수단을 포함하고; 그리고

상기 1차 변환 및 상기 2차 변환을 적용하는 수단은,

중간 변환 계수들을 생성하기 위해 디코딩된 변환 계수들에 상기 2차 변환을 적용하는 수단; 및

상기 현재 블록에 대한 잔차 블록을 생성하기 위해 상기 중간 변환 계수들에 상기 1차 변환을 적용하는 수단을 포함하는, 비디오 데이터를 코딩하기 위한 디바이스.

청구항 37

명령들이 저장된 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 실행될 때, 비디오 데이터를 코딩하기 위한 디바이스의 프로세서로 하여금,

비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하게 하는 것으로서, 상기 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하게 하고;

상기 가용 2차 변환들의 세트로부터의 상기 2차 변환을 나타내는 제 2 코드워드를 코딩하게 하며; 그리고

상기 현재 블록에 대한 잔차 데이터의 코딩 동안 상기 1차 변환 및 상기 2차 변환을 적용하게 하는, 컴퓨터 판독가능 저장 매체.

청구항 38

제 37 항에 있어서,

상기 제 2 코드워드는 저 주파수 분리불가능한 변환 (LFNST) 신택스 엘리먼트에 대한 값을 포함하고, 상기 프로세서로 하여금,

상기 LFNST 신택스 엘리먼트에 대한 값이 상기 선택된 변환 스킵에 따라 코딩된다고 결정하게 하는 명령들을 더 포함하고,

상기 프로세서로 하여금 상기 LFNST 신택스 엘리먼트에 대한 값을 코딩하게 하는 명령들은 상기 프로세서로 하여금 상기 LFNST 신택스 엘리먼트에 대한 값이 코딩된다고 결정하는 것에 응답하여 상기 LFNST 신택스 엘리먼트에 대한 값을 코딩하게 하는 명령들을 포함하는, 컴퓨터 판독가능 저장 매체.

청구항 39

제 37 항에 있어서,

상기 프로세서로 하여금 상기 제 1 코드워드를 코딩하게 하는 명령들은 상기 프로세서로 하여금 상기 제 1 코드워드를 디코딩하게 하는 명령들을 포함하고,

상기 프로세서로 하여금 상기 제 2 코드워드를 코딩하게 하는 명령들은 상기 프로세서로 하여금 상기 제 2 코드워드를 디코딩하게 하는 명령들을 포함하고; 그리고

상기 프로세서로 하여금 상기 1차 변환 및 상기 2차 변환을 적용하게 하는 명령들은, 상기 프로세서로 하여금,

중간 변환 계수들을 생성하기 위해 디코딩된 변환 계수들에 상기 2차 변환을 적용하게 하며; 그리고

상기 현재 블록에 대한 잔차 블록을 생성하기 위해 상기 중간 변환 계수들에 상기 1차 변환을 적용하게 하는 명령들을 포함하는, 컴퓨터 판독가능 저장 매체.

발명의 설명

기술 분야

[0001] 본 출원은 2020년 4월 2일 출원된 미국 출원 제 16/838,553 호, 2019년 4월 5일 출원된 미국 가출원 제 62/830,125 호, 및 2019년 5월 31일 출원된 미국 가출원 제 62/855,398 호의 이익을 주장하며, 이 출원들의 전체 내용들은 본 명세서에 참조로 통합된다.

[0002] 본 개시는 비디오 인코딩 및 비디오 디코딩을 포함하는, 비디오 코딩에 관한 것이다.

배경기술

[0003] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 다이렉트 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인용 디지털 보조기 (PDA) 들, 랩탑 또는 테스크탑 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 이른바 "스마트 폰들", 비디오 멜레컨퍼런싱 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, 어드밴스드 비디오 코딩 (AVC), 고효율 비디오 코딩 (High Efficiency Video Coding; HEVC) 표준, ITU-T H.265/고효율 비디오 코딩 (HEVC), 및 이러한 표준들의 확장들에 의해 정의된 표준들에 기재된 것들과 같은, 비디오 코딩 기법들을 구현한다. 비디오 디바이스들은 그러한 비디오 코딩 기법들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0004] 비디오 코딩 기법들은 비디오 시퀀스들에 내재된 리던던시를 감소 또는 제거하기 위해 공간적 (인트라-픽처) 예측 및/또는 시간적 (인터-픽처) 예측을 포함한다. 블록 기반 비디오 코딩에 대해, 비디오 슬라이스 (즉, 비디오 픽처 또는 비디오 픽처의 일부) 는 비디오 블록들로 파티셔닝될 수도 있으며, 이 비디오 블록들은 또한 코딩 트리 유닛들 (CTU들), 코딩 유닛들 (CU들) 및/또는 코딩 노드들로서 지칭될 수도 있다. 픽처의 인트라-코딩된 (I) 슬라이스에서의 비디오 블록들은 동일한 픽처에 있어서 이웃하는 블록들에서의 참조 샘플들에 대한 공간적 예측을 사용하여 인코딩된다. 픽처의 인터-코딩된 (P 또는 B) 슬라이스에서의 비디오 블록들은 동일한 픽처에 있어서 이웃하는 블록들에서의 참조 샘플들에 대한 공간적 예측 또는 다른 참조 픽처들에서의 참조 샘플들에 대한 시간적 예측을 이용할 수도 있다. 픽처들은 프레임들로 지칭될 수도 있고, 참조 픽처들은 참조 프레임들로 지칭될 수도 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0005] 일반적으로, 본 개시는 비디오 코딩에서 변환 코딩에 관련된 기법들을 설명한다. 변환 코딩은 최신 비디오 압축 표준의 중요한 엘리먼트이다. 본 개시는 다기능 비디오 코딩 (Versatile Video Coding; VVC) / ITU-T H.266 의 것들과 같은, 다른 다중 변환 선택 (multiple transform selection; MTS) 틀들을 확장하는 MTS 설계를 설명한다. 본 개시에 설명된 설계는 인코더가 더 많은 변환 후보들로부터 변환을 선정할 수 있게 하기 때문에, 이 기법들은 코딩 효율을 개선할 수 있다. 본 개시는 또한 코딩 효율에서 현저한 열화없이 인코더 및 디코더 복잡성을 감소시킬 수 있는 저 주파수 분리불가능한 변환 (Low-Frequency Non-separable Transformation; LFNST) 의 다양한 단순화된 버전들을 설명한다. 따라서, 이 기법들은 어드밴스드 비디오 코덱 및 차세대 비디오 코딩 표준들, 예컨대 VVC에서 사용될 수도 있다.

[0006] 일 예에서, 비디오 데이터를 코딩 (인코딩 또는 디코딩) 하는 방법은, 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 단계로서, 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 단계; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하는 단계; 및 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하는 단계를 포함한다.

[0007] 다른 예에서, 비디오 데이터를 코딩하기 위한 디바이스는 비디오 데이터를 저장하도록 구성된 메모리; 및 회로부에서 구현되는 하나 이상의 프로세서들을 포함하고, 하나 이상의 프로세서들은 비디오 데이터의 현재 블록에

대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스Kim을 나타내는 제 1 코드워드를 코딩하는 것으로서, 선택된 변환 스Kim은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하고, 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하며; 그리고 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하도록 구성된다.

[0008] 다른 예에서, 비디오 데이터를 코딩하기 위한 디바이스는, 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스Kim의 변환 후보들의 세트의 선택된 변환 스Kim을 나타내는 제 1 코드워드를 코딩하는 수단으로서, 선택된 변환 스Kim은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 수단; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하는 수단; 및 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하는 수단을 포함한다.

[0009] 다른 예에서, 명령들이 저장된 컴퓨터 판독가능 저장 매체로서, 명령들은 실행될 때, 프로세서로 하여금, 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스Kim의 변환 후보들의 세트의 선택된 변환 스Kim을 나타내는 제 1 코드워드를 코딩하게 하는 것으로서, 선택된 변환 스Kim은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하게 하고; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하게 하며; 그리고 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하게 한다.

[0010] 하나 이상의 예들의 상세들은 첨부 도면들 및 하기의 설명에서 기술된다. 다른 과정들, 목적들 및 이점들은 상세한 설명 및 도면들, 그리고 청구항들로부터 명백해질 것이다.

도면의 간단한 설명

[0011] 도 1 은 본 개시의 기법들을 수행할 수도 있는 예시의 비디오 인코딩 및 디코딩 시스템을 도시하는 블록 다이어그램이다.

도 2a 및 도 2b 는 예시의 퀘드트리 이진 트리 (QTBT) 구조 및 대응하는 코딩 트리 유닛 (CTU) 을 도시하는 개념적 다이어그램들이다.

도 3a 및 도 3b 는 고효율 비디오 코딩 (HEVC) 의 잔차 퀘드트리에 기초한 예시의 변환 스Kim을 도시하는 개념적 다이어그램들이다.

도 4 는 적응적 변환 선택으로 하이브리드 비디오 인코딩을 위한 예시의 시스템을 도시하는 블록 다이어그램이다.

도 5a 및 도 5b 는 별도의 변환 구현으로서 수평 및 수직 변환들을 도시하는 개념적 다이어그램이다.

도 6 은 2 개의 변환들을 식별하는데 사용된 다중 변환 선택 (MTS) 시그널링의 예를 나타내는 개념적 다이어그램이다.

도 7 은 일 예의 변환 할당 및 대응하는 단항 코드워드를 도시하는 개념적 다이어그램이다.

도 8 은 2차 변환들을 지원하는 일 예의 MTS 설계를 도시하는 개념적 다이어그램이다.

도 9 는 비디오 코더 (비디오 인코더 또는 비디오 디코더) 가 적용할 수도 있는 저 주파수 분리불가능한 변환들 (LFNST) 의 예들을 도시하는 개념적 다이어그램이다.

도 10 은 HxW 블록의 (상단-좌측 부분에서) 계수들의 서브세트에 적용된 LFNST 의 예를 도시하는 개념적 다이어그램이다.

도 11a 및 도 11b 는 예시의 2-단계 LFNST 프로세스 구현을 도시하는 개념적 다이어그램들이다.

도 12 는 본 개시의 기법들을 수행할 수도 있는 예시의 비디오 인코더를 도시하는 블록 다이어그램이다.

도 13 은 본 개시의 기법들을 수행할 수도 있는 예시의 비디오 디코더를 도시하는 블록 다이어그램이다.

도 14 는 본 개시의 기법들에 따른 현재 블록을 인코딩하기 위한 예시의 방법을 도시하는 플로우차트이다.

도 15 는 본 개시의 기법들에 따른 비디오 데이터의 현재 블록을 디코딩하기 위한 예시의 방법을 도시하는 플로우차트이다.

도 16 은 본 개시의 기법들에 따른 예시의 비디오 인코딩 방법을 도시하는 플로우차트이다.

도 17 은 본 개시의 기법들에 따른 예시의 비디오 디코딩 방법을 도시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0012] 본 개시는 예를 들어, M. Wien, High Efficiency Video Coding: Coding Tools and Specification, Springer-Verlag, Berlin, 2015 에서 논의된 바와 같이, 최신 비디오 압축 표준의 중요한 요소인, 변환 코딩에 관련된 기법들을 설명한다. 본 개시는 확장된 다중 변환 선택 (MTS) 기법들을 설명한다.

[0013] 일반적으로, 비디오 데이터는 팩터들의 시퀀스 시리즈로서 표현된다. 비디오 코더는 팩터들을 블록들로 패티셔닝하고 블록들의 각각을 코딩한다. 코딩은 일반적으로 예측 및 잔차 코딩을 포함한다. 예측 동안, 비디오 코더는 인트라-예측 (예측 블록이 동일한 팩터의 이웃하는, 이전에 코딩된 블록들로부터 형성됨) 또는 인터-예측 (예측 블록이 이전에 코딩된 팩터들의 이전에 코딩된 블록들로부터 형성됨) 을 사용하여 예측 블록을 형성할 수도 있다. 잔차 블록은 예측 블록과 원래의, 코딩되지 않은 블록 사이의 팩셀별 차이를 나타낸다.

비디오 인코더는 변환 계수들을 포함하는 변환 블록을 생성하기 위해 잔차 블록에 변환을 적용할 수도 있는 반면, 비디오 디코더는 잔차 블록의 버전을 재생하기 위해 변환 블록에 역 변환을 적용할 수도 있다.

[0014] 입력 N-포인트 벡터가 $\mathbf{x} = [x_0, x_1, \dots, x_{N-1}]^T$ 로 표기되고 행렬을 승산함으로써 $\mathbf{y} = [y_0, y_1, \dots, y_{N-1}]^T$ 로서 표기된 또 다른 N-포인트 변환 계수 벡터로 변환된다고 가정하면, 다음의 변환 공식 중 하나에 따라 추가로 예시될 수 있는 프로세스는 다음을 포함하며, 식 중 k 는 0 부터 N-1 까지의 범위이다:

[0015] DCT 타입-I (DCT-1):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N-1}} \cos\left(\frac{\pi \cdot n \cdot k}{N-1}\right) \cdot w_0 \cdot w_1 \cdot x_n,$$

식중 $w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0 \text{ 또는 } n = N-1 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}, w_1 = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \text{ 또는 } k = N-1 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}$

[0016]

[0017] DCT 타입-II (DCT-2):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} \cos\left(\frac{\pi \cdot (n+0.5) \cdot k}{N}\right) \cdot w_0 \cdot x_n,$$

식중 $w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}$

[0018]

[0019] DCT 타입-III (DCT-3):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} \cos\left(\frac{\pi \cdot n \cdot (k+0.5)}{N}\right) \cdot w_0 \cdot x_n,$$

식중 $w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}$

[0020]

[0021] DCT 타입-IV (DCT-4):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} \cos\left(\frac{\pi \cdot (n+0.5) \cdot (k+0.5)}{N}\right) \cdot x_n,$$

[0022]

[0023] DCT 타입-V (DCT-5):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N-0.5}} \cos\left(\frac{\pi \cdot n \cdot k}{N-0.5}\right) \cdot w_0 \cdot w_1 \cdot x_n,$$

식중 $w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}, w_1 = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}$

[0024]

DCT 타입-VI (DCT-6):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N-0.5}} \cos\left(\frac{\pi \cdot (n+0.5) \cdot k}{N-0.5}\right) \cdot w_0 \cdot w_1 \cdot x_n ,$$

$$\text{식중 } w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = N - 1 \text{ 이면} \\ 1, & \text{그렇지 않으면} \end{cases}, w_1 = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \text{ 이면} \\ 1, & \text{그렇지 않으면} \end{cases}$$

DCT 타입-VII (DCT-7):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N-0.5}} \cos\left(\frac{\pi \cdot n \cdot (k+0.5)}{N-0.5}\right) \cdot w_0 \cdot w_1 \cdot x_n ,$$

$$\text{식중 } w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0 \text{ 이면} \\ 1, & 그렇지 않으면 \end{cases}, w_1 = \begin{cases} \frac{1}{\sqrt{2}}, & k = N - 1 \text{ 이면} \\ 1, & 그렇지 않으면 \end{cases}$$

DCT 탑입-VIII (DCT-8):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N+0.5}} \cos\left(\frac{\pi \cdot (n+0.5) \cdot (k+0.5)}{N+0.5}\right) \cdot x_n ,$$

DST 탑입-J (DST-1):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N+1}} \sin \left(\frac{\pi \cdot (n+1) \cdot (k+1)}{N+1} \right) \cdot x_n ,$$

DST 태일=II (DST=2):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} \sin\left(\frac{\pi \cdot (n+0.5) \cdot (k+1)}{N}\right) \cdot w_0 \cdot x_n,$$

$$\text{식중 } w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & k = N - 1 \text{ 일 때} \\ 1, & \text{그렇지 않으면} \end{cases}$$

DST_El91-III (DST-3):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} \sin \left(\frac{\pi \cdot (n+1) \cdot (k+0.5)}{N} \right) \cdot w_0 \cdot x_n ,$$

$$\text{식중 } w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = N - 1 \text{ 일면} \\ 1, & \text{그렇지 않으면} \end{cases}$$

DST E&O-IV (DST-4):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N}} \sin \left(\frac{\pi \cdot (n+0.5) \cdot (k+0.5)}{N} \right) \cdot x_n ,$$

DCT- π block (DCT- π)

$$\gamma_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{n+1}} \sin \left(\frac{\pi \cdot (n+1) \cdot (k+1)}{N} \right) \cdot x_n ,$$

$$\gamma_{\nu} \equiv \sum_{n=0}^{N-1} \sqrt{\frac{2}{\alpha}} \sin \left(\frac{\pi \cdot (n+0.5) \cdot (k+1)}{N} \right) \cdot \chi_n$$

[0043] DST 타입-VII (DST-7):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N+0.5}} \sin\left(\frac{\pi \cdot (n+1) \cdot (k+0.5)}{N+0.5}\right) \cdot x_n,$$

[0044] DST 타입-VIII (DST-8):

$$y_k = \sum_{n=0}^{N-1} \sqrt{\frac{2}{N-0.5}} \cos\left(\frac{\pi \cdot (n+0.5) \cdot (k+0.5)}{N-0.5}\right) \cdot w_0 \cdot w_1 \cdot x_n,$$

$$\text{식중 } w_0 = \begin{cases} \frac{1}{\sqrt{2}}, & n = N - 1 \text{ 이면} \\ 1, & \text{그렇지 않으면} \end{cases}, \quad w_1 = \begin{cases} \frac{1}{\sqrt{2}}, & k = N - 1 \text{ 이면} \\ 1, & \text{그렇지 않으면} \end{cases}$$

[0045] [0046] 변환 타입은 변환 기반 함수의 수학적 공식에 의해 특정된다. 예를 들어, 4-포인트 DST-VII 와 8-포인트 DST-VIII 는 N 의 값에 관계없이, 동일한 변환 타입을 갖는다.

[0047] 일반성을 잃지 않으면서, 위의 변환 타입들 모두는 하기 일반화된 공식을 사용하여 표현될 수 있다:

$$y_m = \sum_{n=0}^{N-1} T_{m,n} \cdot x_n,$$

[0048] [0049] 식 중 T 는 하나의 소정 변환의 정의, 예를 들어 DCT 타입-I ~ DCT 타입-VIII, 또는 DST 타입-I ~ DST 타입-VIII 로 특정되며, T 의 행 벡터들, 예를 들어 $[T_{i,0}, T_{i,1}, T_{i,2}, \dots, T_{i,N-1}]$ 은 i 번째 변환 기반 벡터들이다. N-포인트 입력 벡터 상에 적용된 변환은 N-포인트 변환이라 한다.

[0050] 또한 1-D 입력 데이터 x 상에 적용되는 위의 변환 공식들은, 하기와 같이 행렬 곱셈 형식으로 표현될 수 있다.

$$y = \mathbf{T} \cdot \mathbf{x}$$

[0051] [0052] 식 중 T 는 변환 행렬을 표시하고, x 는 입력 데이터 벡터를 표시하며, y 는 출력 변환 계수 벡터를 표시한다.

[0053] [0054] 위에 도입된 바와 같은 변환들은 1-D 입력 데이터 상에 적용될 수도 있으며, 변환들은 2-D 입력 데이터 소스들에 대해 또한 확장될 수 있다. X 가 입력 MxN 데이터 어레이라고 한다. 2-D 입력 데이터 상에 변환을 적용하는 통상의 방법들은 분리가능한 및 분리불가능한 2-D 변환들을 포함한다.

[0055] [0056] 분리가능한 2-D 변환은 X 의 수평 및 수직 벡터들에 대한 1-D 변환들을 순차적으로 적용하며, 이는 다음과 같이 공식화된다:

$$Y = C \cdot X \cdot R^T$$

[0057] 식 중 C 및 R 은 주어진 MxM 및 NxN 변환 행렬들을 각각 표기한다. 공식으로부터, C 는 X 의 열 벡터들에 대해 1-D 변환들을 적용하는 한편, R 은 X 의 행 벡터들에 대해 1-D 변환들을 적용함을 알 수 있다. 본 개시의 이후 부분에서는, 단순함을 위해, C 및 R 이 좌측 (수직) 및 우측 (수평) 변환들로서 표기될 수 있고 이들 양자 모두는 변환 쌍을 형성하는 것으로 고려될 수 있다. C 가 R 과 같고 직교 행렬일 때의 경우들이 있다. 이러한 경우, 분리가능한 2-D 변환은 단 하나의 변환 행렬에 의해 결정된다.

[0058] [0059] 분리불가능한 2D 변환은 먼저, 예로서 다음의 수학적 매핑을 행함으로써, X 의 모든 엘리먼트들을 단일 벡터, 즉 X' 로 재편성하였다:

$$X'_{(i \cdot N + j)} = X_{i,j}$$

[0060] 그 후 1-D 변환 T' 은 하기와 같이 X' 에 대해 적용된다:

$$Y = T' \cdot X'$$

[0061] [0062] 식 중 T' 는 $(M \times N) \times (M \times N)$ 변환 행렬이다.

[0063] 비디오 코딩에서, 분리가능한 2-D 변환들이 일반적으로 적용되는데, 이는 분리가능한 2-D 변환들이 1-D 변환에

비해 훨씬 더 적은 연산 (덧셈 및 곱셈) 카운트를 필요로 하기 때문이다.

[0064] H.264/AVC 와 같은 종래 비디오 코덱에서, 4 포인트 및 8 포인트 이산 코사인 변환 (DCT) 타입-II 의 정수 근사화는 항상 인트라 및 인터-예측 잔차 양자 모두에 대해 적용된다. 잔차 샘플들의 다양한 통계를 더 잘 수용하기 위해, DCT 타입-II 이외의 더 유연한 타입의 변환들이 차세대 비디오 코덱에서 활용된다. 예를 들어, HEVC 에서, 4-포인트 타입-VII 이산 사인 변환 (DST) 의 정수 근사화는, DST 타입-VII 가 인트라-예측 방향들을 따라 생성된 잔차 벡터들에 대한 DCT 타입-II 보다 더 효율적인 것으로 (J. Han, A. Saxena 및 K. Rose, "Towards jointly optimal spatial prediction and adaptive transform in video/image coding," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2010, pp. 726-729 에서) 이론적으로 입증되고 실험적으로 검증된 인트라-예측 잔차에 활용된다. 예를 들어, DST 타입-VII 는 수평 인트라-예측 방향에 의해 생성된 행 잔차 벡터들에 대해 DCT 타입-II 보다 더 효율적이다. HEVC 에서, 4-포인트 DST 타입-VII 의 정수 근사화는 4x4 루마 인트라-예측 잔차 블록들에 대해서만 적용된다. HEVC 에서 사용된 4 포인트 DST-VII 는 하기에 나타나 있다,

4x4 DST-VII:

{29, 55, 74, 84}

{74, 74, 0,-74}

{84,-29,-74, 55}

{55,-84, 74,-29}

[0065]

[0066] HEVC 에서, 4x4 루마 인트라-예측 잔차 블록들이 아닌 잔차 블록들에 대해, 하기에 나타낸 바와 같이, 4 포인트, 8 포인트, 16 포인트 및 32 포인트 DCT 타입 II 의 정수 근사화가 또한 적용된다:

[0067]

4 포인트 DCT-II :

{64, 64, 64, 64}

{83, 36,-36,-83}

{64,-64,-64, 64}

{36,-83, 83,-36}

[0068]

8 포인트 DCT-II :

{64, 64, 64, 64, 64, 64, 64, 64}

{89, 75, 50, 18,-18,-50,-75,-89}

{83, 36,-36,-83,-83,-36, 36, 83}

{75,-18,-89,-50, 50, 89, 18,-75}

{64,-64,-64, 64, 64,-64,-64, 64}

{50,-89, 18, 75,-75,-18, 89,-50}

{36,-83, 83,-36,-36, 83,-83, 36}

[0070]

{18,-50, 75,-89, 89,-75, 50,-18}

[0071]

16-포인트 DCT-II :

{70,-43,-87, 9, 90, 25,-80,-57, 57, 80,-25,-90, -9, 87, 43,-70}
{64,-64,-64, 64, 64,-64,-64, 64, 64,-64,-64, 64, 64,-64,-64, 64}
{57,-80,-25, 90, -9,-87, 43, 70,-70,-43, 87, 9,-90, 25, 80,-57}
{50,-89, 18, 75,-75,-18, 89,-50,-50, 89,-18,-75, 75, 18,-89, 50}
{43,-90, 57, 25,-87, 70, 9,-80, 80, -9,-70, 87,-25,-57, 90,-43}
{36,-83, 83,-36,-36, 83,-83, 36, 36,-83, 83,-36,-36, 83,-83, 36}
{25,-70, 90,-80, 43, 9,-57, 87,-87, 57, -9,-43, 80,-90, 70,-25}
{18,-50, 75,-89, 89,-75, 50,-18,-18, 50,-75, 89,-89, 75,-50, 18}
{9, -25, 43,-57, 70,-80, 87,-90, 90,-87, 80,-70, 57,-43, 25, -9}

$$\{9, -25, 43, -57, 70, -80, 87, -90, 90, -87, 80, -70, 57, -43, 25, -9\}$$

32-포인트 DCT-II :

```

{87,57,9,-43,-80,-90,-70,-25,25,70,90,80,43,-9,-57,-87,-87,-57,-9,43,80,90,70,25,-25,-
 70,-90,-80,-43,9,57,87}

{85,46,-13,-67,-90,-73,-22,38,82,88,54,-4,-61,-90,-78,-31,31,78,90,61,4,-54,-88,-82,-
 38,22,73,90,67,13,-46,-85}

```

{83,36,-36,-83,-83,-36,36,83,83,36,-36,-83,-83,-36,36,83,83,36,-36,-83,-83,-
 36,36,83,83,36,-36,-83,-83,-36,36,83}
 {82,22,-54,-90,-61,13,78,85,31,-46,-90,-67,4,73,88,38,-38,-88,-73,-4,67,90,46,-31,-85,-
 78,-13,61,90,54,-22,-82}
 {80,9,-70,-87,-25,57,90,43,-43,-90,-57,25,87,70,-9,-80,-80,-9,70,87,25,-57,-90,-
 43,43,90,57,-25,-87,-70,9,80}
 {78,-4,-82,-73,13,85,67,-22,-88,-61,31,90,54,-38,-90,-46,46,90,38,-54,-90,-
 31,61,88,22,-67,-85,-13,73,82,4,-78}
 {75,-18,-89,-50,50,89,18,-75,-75,18,89,50,-50,-89,-18,75,75,-18,-89,-50,50,89,18,-75,-
 75,18,89,50,-50,-89,-18,75}
 {73,-31,-90,-22,78,67,-38,-90,-13,82,61,-46,-88,-4,85,54,-54,-85,4,88,46,-61,-
 82,13,90,38,-67,-78,22,90,31,-73}
 {70,-43,-87,9,90,25,-80,-57,57,80,-25,-90,-9,87,43,-70,-70,43,87,-9,-90,-25,80,57,-57,-
 80,25,90,9,-87,-43,70}
 {67,-54,-78,38,85,-22,-90,4,90,13,-88,-31,82,46,-73,-61,61,73,-46,-82,31,88,-13,-90,-
 4,90,22,-85,-38,78,54,-67}
 {64,-64,-64,64,64,-64,-64,64,64,-64,-64,64,64,-64,-64,64,64,-64,-64,64,64,-64,-
 64,64,64,-64,-64,64,64,-64,-64,64}
 {61,-73,-46,82,31,-88,-13,90,-4,-90,22,85,-38,-78,54,67,-67,-54,78,38,-85,-22,90,4,-
 90,13,88,-31,-82,46,73,-61}
 {57,-80,-25,90,-9,-87,43,70,-70,-43,87,9,-90,25,80,-57,-57,80,25,-90,9,87,-43,-
 70,70,43,-87,-9,90,-25,-80,57}
 {54,-85,-4,88,-46,-61,82,13,-90,38,67,-78,-22,90,-31,-73,73,31,-90,22,78,-67,-38,90,-
 13,-82,61,46,-88,4,85,-54}
 {50,-89,18,75,-75,-18,89,-50,-50,89,-18,-75,75,18,-89,50,50,-89,18,75,-75,-18,89,-50,-
 50,89,-18,-75,75,18,-89,50}
 {46,-90,38,54,-90,31,61,-88,22,67,-85,13,73,-82,4,78,-78,-4,82,-73,-13,85,-67,-22,88,-
 61,-31,90,-54,-38,90,-46}
 {43,-90,57,25,-87,70,9,-80,80,-9,-70,87,-25,-57,90,-43,-43,90,-57,-25,87,-70,-9,80,-
 80,9,70,-87,25,57,-90,43}

{38,-88,73,-4,-67,90,-46,-31,85,-78,13,61,-90,54,22,-82,82,-22,-54,90,-61,-13,78,-
85,31,46,-90,67,4,-73,88,-38}
 {36,-83,83,-36,-36,83,-83,36,36,-83,83,-36,-36,83,-83,36,36,-83,83,-36,-36,83,-
83,36,36,-83,83,-36,-36,83,-83,36}
 {31,-78,90,-61,4,54,-88,82,-38,-22,73,-90,67,-13,-46,85,-85,46,13,-67,90,-73,22,38,-
82,88,-54,-4,61,-90,78,-31}
 {25,-70,90,-80,43,9,-57,87,-87,57,-9,-43,80,-90,70,-25,-25,70,-90,80,-43,-9,57,-87,87,-
57,9,43,-80,90,-70,25}
 {22,-61,85,-90,73,-38,-4,46,-78,90,-82,54,-13,-31,67,-88,88,-67,31,13,-54,82,-90,78,-
46,4,38,-73,90,-85,61,-22}
 {18,-50,75,-89,89,-75,50,-18,-18,50,-75,89,-89,75,-50,18,18,-50,75,-89,89,-75,50,-18,-
18,50,-75,89,-89,75,-50,18}
 {13,-38,61,-78,88,-90,85,-73,54,-31,4,22,-46,67,-82,90,-90,82,-67,46,-22,-4,31,-54,73,-
85,90,-88,78,-61,38,-13}
 {9,-25,43,-57,70,-80,87,-90,90,-87,80,-70,57,-43,25,-9,-9,25,-43,57,-70,80,-87,90,-
90,87,-80,70,-57,43,-25,9}
 {4,-13,22,-31,38,-46,54,-61,67,-73,78,-82,85,-88,90,-90,90,-90,88,-85,82,-78,73,-
67,61,-54,46,-38,31,-22,13,-4}

[0079]

[0080] 도 1 은 본 개시의 기법들을 수행할 수도 있는 비디오 인코딩 및 디코딩 시스템 (100) 의 예를 도시하는 블록 다이어그램이다. 본 개시의 기법들은 일반적으로, 비디오 데이터를 코딩 (인코딩 및/또는 디코딩) 하는 것에 관련된다. 일반적으로, 비디오 데이터는 비디오를 프로세싱하기 위한 임의의 데이터를 포함한다. 따라서, 비디오 데이터는 원시, 코딩되지 않은 비디오, 인코딩된 비디오, 디코딩된 (예를 들어, 복원된) 비디오, 및 비디오 메타데이터, 예컨대 시그널링 데이터를 포함할 수도 있다.

[0081]

도 1 에 나타낸 바와 같이, 시스템 (100) 은 이 예에서 목적지 디바이스 (14) 에 의해 디코딩 및 디스플레이될 인코딩된 비디오 데이터를 제공하는 소스 디바이스 (102) 를 포함한다. 특히, 소스 디바이스 (102) 는 컴퓨터 판독 가능 매체 (110) 를 통해 목적지 디바이스 (116) 에 비디오 데이터를 제공한다. 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 테스크탑 컴퓨터들, 노트북 (즉, 랙톱) 컴퓨터들, 태블릿 컴퓨터들, 셋톱 박스들, 전화기 핸드셋, 예컨대 스마트폰들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함한, 광범위한 디바이스들 중 임의의 것을 포함할 수도 있다. 일부 경우들에서, 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 무선 통신을 위해 장비될 수도 있고, 따라서 무선 통신 디바이스들로서 지칭될 수도 있다.

[0082]

도 1 의 예에서, 소스 디바이스 (102) 는 비디오 소스 (104), 메모리 (106), 비디오 인코더 (200) 및 출력 인터페이스 (108) 를 포함한다. 목적지 디바이스 (116) 는 입력 인터페이스 (122), 비디오 디코더 (300), 메모리 (120), 및 디스플레이 디바이스 (118) 를 포함한다. 본 개시에 따라, 소스 디바이스 (102) 의 비디오 인코더 (200) 및 목적지 디바이스 (116) 의 비디오 디코더 (300) 는 MTS 데이터를 코딩하기 위한 기법들을 적용하도록 구성될 수도 있다. 따라서, 소스 디바이스 (102) 는 비디오 인코딩 디바이스의 예를 나타내는 한편, 목적지 디바이스 (116) 는 비디오 디코딩 디바이스의 예를 나타낸다. 다른 예들에서, 소스 디바이스 및 목적지 디바이스는 다른 컴포넌트들 또는 배열들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (102) 는 외부 카메라와 같은 외부 비디오 소스로부터 비디오 데이터를 수신할 수도 있다. 마찬가지로, 목적지 디바이스 (116) 는 통합된 디스플레이 디바이스를 포함하기 보다는, 외부 디스플레이 디바이스와 인터페이스할 수도 있다.

[0083]

도 1 에 나타낸 시스템 (100) 은 단지 하나의 예일 뿐이다. 일반적으로, 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스는 MTS 데이터를 코딩하기 위한 기법들을 수행할 수도 있다. 소스 디바이스 (102) 및 목적지 디바이스 (116) 는 소스 디바이스 (102) 가 목적지 디바이스 (116) 로의 송신을 위한 코딩된 비디오 데이터를 생성하는 이러한 코딩 디바이스들의 예들일 뿐이다. 본 개시는 데이터의 코딩 (인코딩 및/또는 디코

딩) 을 수행하는 디바이스로서 "코딩" 디바이스를 언급한다. 따라서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 코딩 디바이스, 특히 각각 비디오 인코더 및 비디오 디코더의 예들을 나타낸다. 일부 예들에 있어서, 디바이스들 (102, 116) 은 디바이스들 (102, 116) 의 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 이로써, 시스템 (100) 은 예를 들어, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 텔레포니를 위해, 비디오 디바이스들 (102, 116) 사이의 일방향 또는 이방향 비디오 송신을 지원할 수도 있다.

[0084] 일반적으로, 비디오 소스 (104) 는 비디오 데이터 (즉, 원시, 코딩되지 않은 비디오 데이터) 의 소스를 나타내며 핵심에 대한 데이터를 인코딩하는 비디오 인코더 (200) 로 비디오 데이터의 순차적인 일련의 핵심들 (또한 "프레임들"으로서 지칭됨) 을 제공한다. 소스 디바이스 (102) 의 비디오 소스 (104) 는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 원시 비디오를 포함하는 비디오 아카이브, 및/또는 비디오 컨텐츠 제작자로부터 비디오를 수신하기 위한 비디오 퍼드 인터페이스를 포함할 수도 있다. 추가적인 대안으로서, 비디오 소스 (104) 는 컴퓨터 그래픽 기반 데이터를 소스 비디오로서, 또는 라이브 비디오, 아카이브된 비디오, 및 컴퓨터 생성된 비디오의 조합으로서 생성할 수도 있다. 각각의 경우에 있어서, 비디오 인코더 (200) 는 캡처되거나 사전-캡처되거나 또는 컴퓨터 생성된 비디오 데이터를 인코딩한다. 비디오 인코더 (200) 는 수신된 순서 (때때로 "디스플레이 순서"로 지칭됨) 로부터 핵심들을 코딩을 위한 코딩 순서로 재배열할 수도 있다. 비디오 인코더 (200) 는 인코딩된 비디오 데이터를 포함하는 비트스트림을 생성할 수도 있다. 소스 디바이스 (102) 는 그 후 예를 들어 목적지 디바이스 (116) 의 입력 인터페이스 (122) 에 의한 수신 및/또는 취출을 위해 컴퓨터 판독 가능 매체 (110) 상으로 출력 인터페이스 (108) 를 통해 인코딩된 비디오 데이터를 출력할 수도 있다.

[0085] 소스 디바이스 (102) 의 메모리 (106) 및 목적지 디바이스 (116) 의 메모리 (120) 는 범용 메모리들을 나타낸다. 일부 예에서, 메모리들 (106, 120) 은 원시 비디오 데이터, 예를 들어 비디오 소스 (104) 로부터의 원시 비디오, 및 비디오 디코더 (300) 로부터의 원시, 디코딩된 비디오 데이터를 저장할 수도 있다. 부가적으로 또는 대안으로, 메모리들 (106, 120) 은 예를 들어, 비디오 인코더 (200) 및 비디오 디코더 (300) 에 의해 각각 실행가능한 소프트웨어 명령을 저장할 수도 있다. 이 예에서는 비디오 인코더 (200) 및 비디오 디코더 (300) 와 별도로 나타나 있지만, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 또한 기능적으로 유사하거나 동등한 목적을 위한 내부 메모리들을 포함할 수 있음을 이해해야 한다. 또한, 메모리들 (106, 120) 은 예를 들어, 비디오 인코더 (200) 로부터 출력되고 비디오 디코더 (300) 에 입력된 인코딩된 비디오 데이터를 저장할 수도 있다. 일부 예들에서, 메모리들 (106, 120) 의 부분들은 예를 들어, 원시, 디코딩 및/또는 인코딩된 비디오 데이터를 저장하기 위해 하나 이상의 비디오 버퍼로서 할당될 수도 있다.

[0086] 컴퓨터 판독 가능 매체 (110) 는 인코딩된 비디오 데이터를 소스 디바이스 (102) 로부터 목적지 디바이스 (116) 로 전송할 수 있는 임의의 타입의 매체 또는 디바이스를 나타낼 수도 있다. 일부 예에서, 컴퓨터 판독 가능 매체 (110) 는 소스 디바이스 (102) 가 실시간으로, 예를 들어 무선 주파수 네트워크 또는 컴퓨터 기반 네트워크를 통해 직접 목적지 디바이스 (116) 로 인코딩된 비디오 데이터를 송신하는 것을 가능하게 하기 위한 통신 매체를 나타낸다. 출력 인터페이스 (108) 는 인코딩된 비디오 데이터를 포함하는 송신 신호를 변조할 수도 있고, 입력 인터페이스 (122) 는 무선 통신 프로토콜과 같은 통신 표준에 따라 수신된 송신 신호를 변조할 수도 있다. 통신 매체는 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 패킷 기반 네트워크, 예컨대 로컬 영역 네트워크, 광역 네트워크, 또는 글로벌 네트워크, 예컨대 인터넷의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (102) 로부터 목적지 디바이스 (116) 로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0087] 일부 예들에서, 소스 디바이스 (102) 는 출력 인터페이스 (108) 로부터 저장 디바이스 (112) 로 인코딩된 데이터를 출력할 수도 있다. 유사하게, 목적지 디바이스 (116) 는 입력 인터페이스 (122) 를 통해 저장 디바이스 (112) 로부터의 인코딩된 데이터에 액세스할 수도 있다. 저장 디바이스 (116) 는 하드 드라이브, 블루레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 휘발성 또는 비휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체와 같은 다양한 분산된 또는 로컬 액세스된 데이터 저장 매체 중 임의의 것을 포함할 수도 있다.

[0088] 일부 예들에서, 소스 디바이스 (102) 는 소스 디바이스 (102) 에 의해 생성된 인코딩된 비디오를 저장할 수도 있는 파일 서버 (114) 또는 다른 중간 저장 디바이스에 인코딩된 비디오 데이터를 출력할 수도 있다. 목적지 디바이스 (116) 는 스트리밍 또는 다운로드를 통해 파일 서버 (114) 로부터 저장된 비디오 데이터에 액세스

할 수도 있다. 파일 서버 (114)는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (116)로 송신할 수 있는 임의의 타입의 서버 디바이스일 수도 있다. 파일 서버 (114)는 (예를 들어, 웹 사이트를 위한) 웹 서버, 파일 전송 프로토콜 (FTP) 서버, 컨텐츠 전달 네트워크 디바이스, 또는 NAS (network attached storage) 디바이스를 나타낼 수도 있다. 목적지 디바이스 (116)는 인터넷 접속을 포함하는, 임의의 표준 데이터 접속을 통해 파일 서버 (114)로부터 인코딩된 비디오 데이터에 액세스할 수도 있다. 이것은 파일 서버 (114) 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양자의 조합을 포함할 수도 있다. 파일 서버 (114) 및 입력 인터페이스 (122)는 스트리밍 송신 프로토콜, 다운로드 송신 프로토콜 또는 이들의 조합에 따라 동작하도록 구성될 수도 있다.

[0089] 출력 인터페이스 (108) 및 입력 인터페이스 (122)는 무선 송신기/수신기, 모뎀, 유선 네트워킹 컴포넌트 (예를 들어, 이더넷 카드), 다양한 IEEE 802.11 표준 중 임의의 것에 따라 동작하는 무선 통신 컴포넌트, 또는 다른 물리적 컴포넌트를 나타낼 수 있다. 출력 인터페이스 (108) 및 입력 인터페이스 (122)가 무선 컴포넌트를 포함하는 예들에 있어서, 출력 인터페이스 (108) 및 입력 인터페이스 (122)는 4G, 4G-LTE (Long-Term Evolution), LTE 어드밴스드, 5G 등과 같은 셀룰러 통신 표준에 따라, 인코딩된 비디오 데이터와 같은 데이터를 전송하도록 구성될 수도 있다. 출력 인터페이스 (22)가 무선 송신기를 포함하는 일부 예들에 있어서, 출력 인터페이스 (108) 및 입력 인터페이스 (122)는 IEEE 802.11 사양, IEEE 802.15 사양 (예를 들어, ZigBeeTM), BluetoothTM 표준 등과 같은 다른 무선 표준들에 따라, 인코딩된 비디오 데이터와 같은 데이터를 전송하도록 구성될 수도 있다. 일부 예들에서, 소스 디바이스 (102) 및/또는 목적지 디바이스 (116)는 개개의 시스템-온-칩 (SoC) 디바이스들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (102)는 비디오 인코더 (200) 및/또는 출력 인터페이스 (108)에 기인하는 기능성을 수행하기 위한 SoC 디바이스를 포함할 수도 있고, 목적지 디바이스 (116)는 비디오 디코더 (300) 및/또는 입력 인터페이스 (122)에 기인하는 기능성을 수행하기 위한 SoC 디바이스를 포함할 수도 있다.

[0090] 본 개시의 기법들은 오버-디-에어 (over-the-air) 텔레비전 브로드캐스트, 케이블 텔레비전 송신, 위성 텔레비전 송신, 인터넷 스트리밍 비디오 송신, 예컨대 DASH (dynamic adaptive streaming over HTTP), 데이터 저장 매체 상으로 인코딩되는 디지털 비디오, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션과 같은 다양한 멀티미디어 애플리케이션들 중 임의의 것을 지원하는 비디오 코딩에 적용될 수도 있다.

[0091] 목적지 디바이스 (116)의 입력 인터페이스 (122)는 컴퓨터 판독가능 매체 (110) (예를 들어, 저장 디바이스 (112), 파일 서버 (114) 등)로부터 인코딩된 비디오 비트스트림을 수신한다. 컴퓨터 판독가능 매체 (110)의 인코딩된 비디오 비트스트림은 비디오 블록들 또는 다른 코딩된 유닛들 (예를 들어, 슬라이스들, 픽처들, 픽처들의 그룹들, 시퀀스들 등)의 프로세싱 및/또는 특징들을 기술하는 값들을 갖는 신팩스 엘리먼트들과 같은, 비디오 디코더 (300)에 의해 또한 사용되는, 비디오 인코더 (200)에 의해 정의된 시그널링 정보를 포함할 수도 있다. 디스플레이 디바이스 (118)는 디코딩된 비디오 데이터의 디코딩된 픽처들을 사용자에게 디스플레이 한다. 디스플레이 디바이스 (118)는 음극선관 (CRT), 액정 디스플레이 (LCD), 평면 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 것을 나타낼 수도 있다.

[0092] 도 1에 나타내지는 않았지만, 일부 예들에서, 비디오 인코더 (200) 및 비디오 디코더 (300)는 각각 오디오 인코더 및/또는 오디오 디코더와 통합될 수도 있고, 공통 데이터 스트림에서 오디오 및 비디오 양자 모두를 포함하는 멀티플렉싱된 스트림을 핸들링하기 위해, 적절한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및/또는 소프트웨어를 포함할 수도 있다. 적용가능하다면, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜 또는 다른 프로토콜들, 예컨대 사용자 데이터그램 프로토콜 (UDP)을 따를 수도 있다.

[0093] 비디오 인코더 (200) 및 비디오 디코더 (300)는 각각 다양한 적합한 인코더 및/또는 디코더 회로부, 예컨대 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서 (DSP)들, 주문형 집적회로 (ASIC)들, 필드 프로그램가능 게이트 어레이 (FPGA)들, 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들으로서 구현될 수도 있다. 기법들이 부분적으로 소프트웨어로 구현될 때, 디바이스는 적합한 비일시적 컴퓨터 판독가능 매체에 소프트웨어를 위한 명령들을 저장하고 본 개시의 기법들을 수행하기 위해 하나 이상의 프로세서들을 사용하는 하드웨어에서 명령들을 실행할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300)의 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있는데, 이를 중 어느 하나는 개개의 디바이스에 있어서 조합된 인코더/디코더 (CODEC)의 부분으로서 통합될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300)

를 포함하는 디바이스는 집적 회로, 마이크로프로세서, 및/또는 무선 통신 디바이스, 예컨대 셀룰러 전화기를 포함할 수도 있다.

[0094] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 고효율 비디오 코딩 (HEVC) 으로서 또한 지정되는 ITU-T H.265 와 같은 비디오 코딩 표준 또는 그에 대한 확장들, 예컨대 멀티-뷰 및/또는 스케일러블 비디오 코딩 확장들에 따라 동작할 수도 있다. 대안으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 ITU-T H.266 이 되도록 계획되는, 다가오는 다기능 비디오 코딩 (VVC) 와 같은, 다른 독점 또는 산업 표준들에 따라 동작할 수도 있다.

VVC 의 작업 초안은 Bross 등이, "Versatile Video Coding (Draft 5)" ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 JVET (Joint Video Experts Team), 14th Meeting, Geneva, CH, 19-27 Mar. 2019, document JVET-N1001-v5 이다. 하지만, 본 개시의 기법들은 임의의 특정 코딩 표준에 제한되지 않는다.

[0095] 일반적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 빡처들의 블록 기반 코딩을 수행할 수도 있다. 용어 "블록" 은 일반적으로 프로세싱될 (예를 들어, 인코딩될, 디코딩될, 또는 인코딩 및 / 또는 디코딩 프로세스에서 사용될) 데이터를 포함하는 구조를 지칭한다. 예를 들어, 블록은 루미넌스 및/또는 크로미넌스 데이터의 샘플들의 2차원 행렬을 포함할 수도 있다. 일반적으로, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 YUV (예를 들어, Y, Cb, Cr) 포맷으로 표현된 비디오 데이터를 코딩할 수도 있다. 즉, 빡처의 샘플들에 대한 적색, 녹색, 및 청색 (RGB) 을 코딩하기 보다, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 (루마) 및 크로미넌스 성분들을 코딩할 수도 있으며, 여기서 크로미넌스 성분들은 적색 색조 및 청색 색조 크로미넌스 성분들 양자 모두를 포함할 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 인코딩 이전에 수신된 RGB 포맷된 데이터를 YUV 표현으로 변환하고, 비디오 디코더 (300) 는 YUV 표현을 RGB 포맷으로 변환한다. 대안으로, 프리-프로세싱 및 포스트-프로세싱 유닛들 (도시되지 않음) 이 이들 변환들을 수행할 수도 있다.

[0096] 본 개시는 일반적으로 빡처의 데이터를 인코딩 또는 디코딩하는 프로세스를 포함하기 위해 빡처들의 코딩 (예를 들어, 인코딩 및 디코딩) 을 언급한다. 유사하게, 본 개시는 블록들에 대한 데이터를 인코딩 또는 디코딩하는 프로세스, 예를 들어 예측 및/또는 잔차 코딩을 포함하기 위해 빡처의 블록들의 코딩을 언급할 수도 있다. 인코딩된 비디오 비트스트림은 일반적으로 빡처들의 블록들로의 파티셔닝 및 코딩 결정들 (예를 들어, 코딩 모드들) 을 나타내는 선택스 엘리먼트들에 대한 일련의 값들을 포함한다. 따라서, 빡처 또는 블록을 코딩하는 것에 대한 참조들은 일반적으로 빡처 또는 블록을 형성하는 선택스 엘리먼트에 대한 코딩 값들로서 이해되어야 한다.

[0097] HEVC 는 코딩 유닛 (CU), 예측 유닛 (PU) 및 변환 유닛 (TU) 을 포함하는 다양한 블록들을 정의한다. HEVC 에 따라, (비디오 인코더 (200) 와 같은) 비디오 코더는 퀘드트리 구조에 따라 코딩 트리 유닛 (CTU) 을 CU들로 파티셔닝한다. 즉, 비디오 코더는 CTU들 및 CU들을 4 개의 동등한, 오버랩하지 않는 정사각형들로 파티셔닝하고, 퀘드트리의 각 노드는 0 또는 4 개의 자식 노드들을 갖는다. 자식 노드가 없는 노드들은 "리프 노드들" 로서 지정될 수도 있으며, 이러한 리프 노드들의 CU들은 하나 이상의 PU들 및/또는 하나 이상의 TU들을 포함할 수도 있다. 비디오 코더는 PU들 및 TU들을 추가로 파티셔닝할 수도 있다. 예를 들어, HEVC 에서, 잔차 퀘드트리 (RQT) 는 TU들의 파티셔닝을 나타낸다. HEVC에서, PU들은 인터-예측 데이터를 나타내는 한편, TU들은 잔차 데이터를 나타낸다. 인트라-예측되는 CU들은 인트라-모드 표시와 같은 인트라-예측 정보를 포함한다.

[0098] 다른 예로서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 VVC 에 따라 동작하도록 구성될 수도 있다. VVC 에 따라, 비디오 코더 (예컨대 비디오 인코더 (200)) 는 빡처를 복수의 코딩 트리 유닛들 (CTU들) 로 파티셔닝한다. 비디오 인코더 (200) 는 퀘드트리 이진 트리 (QTBT) 구조와 같은 트리 구조에 따라 CTU 를 파티셔닝할 수도 있다. VVC 의 QTBT 구조는 HEVC 의 CU들, PU들, 및 TU들 사이의 분리와 같은, 다중 파티션 태입들의 개념들을 제거한다. VVC 의 QTBT 구조는 2 개의 레벨들: 퀘드트리 파티셔닝에 따라 파티셔닝된 제 1 레벨, 및 이진 트리 파티셔닝에 따라 파티셔닝된 제 2 레벨을 포함한다. QTBT 구조의 루트 노드는 CTU 에 대응한다. 이진 트리들의 리프 노드들은 코딩 유닛들 (CU들) 에 대응한다.

[0099] 일부 예들에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 및 크로미넌스 성분들의 각각을 나타내기 위해 단일 QTBT 구조를 사용할 수도 있는 한편, 다른 예들에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 루미넌스 성분에 대한 하나의 QTBT 구조 및 양자의 크로미넌스 성분들에 대한 다른 QTBT 구조 (또는 개개의 크로미넌스 성분들에 대한 2 개의 QTBT 구조들) 와 같은 2 이상의 QTBT 구조들을 사용할 수도 있다.

[0100] 비디오 인코더 (200) 및 비디오 디코더 (300) 는 HEVC 마다의 퀘드트리 파티셔닝, VVC 에 따른 QTBT 파티셔닝,

또는 다른 파티셔닝 구조들을 사용하도록 구성될 수도 있다. 설명의 목적을 위해, 본 개시의 기법들의 기재는 QTBT 파티셔닝에 대하여 제시된다. 그러나, 본 개시의 기법들은 또한, 퀘드트리 파티셔닝, 또는 다른 타입들의 파티셔닝에도 사용하도록 구성된 비디오 코더들에 적용될 수도 있음이 이해되어야 한다.

[0101] 본 개시는 수직 및 수평 치수들에 관하여 (CU 또는 다른 비디오 블록과 같은) 블록의 샘플 치수들을 지칭하기 위해 "NxN" 및 "N 바이 N", 예를 들어 16x16 샘플들 또는 16 바이 16 샘플들을 상호교환가능하게 사용할 수도 있다. 일반적으로, 16x16 CU 는 수직 방향에서 16 샘플들 ($y = 16$) 그리고 수평 방향에서 16 샘플들 ($x = 16$) 을 가질 것이다. 마찬가지로, NxN CU 는 일반적으로 수직 방향에서 N 샘플들 및 수평 방향에서 N 샘플들을 갖고, 여기서 N 은 음이 아닌 정수 값을 나타낸다. CU 에서의 샘플들은 행들 및 열들로 배열될 수도 있다. 더욱이, CU들은 수직 방향에서의 동일한 수의 샘플들을 수평 방향에서 반드시 가질 필요는 없다. 예를 들면, CU들은 NxM 샘플들을 포함할 수도 있고, 여기서 M 은 N 과 반드시 동일한 것은 아니다.

[0102] 비디오 인코더 (200) 는 예측 및/또는 잔차 정보를 나타내는 CU들에 대한 비디오 데이터, 및 다른 정보를 인코딩한다. 예측 정보는 CU 에 대한 예측 블록을 형성하기 위해 CU 가 어떻게 예측될지를 표시한다. 잔차 정보는 일반적으로 인코딩 전의 CU 의 샘플들과 예측 블록 사이의 샘플 별 차이들을 나타낸다.

[0103] CU 를 예측하기 위해, 비디오 인코더 (200) 는 일반적으로 인터-예측 또는 인트라-예측을 통해 CU 에 대한 예측 블록을 형성할 수도 있다. 인터-예측은 일반적으로 이전에 코딩된 픽처의 데이터로부터 CU 를 예측하는 것을 지칭하는 반면, 인트라-예측은 일반적으로 동일한 픽처의 이전에 코딩된 데이터로부터 CU 를 예측하는 것을 지칭한다. 인터-예측을 수행하기 위해, 비디오 인코더 (200) 는 하나 이상의 모션 벡터들 사용하여 예측 블록을 생성할 수도 있다. 비디오 인코더 (200) 는 일반적으로, 예를 들어 CU 와 참조 블록 사이의 차이에 관하여, CU 와 밀접하게 매칭하는 참조 블록을 식별하기 위해 모션 탐색을 수행할 수도 있다. 비디오 인코더 (200) 는 참조 블록이 현재 CU 와 밀접하게 매칭하는지 여부를 결정하기 위해 절대차의 합 (sum of absolute difference; SAD), 제곱차의 합 (sum of squared differences; SSD), 평균 절대차 (mean absolute difference; MAD), 평균 제곱차 (mean squared differences; MSD) 또는 다른 그러한 차이 계산들을 사용하여 차이 메트릭을 계산할 수도 있다. 일부 예들에서, 비디오 인코더 (200) 는 단방향 예측 또는 양방향 예측을 사용하여 현재 CU 를 예측할 수도 있다.

[0104] VVC 는 또한, 인터-예측 모드로 고려될 수도 있는 아핀 모션 보상 모드를 제공한다. 아핀 모션 보상 모드에서, 비디오 인코더 (200) 는 줌인 또는 줌아웃, 회전, 원근 모션 (perspective motion), 또는 다른 불규칙한 모션 타입들과 같은 비-병진 모션을 나타내는 2 이상의 모션 벡터들을 결정할 수도 있다.

[0105] 인트라-예측을 수행하기 위해, 비디오 인코더 (200) 는 예측 블록을 생성하기 위해 인트라-예측 모드를 선택할 수도 있다. VVC 는 평면 모드 및 DC 모드 뿐만 아니라, 다양한 방향성 모드들을 포함한, 67 개의 인트라-예측 모드들을 제공한다. 일반적으로, 비디오 인코더 (200) 는 현재 블록의 샘플들을 예측하기 위한 현재 블록 (예를 들어, CU 의 블록) 에 대해 이웃하는 샘플들을 기술하는 인트라-예측 모드를 선택한다. 이러한 샘플들은 일반적으로, 비디오 인코더 (200) 가 래스터 스캔 순서로 (좌측에서 우측으로, 상단에서 하단으로) CTU 들 및 CU들을 코딩한다고 가정하여, 현재 블록과 동일한 픽처에서 현재 블록의 상측, 상측 및 좌측으로, 또는 좌측으로 있을 수도 있다.

[0106] 비디오 인코더 (200) 는 현재 블록에 대한 예측 모드를 나타내는 데이터를 인코딩한다. 예를 들어, 인터-예측 모드들에 대해, 비디오 인코더 (200) 는 이용가능한 다양한 인터-예측 모드들 중 어느 것이 사용되는지를 나타내는 데이터 뿐만 아니라, 대응하는 모드에 대한 모션 정보를 인코딩할 수도 있다. 단방향 또는 양방향 인터-예측을 위해, 예를 들어 비디오 인코더 (200) 는 어드밴스드 모션 벡터 예측 (AMVP) 또는 병합 모드를 사용하여 모션 벡터들을 인코딩할 수도 있다. 비디오 인코더 (200) 는 유사한 모드들을 사용하여 아핀 모션 보상 모드에 대한 모션 벡터들을 인코딩할 수도 있다.

[0107] 블록의 인트라-예측 또는 인터-예측과 같은 예측에 후속하여, 비디오 인코더 (200) 는 블록에 대한 잔차 데이터를 계산할 수도 있다. 잔차 블록과 같은 잔차 데이터는 대응하는 예측 모드를 사용하여 형성된, 블록과 블록에 대한 예측 블록 사이의 샘플 별 차이들을 나타낸다. 비디오 인코더 (200) 는 샘플 도메인 대신 변환 도메인에서 변환된 데이터를 생성하기 위해, 하나 이상의 변환들을 잔차 블록에 적용할 수도 있다. 예를 들어, 비디오 인코더 (200) 는 이산 코사인 변환 (DCT), 정수 변환, 웨이블릿 변환, 또는 개념적으로 유사한 변환을 잔차 비디오 데이터에 적용할 수도 있다. 부가적으로, 비디오 인코더 (200) 는 MDNSST (mode-dependent non-separable secondary transform), 신호 의존적 변환, Karhunen-Loeve 변환 (KLT) 등과 같은, 제 1 변환에 후속하는 2차 변환을 적용할 수도 있다. 비디오 인코더 (200) 는 하나 이상의 변환들의 적용에 후속하여 변

환 계수들을 생성한다.

[0108]

본 개시의 기법들에 따라, 비디오 인코더 (200) 는 현재 블록에 대한 잔차 블록에 적용할 특정 타입의 변환 (또는 다중 변환) 을 결정할 수도 있다. 결정된 변환의 타입은 1차 변환을 포함할 수도 있으며, 이는 수평 변환 및 수직 변환을 포함하는 분리가능한 변환일 수도 있다. 일부 예들에서, 결정된 변환의 타입은 2차 변환 (예를 들어, 분리불가능한 변환) 을 더 포함할 수도 있다. 비디오 인코더 (200) 는 선택된 변환의 타입이 2차 변환을 포함하는지 여부 및 1차 변환을 나타내는 선택된 변환의 타입을 나타내는 제 1 코드워드를 인코딩할 수도 있다. 제 1 코드워드가 선택된 변환의 타입이 2차 변환을 포함한다는 것을 표시하는 경우, 비디오 인코더 (200) 는 가용 2차 변환들의 세트의 선택된 2차 변환을 나타내는 제 2 코드워드를 추가로 인코딩할 수도 있다. 게다가, 비디오 인코더 (200) 는 1차 변환 및 2차 변환 양자 모두를 적용할 수도 있다. 이러한 코드워드들의 조합의 예는 표 1 내지 표 12 및 도 6 내지 도 8 과 관련하여 하기에서 더 상세히 설명된다.

[0109]

위에 언급된 바와 같이, 변환 계수들을 생성하기 위한 임의의 변환들에 후속하여, 비디오 인코더 (200) 는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 일반적으로 변환 계수들이 가능하게는 그 계수들을 나타내는데 사용된 데이터의 양을 감소시키도록 양자화되어, 추가 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스를 수행함으로써, 비디오 인코더 (200) 는 계수들의 일부 또는 모두와 연관된 비트 깊이를 감소시킬 수도 있다. 예를 들어, 비디오 인코더 (200) 는 양자화 동안 n -비트 값은 m -비트 값으로 라운딩 다운할 수도 있으며, 여기서 n 은 m 보다 크다. 일부 예들에서, 양자화를 수행하기 위해, 비디오 인코더 (200) 는 양자화될 값의 비트단위 우측-시프트를 수행할 수도 있다.

[0110]

양자화에 후속하여, 비디오 인코더 (200) 는 변환 계수들을 스캔하여, 양자화된 변환 계수들을 포함하는 2차원 행렬로부터 1차원 벡터를 생성할 수도 있다. 스캔은 더 높은 에너지 (및 따라서 더 낮은 주파수) 계수들을 벡터의 전방에 배치하고 그리고 더 낮은 에너지 (및 따라서 더 높은 주파수) 변환 계수들을 벡터의 후방에 배치하도록 설계될 수도 있다. 일부 예들에 있어서, 비디오 인코더 (200) 는 양자화된 변환 계수들을 스캐닝하기 위해 미리정의된 스캔 순서를 활용하여 직렬화된 벡터를 생성하고, 그 후 벡터의 양자화된 변환 계수들을 엔트로피 인코딩할 수도 있다. 다른 예들에서, 비디오 인코더 (200) 는 적응적 스캔을 수행할 수도 있다. 1차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캐닝한 후, 비디오 인코더 (200) 는 예를 들어, 컨텍스트 적응적 이진 산술 코딩 (CABAC) 에 따라 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (200) 는 또한, 비디오 데이터를 디코딩하는데 있어서 비디오 디코더 (300) 에 의한 사용을 위해 인코딩된 비디오 데이터와 연관된 메타데이터를 기술하는 신택스 엘리먼트들에 대한 값을 엔트로피 인코딩할 수도 있다.

[0111]

CABAC 을 수행하기 위해, 비디오 인코더 (200) 는 송신될 심볼에 컨텍스트 모델 내의 컨텍스트를 할당할 수도 있다. 컨텍스트는 예를 들어, 심볼의 이웃하는 값들이 제로 값인지 여부와 관련될 수도 있다. 확률 결정은 심볼에 할당된 컨텍스트에 기초할 수도 있다.

[0112]

비디오 인코더 (200) 는 신택스 데이터, 예컨대 블록-기반 신택스 데이터, 픽처-기반 신택스 데이터, 및 시퀀스-기반 신택스 데이터를, 비디오 디코더 (300) 에, 예를 들어, 픽처 헤더, 블록 헤더, 슬라이스 헤더, 또는 다른 신택스 데이터, 예컨대 시퀀스 파라미터 세트 (SPS), 픽처 파라미터 세트 (PPS), 또는 비디오 파라미터 세트 (VPS) 에서 추가로 생성할 수도 있다. 비디오 디코더 (300) 는 마찬가지로 대응하는 비디오 데이터를 디코딩하는 방법을 결정하기 위해 그러한 신택스 데이터를 디코딩할 수도 있다.

[0113]

이러한 방식으로, 비디오 인코더 (200) 는 인코딩된 비디오 데이터, 예를 들어 픽처의 블록들 (예를 들어, CU들) 로의 파티셔닝을 기술하는 신택스 엘리먼트들 및 블록들에 대한 예측 및/또는 잔차 정보를 포함하는 비트 스트림을 생성할 수도 있다. 궁극적으로, 비디오 디코더 (300) 는 비트스트림을 수신하고 인코딩된 비디오 데이터를 디코딩할 수도 있다.

[0114]

일반적으로, 비디오 디코더 (300) 는 비트스트림의 인코딩된 비디오 데이터를 디코딩하기 위해 비디오 인코더 (200) 에 의해 수행되는 것과 상반되는 프로세스를 수행한다. 예를 들어, 비디오 디코더 (300) 는 비디오 인코더 (200) 의 CABAC 인코딩 프로세스와 실질적으로 유사하지만, 상반되는 방식으로 CABAC 을 이용하여 비트 스트림의 신택스 엘리먼트들에 대한 값을 디코딩할 수도 있다. 신택스 엘리먼트들은 픽처의 CTU들로의 파티셔닝 정보, 및 QBT 구조와 같은 대응하는 파티션 구조에 따라 각각의 CTU 의 파티셔닝을 정의하여, CTU 의 CU들을 정의할 수도 있다. 신택스 엘리먼트들은 추가로 비디오 데이터의 블록들 (예를 들어, CU들) 에 대한 예측 및 잔차 정보를 정의할 수도 있다.

[0115]

잔차 정보는 예를 들어, 양자화된 변환 계수들로 표현될 수도 있다. 비디오 디코더 (300) 는 블록에 대한

잔차 블록을 재생하기 위해 블록의 양자화된 변환 계수들을 역 양자화 및 역 변환할 수도 있다.

[0116] 본 개시의 기법들에 따라, 비디오 디코더 (300) 는 비디오 데이터의 현재 블록에 대한 디코딩된 변환 계수들에 적용될 변환의 타입을 나타내는 제 1 코드워드를 디코딩할 수도 있다. 위에서 논의된 바와 같이, 변환의 타입은 1차 변환을 나타낼 수도 있으며, 이는 수평 변환 및 수직 변환을 포함하는 분리가능한 변환일 수도 있다.

변환의 타입은 2차 변환을 더 포함할 수도 있다. 변환의 타입이 2차 변환을 포함하는 경우, 비디오 디코더 (300) 는 가용 2차 변환들의 세트에 포함될 수도 있는 2차 변환을 나타내는 제 2 코드워드를 디코딩할 수도 있다. 비디오 디코더 (300) 는 그 후 변환 계수들의 중간 세트를 생성하기 위해 디코딩된 변환 계수들에 2차 변환을 적용할 수도 있고, 그 후 현재 블록에 대한 잔차 블록을 재생하기 위해 변환 계수들의 중간 세트에 1차 변환을 적용할 수도 있다.

[0117] 비디오 디코더 (300) 는 시그널링된 예측 모드 (인트라-예측 또는 인터-예측) 및 관련된 예측 정보 (예를 들어, 인터-예측을 위한 모션 정보) 를 사용하여 블록에 대한 예측 블록을 형성한다. 비디오 디코더 (300) 는 그 후 예측 블록과 잔차 블록을 (샘플 단위로) 조합하여 원래의 블록을 재생할 수도 있다. 비디오 디코더 (300) 는 블록의 경계들을 따라 시각적 아티팩트들을 감소시키기 위해 디블로킹 프로세스를 수행하는 것과 같은, 부가 프로세싱을 수행할 수도 있다.

[0118] 위에 언급된 바와 같이, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 CABAC 인코딩 및 디코딩을 신팩스 엘리먼트들의 값들에 적용할 수도 있다. CABAC 인코딩을 신팩스 엘리먼트에 적용하기 위해, 비디오 인코더 (200) 는 신팩스 엘리먼트의 값을 이진화하여 "빈들"로서 지정되는 일련의 하나 이상의 비트들을 형성할 수도 있다. 또한, 비디오 인코더 (200) 는 코딩 컨텍스트를 식별할 수도 있다. 코딩 컨텍스트는 특정 값을 갖는 빈들의 확률을 식별할 수도 있다. 예를 들어, 코딩 컨텍스트는 0-값의 빈을 코딩하는 0.7 확률 및 1-값의 빈을 코딩하는 0.3 확률을 표시할 수도 있다. 코딩 컨텍스트를 식별한 후, 비디오 인코더 (200) 는 인터벌을 하위 서브-인터벌 및 상위 서브-인터벌로 분할할 수도 있다. 하위-인터벌들 중 하나는 값 0 과 연관될 수도 있고 다른 하위-인터벌은 값 1 과 연관될 수도 있다.

[0119] 서브 간격들의 폭은 식별된 코딩 컨텍스트에 의해 연관된 값들에 대해 표시된 확률에 비례할 수도 있다. 신팩스 엘리먼트의 빈이 하위 서브-인터벌과 연관된 값을 갖는 경우, 인코딩된 값은 하위 서브-인터벌의 하위 경계와 동일할 수도 있다. 신팩스 엘리먼트의 동일한 빈이 상위 서브-인터벌과 연관된 값을 갖는 경우, 인코딩된 값은 상위 서브-인터벌의 하위 경계와 동일할 수도 있다. 신팩스 엘리먼트의 다음 빈을 인코딩하기 위해, 비디오 인코더 (200) 는 인코딩된 비트의 값을 연관된 서브-인터벌인 인터벌로 이러한 단계들을 반복할 수도 있다. 비디오 인코더 (200) 가 다음 빈에 대해 이러한 단계들을 반복할 때, 비디오 인코더 (200) 는 인코딩된 빈들의 실제 값 및 식별된 코딩 컨텍스트에 의해 표시된 확률에 기초하여 수정된 확률을 사용할 수도 있다.

[0120] 비디오 디코더 (300) 가 신팩스 엘리먼트의 값에 대해 CABAC 디코딩을 수행할 때, 비디오 디코더 (300) 는 코딩 컨텍스트를 식별할 수도 있다. 비디오 디코더 (300) 는 그 후 인터벌을 하위 서브-인터벌 및 상위 서브-인터벌로 분할할 수도 있다. 하위-인터벌들 중 하나는 값 0 과 연관될 수도 있고 다른 하위-인터벌은 값 1 과 연관될 수도 있다. 서브 간격들의 폭은 식별된 코딩 컨텍스트에 의해 연관된 값들에 대해 표시된 확률에 비례할 수도 있다. 인코딩된 값이 하위 서브-인터벌 내에 있는 경우, 비디오 디코더 (300) 는 하위 서브-인터벌과 연관된 값을 갖는 빈을 디코딩할 수도 있다. 인코딩된 값이 상위 서브-인터벌 내에 있는 경우, 비디오 디코더 (300) 는 상위 서브-인터벌과 연관된 값을 갖는 빈을 디코딩할 수도 있다. 신팩스 엘리먼트의 다음 빈을 디코딩하기 위해, 비디오 디코더 (300) 는 인코딩된 값을 포함하는 서브-인터벌인 인터벌로 이러한 단계들을 반복할 수도 있다. 비디오 디코더 (300) 가 다음 빈에 대해 이러한 단계들을 반복할 때, 비디오 디코더 (300) 는 디코딩된 빈들 및 식별된 코딩 컨텍스트에 의해 표시된 확률에 기초하여 수정된 확률을 사용할 수도 있다. 비디오 디코더 (300) 는 그 후 신팩스 엘리먼트의 값을 복구하기 위해 빈들을 역 이진화할 수도 있다.

[0121] HEVC 이전의 비디오 코딩 표준들에서는, 고정된 분리가능한 변환만이 사용되며, 여기서 DCT-2 는 수직 및 수평 양자 모두로 사용된다. HEVC 에서, DCT-2 에 부가하여, DST-7 은 또한 고정된 분리가능한 변환으로서 4x4 블록들에 대해 채용된다.

[0122] 미국 특허 10,306,229, 미국 특허 공개 2018/0020218 및 미국 가특허 출원 62/679,570 은 다중 변환 선택 (MTS) 기법들을 설명한다. MTS 는 이전에 적용된 다중 변환 (Adaptive Multiple Transforms; AMT) 으로 불렸다. 미국 가특허출원 62/679,570 에서 MTS 의 예는 JVET (Joint Video Experts Team) 의 JEM-7.0 (Joint

Experimental Model) 에서 채택되었고 이후 MTS 의 단순화된 버전이 VVC 에 채택된다.

[0123] 본 개시는 일반적으로 신택스 엘리먼트들과 같은, 소정의 정보의 "시그널링" 을 언급할 수도 있다. 용어 "시그널링" 은 일반적으로, 인코딩된 비디오 데이터를 디코딩하는데 사용된 신택스 엘리먼트들 및/또는 다른 데이터 값들의 통신을 지칭할 수도 있다. 즉, 비디오 인코더 (200) 는 비트스트림에서 신택스 엘리먼트들에 대한 값을 시그널링할 수도 있다. 일반적으로, 시그널링은 비트스트림에서 값을 생성하는 것을 지칭한다.

위에 언급된 바와 같이, 소스 디바이스 (102) 는 목적지 디바이스 (116) 에 의한 나중 취출을 위해 저장 디바이스 (112) 에 신택스 엘리먼트를 저장할 때 발생할 수 있는 것과 같은, 실질적으로 실시간으로 또는 비실시간으로, 비트스트림을 목적지 디바이스 (116) 로 전송할 수도 있다.

[0124] 도 2a 및 도 2b 는 예시의 큐드트리 이진 트리 (QTBT) 구조 (130), 및 대응하는 코딩 트리 유닛 (CTU) (132) 을 도시하는 개념적 다이어그램이다. 실선들은 큐드트리 분할을 나타내고, 점선들은 이진 트리 분할을 나타낸다. 이진 트리의 각각의 분할된 (즉, 비-리프) 노드에서, 어느 분할 타입 (즉, 수평 또는 수직) 이 사용되는지를 나타내기 위해 하나의 플래그가 시그널링되며, 여기서 0 은 수평 분할을 표시하고 1 은 수직 분할을 표시한다. 큐드트리 스플릿팅에 대해, 스플릿팅 타입을 표시할 필요는 없는데, 이는 큐드트리 노드들이 동일한 사이즈를 가진 4 개의 서브-블록들로 수평으로 및 수직으로 블록을 스플릿팅하기 때문이다. 이에 따라, QTBT 구조 (130) 의 영역 트리 레벨 (즉, 실선들) 에 대한 신택스 엘리먼트들 (이를 템면 분할 정보) 및 QTBT 구조 (130) 의 예측 트리 레벨 (즉, 점선들) 에 대한 신택스 엘리먼트들 (이를 템면 분할 정보) 을, 비디오 인코더 (200) 가 인코딩할 수도 있고, 비디오 디코더 (300) 가 디코딩할 수도 있다. QTBT 구조 (130) 의 종단 리프 노드들에 의해 표현된 CU들에 대해, 예측 및 변환 데이터와 같은 비디오 데이터를, 비디오 인코더 (200) 가 인코딩할 수도 있고, 비디오 디코더 (300) 가 디코딩할 수도 있다.

[0125] 일반적으로, 도 2b 의 CTU (132) 는 제 1 및 제 2 레벨들에서 QTBT 구조 (130) 의 노드들에 대응하는 블록들의 사이즈들을 정의하는 파라미터들과 연관될 수도 있다. 이들 파라미터들은 CTU 사이즈 (샘플들에서 CTU (132) 의 사이즈를 나타냄), 최소 큐드트리 사이즈 (최소 허용된 큐드트리 리프 노드 사이즈를 나타내는 MinQTSIZE), 최대 이진 트리 사이즈 (최대 허용된 이진 트리 루트 노드 사이즈를 나타내는 MaxBTSIZE), 최대 이진 트리 깊이 (최대 허용된 이진 트리 깊이를 나타내는 MaxBTDepth), 및 최소 이진 트리 사이즈 (최소 허용된 이진 트리 리프 노드 사이즈를 나타내는 MinBTSIZE) 를 포함할 수도 있다.

[0126] CTU 에 대응하는 QTBT 구조의 루트 노드는 QTBT 구조의 제 1 레벨에서 4 개의 자식 노드들을 가질 수도 있고, 이들의 각각은 큐드트리 파티셔닝에 따라 파티셔닝될 수도 있다. 즉, 제 1 레벨의 노드들은 리프 노드들 (자식 노드들이 없음) 이거나 또는 4 개의 자식 노드들을 갖는다. QTBT 구조 (130) 의 예는 그러한 노드들을 브랜치들에 대한 실선들을 갖는 부모 노드 및 자식 노드들을 포함하는 것으로서 나타낸다. 최대 허용된 이진 트리 루트 노드 사이즈 (MaxBTSIZE) 보다 더 크지 않은 제 1 레벨의 노드들은 개개의 이진 트리들에 의해 추가로 파티셔닝될 수 있다. 하나의 노드의 이진 트리 분할은 스플릿트로부터 발생하는 노드들이 최소 허용된 이진 트리 리프 노드 사이즈 (MinBTSIZE) 또는 최대 허용된 이진 트리 깊이 (MaxBTDepth) 에 도달할 때까지 반복될 수 있다. QTBT 구조 (130) 의 예는 그러한 노드들을 브랜치들에 대한 점선들을 갖는 것으로서 나타낸다. 이진 트리 리프 노드는 임의의 추가 파티셔닝 없이, 예측 (예를 들어, 인트라-픽처 또는 인터-픽처 예측) 및 변환을 위해 사용되는 코딩 유닛 (CU) 으로 지정된다. 위에 논의된 바와 같이, CU들은 또한, "비디오 블록들" 또는 "블록들" 로 지칭될 수도 있다.

[0127] QTBT 파티셔닝 구조의 하나의 예에서, CTU 사이즈는 128x128 (루마 샘플들 및 2 개의 대응하는 64x64 크로마 샘플들) 로서 설정되고, MinQTSIZE 는 16x16 으로서 설정되고, MaxBTSIZE 는 64x64 로서 설정되고, (폭 및 높이 양자 모두에 대한) MinBTSIZE 는 4 로서 설정되고, 그리고 MaxBTDepth 는 4 로서 설정된다. 큐드트리 파티셔닝은 큐드트리 리프 노드들을 생성하기 위해 먼저 CTU 에 적용된다. 큐드트리 리프 노드들은 16×16 (즉, MinQTSIZE) 으로부터 128×128 (즉, CTU 사이즈) 까지의 사이즈를 가질 수도 있다. 리프 큐드트리 노드가 128×128 인 경우, 사이즈가 MaxBTSIZE (즉, 이 예에서는 64×64) 를 초과하기 때문에 그것은 이진 트리에 의해 추가로 스플릿팅되지 않을 것이다. 그렇지 않으면, 리프 큐드트리 노드는 이진 트리에 의해 추가로 파티셔닝될 것이다. 따라서, 큐드트리 리프 노드는 또한 이진 트리에 대한 루트 노드이고 이진 트리 깊이를 0 으로서 갖는다. 이진 트리 깊이가 MaxBTDepth (이 예에서는 4) 에 도달할 때, 추가의 스플릿팅이 허용되지 않는다. 이진 트리 노드가 MinBTSIZE (이 예에서는 4) 와 동일한 폭을 가질 때, 그것은 추가의 수평 스플릿팅이 허용되지 않음을 암시한다. 유사하게, 높이가 MinBTSIZE 와 동일한 이진 트리 노드는 그 이진 트리 노드에 대해 추가의 수직 분할이 허용되지 않음을 암시한다. 위에 언급된 바와 같이, 이진 트리의 리프 노드들

은 CU들로 지칭되고, 추가 파티셔닝 없이 예측 및 변환에 따라 추가로 프로세싱된다.

[0128] 도 3a 및 도 3b 는 HEVC 의 잔차 쿼드트리에 기초한 예시의 변환 스킵을 도시하는 개념적 다이어그램들이다.

HEVC 에서 잔차 쿼드트리 (RQT) 를 사용하는 변환 코딩 구조는 www.hhi.fraunhofer.de/fields-of-competence/image-processing/research-groups/image-video-coding/hevc-high-efficiency-video-coding/transform-coding-using-the-residual-quadtrees-rqt.html로부터 채택된, 다음과 같이 간략하게 설명되는 잔차 블록들의 다양한 특징들을 채택하는데 적용된다.

[0129] HEVC 에서, 각각의 꽈치는 특정 타일 또는 슬라이스에 대해 레스터 스캔 순서로 코딩되는 코딩 트리 유닛 (CTU) 들로 나눠진다. CTU 는 정사각형 블록이며 쿼드트리, 즉 코딩 트리의 루트를 나타낸다. CTU 사이즈는 8x8 내지 64x64 의 루마 샘플들 범위일 수도 있지만, 통상적으로 64x64 가 사용된다. 각각의 CTU 는 추가로 코딩 유닛 (CU) 들이라고 하는 더 작은 정사각형 블록들로 분할될 수도 있다.

[0130] CTU 가 재귀적으로 CU 로 분할된 후, 각각의 CU 는 추가로 예측 유닛 (PU) 들 및 변환 유닛 (TU) 들로 나눠진다. TU들로의 CU 의 파티셔닝은 쿼드트리 접근법에 기초하여 재귀적으로 수행되며, 이에 따라 각각의 CU 의 잔차 신호는 트리 구조 즉, 잔차 쿼드트리 (RQT) 에 의해 코딩된다. RQT 는 4x4 에서 최대 32x32 루마 샘플들까지 TU 사이즈들을 허용한다.

[0131] 도 3a 는 CU 가 문자들 a 내지 j 로 라벨링된, 10 개의 TU들 및 대응하는 블록 파티셔닝을 포함하는 예를 도시한다. 도 3b 에 나타낸 RQT 의 각각의 노드는 실제로 도 3a 에 대응하는 변환 유닛 (TU) 이다. 개별 TU 들은 깊이-우선 순회로 재귀적 Z-스캔에 후속하는, 알파벳 순서로 도 3a 에 도시되는, 깊이 우선 트리 순회 순서로 프로세싱된다. 쿼드트리 접근법은 잔차 신호의 다양한 공간 주파수 특징들에 대한 변환의 적응을 가능하게 한다.

[0132] 통상적으로, 더 큰 공간적 지원을 갖는 더 큰 변환 블록 사이즈들은 우수한 주파수 해상도를 제공한다. 그러나, 더 큰 공간적 지원을 갖는 더 작은 변환 블록 사이즈들은 우수한 공간적 해상도를 제공한다. 2 개의 공간 해상도와 주파수 해상도 사이의 트레이드-오프는 예를 들어, 레이트-왜곡 최적화 기법에 기초하여, 인코더 모드 판정에 의해 선정된다. 레이트 왜곡 최적화 기법은 각각의 코딩 모드 (예를 들어, 특정 RQT 분할 구조) 에 대해, 코딩 비트들 및 복원 왜곡의 각종 합, 즉 레이트 왜곡 비용을 계산하고, 최상의 모드로서 최소 레이트 왜곡 비용으로 코딩 모드를 선택한다.

[0133] HEVC 마다 RQT 에서는 3 개의 파라미터들: 트리의 최대 깊이, 허용된 최소 변환 사이즈 및 허용된 최대 변환 사이즈가 정의된다. 최소 및 최대 변환 사이즈들은 이전 단락에서 언급된 지원된 블록 변환들에 대응하는 4x4 부터 32x32 샘플들 범위 내에서 달라질 수 있다. RQT 의 허용된 최대 깊이는 TU들의 수를 제한한다. 0 과 같은 최대 깊이는 각각의 포함된 TB 가 허용된 최대 변환 사이즈, 예를 들어 32x32 에 도달하는 경우 CB 가 이 더 이상 분할될 수 없음을 의미한다.

[0134] 이러한 모든 파라미터들은 HEVC 에서 RQT 구조와 상호작용하며 이에 영향을 미친다. 루트 CB 사이즈가 64x64 이고 최대 깊이가 0 과 같고 최대 변환 사이즈가 32x32 인 경우를 고려한다. 이 경우, CB 는 적어도 한번 파티셔닝되어야 하는데, 이는 그렇지 않으면 이것이 허용되지 않는 64×64 TB 로 이어질 것이기 때문이다. RQT 파라미터들, 즉 최대 RQT 깊이, 최소 및 최대 변환 사이즈는, HEVC 마다, 시퀀스 파라미터 세트 레벨로 비트스트림에서 송신된다. RQT 깊이에 관하여, 인트라 및 인터 코딩된 CU들에 대해 상이한 값들이 특정되고 시그널링될 수 있다.

[0135] 쿼드트리 변환은 HEVC 에서 인트라 및 인터 잔차 블록들 모두에 대해 적용된다. 통상적으로, 현재 잔차 쿼드트리 파티션과 동일한 사이즈의 DCT-II 변환이 잔차 블록에 대해 적용된다. 그러나, 현재 잔차 쿼드트리 블록이 4x4 이고 인트라-예측에 의해 생성되는 경우, 위의 4x4 DST-VII 변환들이 적용된다.

[0136] HEVC 에서, 더 큰 사이즈의 변환들, 예를 들어 64x64 변환은 주로 그의 제한된 이익을 고려하고 상대적으로 작은 해상도 비디오들에 대한 상대적으로 높은 복잡성으로 인해 채택되지 않는다.

[0137] 도 4 는 적응적 변환 선택으로 하이브리드 비디오 인코딩을 위한 예시의 시스템 (140) 을 도시하는 블록 다이어그램이다. 본 개시의 기법들은 이러한 시스템 또는 대응하는 디코딩 시스템에 의해 수행될 수도 있다. 일반적으로, 본 개시의 기법들은 적응적 변환 코딩 스킵에 적용가능하며, 여기서 예측 잔차들의 각각의 블록에 대해, 상이한 변환들이 비디오 인코더에 의해 선택되고, 사이드 정보로서 시그널링되며, 사이드 정보를 사용하여 비디오 디코더에 의해 결정될 수 있다.

- [0138] 도 4 의 시스템 (104) 은 블록 분리 유닛 (142), 블록 예측 유닛 (144), 잔차 생성 유닛 (146), 블록 변환 유닛 (148), 변환 뱅크 (150), 양자화 유닛 (152), 엔트로피 인코딩 유닛 (154), 역 양자화 유닛 (156), 역 블록 변환 유닛 (158), 블록 복원 유닛 (160), 및 프레임 버퍼 (162) 를 포함한다. 블록 분리 유닛 (142) 은 일반적으로 원시의 코딩되지 않은 원시 비디오 데이터를 수신하고 비디오 데이터의 팩처를 블록으로 분할한다. 블록 예측 유닛 (144) 은 인코딩될 비디오 데이터의 현재 블록에 대한 예측 블록을 생성한다. 블록 분리 유닛 (142) 은 현재 블록을 잔차 생성 유닛 (146) 에 제공하고 블록 예측 유닛 (144) 은 예측 블록을 잔차 생성 유닛 (146) 에 제공한다. 잔차 생성 유닛 (146) 은 잔차 블록 (r) 을 생성하고 잔차 블록을 블록 변환 유닛 (148) 에 제공한다.
- [0139] 블록 변환 유닛 (148) 은 변환 뱅크 (150) 로부터 하나 이상의 변환들을 선택한다. 예를 들어, 본 개시의 기법들에 따라, 변환 뱅크 (150) 는 하나 이상의 1차 변환들 (예를 들어, 분리가능한 변환들) 및 하나 이상의 2차 변환들 (예를 들어, 분리불가능한 변환들) 을 포함할 수도 있다. 그 후 블록 변환 유닛 (148) 은 1차 변환 및 적용가능한 경우 2차 변환을 적용하여 변환 계수들을 생성할 수도 있다. 또한, 블록 변환 유닛 (148) 은 변환(들) 의 표시 (t) 를 엔트로피 인코딩 유닛 (154) 에 전송할 수도 있다. 블록 변환 유닛 (148) 은 변환 계수들 ($T^{(1)}r$) 을 양자화 유닛 (152) 에 제공한다.
- [0140] 양자화 유닛 (152) 은 예를 들어, 현재 블록에 대한 양자화 파라미터 (QP) 에 따라 변환 계수들의 비트 깊이를 감소시킴으로써 변환 계수들을 양자화한다. 양자화 유닛 (152) 은 양자화된 변환 계수들을 엔트로피 코딩 유닛 (154) 및 역 양자화 유닛 (156) 에 제공한다.
- [0141] 엔트로피 인코딩 유닛 (154) 은 변환 (t) 및 양자화된 변환 계수들의 표시를 포함하는, 싱택스 엘리먼트들에 대한 값들의 엔트로피 인코딩을 수행한다. 본 개시의 기법들에 따라, 엔트로피 인코딩 유닛 (154) 은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스킴을 나타내는 제 1 코드워드를 인코딩할 수도 있다. 선택된 변환 스킴은 1차 변환을, 그리고 일부 예들에서는 1차 변환에 부가하여 적용될 2차 변환을 포함할 수도 있다. 선택된 변환 스킴이 2차 변환을 포함하는 경우, 엔트로피 인코딩 유닛 (154) 은 가용 2차 변환들의 세트에서 2차 변환을 나타내는 제 2 코드워드를 인코딩할 수도 있다. 엔트로피 인코딩 유닛 (154) 은 인코딩된 비디오 비트스트림에 엔트로피 인코딩된 데이터 (예를 들어, 제 1 및/또는 제 2 코드워드들 및 양자화된 변환 계수들에 대한 엔트로피 인코딩된 싱택스 엘리먼트들) 을 포함할 수도 있다.
- [0142] 역 양자화 유닛 (156) 은 양자화된 변환 계수들을 역 양자화하고 결과의 변환 계수들을 역 블록 변환 유닛 (158) 에 전달할 수도 있다. 역 블록 변환 유닛 (158) 은 1차 변환을, 그리고 적용가능한 경우, 2차 변환을 변환 계수들에 적용하여 잔차 블록을 재생할 수도 있다. 역 블록 변환 유닛 (158) 은 잔차 블록을 블록 복원 유닛 (160) 에 제공할 수도 있으며, 이는 잔차 블록을 예측 블록과 조합하여 복원된 블록을 생성하고, 복원된 블록을 프레임 버퍼 (162) 에 저장할 수도 있다. 프레임 버퍼 (162) 는 또한 디코딩된 팩처 버퍼 (DPB) 로서 지칭될 수도 있다.
- [0143] 도 4 의 다양한 컴포넌트들의 각각은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합에서 구현될 수도 있다. 소프트웨어 또는 펌웨어로 구현될 때, 다양한 동작들을 위한 명령들이 메모리에 저장되고 하나 이상의 프로세싱 유닛들에 의해 실행될 수도 있다. 프로세싱 유닛들 및 메모리는 회로부에서 구현될 수도 있다. 프로세싱 유닛들은 예를 들어, 하나 이상의 디지털 신호 프로세서 (DSP), 범용 마이크로프로세서, 주문형 집적 회로 (ASIC), 필드 프로그램가능 게이트 어레이 (FPGA), 또는 다른 등가 집적 또는 이산 로직 회로부를, 임의의 조합으로 포함할 수도 있다.
- [0144] 이러한 방식으로, 도 4 의 시스템 (140) 은 비디오 데이터를 저장하도록 구성된 메모리; 및 회로부에서 구현되는 하나 이상의 프로세서들을 포함하는 비디오 인코더의 예를 나타내고, 하나 이상의 프로세서들은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스kim을 나타내는 제 1 코드워드를 코딩하는 것으로서, 선택된 변환 스kim은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하고, 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하며; 그리고 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하도록 구성된다.
- [0145] 도 5a 및 도 5b 는 별도의 변환 구현으로서 수평 및 수직 변환들을 도시하는 개념적 다이어그램이다. 특히, 잔차 값들의 수평 및 수직 라인들은 수평 및 수직 변환들을 사용하여 독립적으로 변환될 수도 있다 (예를 들어, 계산 복잡성을 감소시키기 위해, 블록 변환들은 분리가능한 방식으로 계산될 수도 있다).

- [0146] HEVC 이전의 비디오 코딩 표준들에서는, 고정된 분리가능한 변환만이 사용되며, 여기서 DCT-2 는 수직 및 수평 양자 모두로 사용된다. HEVC 에서, DCT-2 에 부가하여, DST-7 은 또한 고정된 분리가능한 변환으로서 4x4 블록들에 대해 채용된다. 미국 특허 출원 제 15/005,736 및 15/649,612 호는 이러한 고정 변환의 적용 확장을 기재하며, MTS (또한 적응 다중 변환 (AMT) 으로 지칭됨) 의 예는 2016 년 1 월 25 일 출원된 미국 특허 출원 제 15/005,736 호; 2017 년 7 월 13 일 출원된 15/649,612 호에 기재되어 있으며; 및 2018 년 6 월 1 일 출원된 62/679,570 호는 JVET (Joint Video Experts Team) (jvet.hhi.fraunhofer.de/svn/svn_HMJEMSoftware/tags/HM-16.6-JEM-7.0 에서 입수가능한, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11, JEM 소프트웨어의 JVET (Joint Video Experts Team))의 JEM(Joint Experimental Model) 에서 채택되었다.
- [0147] 도 6 은 2 개의 변환들을 식별하는데 사용된 MTS 시그널링의 예를 나타내는 개념적 다이어그램이다. VTM 의 현재 버전에서 (phenix.it-sudparis.eu/jvet/doc_end_user/documents/13_Marrakech/wg11/JVET-M1001-v7.zip 에서 입수가능한, Versatile Video Coding (Draft 4), Joint Video Experts Team (JVET), ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 13th Meeting: Marrakech, MA, 9-18 Jan. 2019, Document JVET-M1001), 다중 변환 후보들은 도 6 및 도 7 에서 이진 트리를 연접함으로써 예시될 수 있는, 잘려진 단항 이잔화에 기초하여 시그널링된다. 그 후, 변환 후보들은 연접 (concatenation) 에 의해 획득된 코드워드들과 연관된다.
- [0148] 도 7 은 일 예의 변환 할당 및 대응하는 단항 코드워드를 도시하는 개념적 다이어그램이다. VVC 의 현재 버전에서 MTS 시그널링은 도 6 에서 이진 트리를 연접함으로써 획득된 코드워드에 변환을 할당하는 것을 포함하며, 여기서 "H: DCT-8, V: DST-7" 은 분리가능한 변환에 대해 DCT-8 이 수평으로 적용되고 DST-7 이 수직으로 적용되는 것을 의미하고, IDT 는 1-D 아이덴티티 변환 (스케일링 수행) 을 표기한다.
- [0149] VVC 의 MTS (다중 변환 선택) 설계는 6 개의 변환 후보들을 사용하며 (도 7 에서와 같이), 수평 및 수직 방향 양자 모두에서 단일 타입의 변환을 사용하는 것 외에 DST-7 및 DCT-8 과의 조합을 지원한다 (즉, IDT, DCT-2 및 DST-7 을 수평 및 수직으로 적용). 실제로, 우수한 코딩 효율은 더 많은 수의 후보들을 허용함으로써 달성될 수 있다. 본 개시는 코딩 효율을 개선할 수도 있는 현재 MTS 설계의 다양한 확장을 설명한다.
- [0150] MTS 스킴은 특정된 시그널링 방법의 코드워드에 변환을 할당함으로써 정의될 수도 있다. 비디오 인코더 (200) 및/또는 비디오 디코더 (300) 는 위에서 그리고 하기에서 더 상세히 논의된 바와 같이, 본 개시의 기법들에 따라 구성될 수도 있다. 특히, 본 개시에 따른 MTS 스킴은 특정된 시그널링 방법의 코드워드에 변환을 할당함으로써 정의될 수도 있다. 따라서, MTS 스킴은 다음을 특정함으로써 완전히 정의될 수도 있다: (i) 단일 세트 또는 다중 세트의 변환 (즉, 변환 후보), 및 (ii) 연관된 시그널링 방법. 따라서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 단독으로 또는 임의의 조합으로, 본 개시의 기법들 중 임의의 것을 사용하여 MTS 스킴의 표시를 코딩하도록 구성될 수도 있다.
- [0151] MTS 스킴의 표시는 코드워드일 수도 있다. 일부 예들에서, MTS 스킴은 분리가능한 변환 (예를 들어, 수평 변환 및 수직 변환) 과 같은 1차 변환 및 2차 변환 양자 모두를 포함할 수도 있다. 그러한 예들에서, 비디오 인코더 (200) 및 비디오 디코더 (300) 는 2차 변환을 나타내는 제 2 코드워드를 코딩할 수도 있고, 여기서 제 2 코드워드는 가용 2차 변환들의 세트에서 2차 변환을 식별할 수도 있다.
- [0152] VVC 의 MTS 설계는 하기 표 1 에 나타낸 바와 같이 6 개의 분리가능한 변환 후보들을 포함하는 단일 세트의 변환들을 사용한다:
- [0153] 표 1 - 도 7 에 나타낸 바와 같이 VVC 에서 허용된 변환 후보들 및 후보들을 시그널링하는데 사용된 대응하는 코드워드들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	DCT-8	DCT-8	11111

[0154]

[0155] 위의 예시의 6 개의 변환 후보들은 도 7 (우측)에 나타낸 바와 같이 이진 트리 (도 6)를 연접함으로써 생성된 코드워드들을 사용하여 시그널링될 수도 있다. 각각의 코드워드에 대해, 변환 후보는 도 7 (좌측)에 도시된 바와 같이 할당될 수도 있다.

[0156] 대안의 MTS 설계는 다음 기법들의 하나 이상의 조합에 기초하여 정의될 수도 있다. 즉, 비디오 인코더 (200) 및 비디오 디코더 (300)는 단독으로 또는 임의의 조합으로 하기에 후술되는 기법들 중 임의의 것을 수행할 수도 있다.

[0157] 1. MTS 설계는 표 1에 나타낸, VVC에서 현재 후보들의 세트의 일부를 교체하거나 교체하지 않으면서 새로운 변환 후보들을 포함함으로써 확장될 수 있다.

[0158] 2. DCT-2 및 DST-7의 조합들이 부가 변환 후보들로서 포함될 수 있다.

[0159] a. 일 예에서, 현재 VVC의 상단 상에 2 개의 변환 후보들이 부가될 수 있으므로, 표 2에 나타낸 바와 같이 총 8 개의 분리가능한 변환 후보들이 허용된다:

표 2 - 변환 후보들 및 연관된 코드워드들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	DCT-8	DCT-8	111110
7	DCT-2	DST-7	1111110
8	DST-7	DCT-2	1111111

[0161]

[0162] b. 또 다른 예에서, 2 개의 변환 후보들이 "H:DCT-8, V:DCT-8" 조합을 제거함으로써 부가될 수 있으므로, 표 3에 나타낸 바와 같이, 총 7 개의 분리가능한 변환 후보들이 허용된다:

[0163]

표 3 - H: DCT-8, V: DCT-8 조합이 없는 DCT2 및 DST7 의 조합들을 포함하는 예시의 변환 후보들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	DCT-2	DST-7	111110
7	DST-7	DCT-2	111111

[0164]

3. IDT 및 DST-7 의 조합들이 부가 변환 후보들로서 포함될 수 있다.

[0166]

a. 예를 들어, 다음 7 개의 변환 후보들은 표 4 에 나타낸 바와 같이, MTS 에서 사용될 수도 있다.

[0167]

표 4 - H: DCT-8, V: DCT-8 조합이 없는 IDT 및 DST7 의 조합들을 포함하는 예시의 변환 후보들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	IDT	DST-7	111110
7	DST-7	IDT	111111

[0168]

b. 또 다른 예에서, 다음 10 개의 변환 후보들은 DCT-2 및 DST-7 의 조합들 뿐만 아니라 IDT 및 DST-7 의 조합들을 부가함으로써 MTS 에서 사용될 수도 있다.

[0170]

표 5 - 6 번째 후보로서 H: DCT-8, V: DCT-8 조합을 유지하는 것에 의해 IDT 및 DST-7 뿐만 아니라 DCT-2 및 DST-7 의 조합들을 포함하는 예시의 변환 후보들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	DCT-8	DCT-8	111110
7	IDT	DST-7	1111110
8	DST-7	IDT	11111110
9	DCT-2	DST-7	111111110
10	DST-7	DCT-2	111111111

[0171]

[0172]

c. 또 다른 예에서, 다음의 9 개의 변환 스윕 (아이덴티티 후보들을 적용하는 것과 동일함) 이 다음과 같이 위의 리스트로부터 DCT-8 및 DCT-8 조합을 교체함으로써 MTS 에서 사용될 수도 있다:

[0173]

표 6 - H: DCT-8, V: DCT-8 조합을 제거하는 것에 의해 IDT 및 DST-7 뿐만 아니라 DCT-2 및 DST-7 의 조합들을 포함하는 예시의 변환 후보들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	IDT	DST-7	111110
7	DST-7	IDT	1111110
8	DCT-2	DST-7	11111110
9	DST-7	DCT-2	11111111

[0174]

[0175]

4. 후보 및 이들의 연관된 이진화 (즉, 코드워드) 는 상이한 순서화를 가질 수도 있다.

[0176]

a. 순서화는 미리 정의될 수도 있으며 각각의 변환 후보의 통계/빈도에 기초한 고정 설계일 수 있다.

[0177]

b. 예를 들어, 사용되는 각각의 변환 후보의 빈도를 랭킹함으로써 순서화가 행해질 수 있다.

[0178]

c. 예를 들어, 변환 후보들 (예를 들어, 사용된 각각의 후보의 확률에 기초하여 생성된 허프만 코드) 을 시그널링하기 위해 사용된 평균 코드워드 길이를 감소시키도록 설계될 수 있다.

[0179]

d. 예를 들어, 실제 코덱에서, H:DST-7, V:DST-7 및 H:DCT-2, V:DCT-2 조합들이 자주 사용된다. 따라서, 시그널링 오버헤드를 감소시키기 위해, 표 1 에서의 MTS 설계는 표 7 의 다음 예에서와 같이 순서화될 수 있다:

[0180] 표 7 - 표 1에서 변환 후보들을 재순서화하는 예, 여기서 제 1 및 제 3 변환 후보들이 스와핑된다.

후보	수평(H)	수직(V)	코드워드
1	DST-7	DST-7	0
2	DCT-2	DCT-2	10
3	IDT	IDT	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	DCT-8	DCT-8	11111

[0181] [0182] 5. 예측 모드 및/또는 블록 사이즈에 의존하여, 블록을 코딩하기 위해 상이한 MTS 설계가 사용될 수 있으며, 여기서 블록은 변환 유닛 (TU) 또는 코딩 유닛 (CU) 일 수 있다.

[0183] a. 상이한 MTS 설계는 다음을 포함할 수도 있다:

[0184] i. 상이한 변환 후보들의 세트;

[0185] ii. 상이한 시그널링 및 이진화 (즉, 각각의 후보에 대해 사용된 코드워드);

[0186] iii. 위의 i) 및 ii) 양자 모두.

[0187] b. 다중 MTS 설계는 인트라 및/또는 인터-예측 모드에 의존하여 변환 선정을 결정하는데 사용될 수도 있다:

[0188] i. 상이한 타입의 예측 방법들 (예를 들어, 인트라 및 인터-예측) 은 상이한 MTS 설계를 사용할 수도 있다. 예를 들어, 블록들에 대해 인터-예측된 코딩을 위해 표 1에 정의된 MTS 가 사용될 수도 있는 한편, 인트라-예측된 블록에 대해 표 5에 정의된 MTS 가 변환을 결정하는데 사용될 수도 있다.

[0189] ii. 인트라-예측 모드의 상이한 서브세트는 상이한 MTS 설계를 사용할 수도 있다. 모드의 상이한 서브세트는 평면, DC 및 각도 모드의 서브세트의 상호 배타적이고 집합적으로 철저한 선택에 의해 정의될 수 있다. 예를 들어, 평면 (0), DC (1) 및 대각 모드들 (34)에 대해, 3 개의 후보들이 있는 표 8에서의 MTS 설계가 사용될 수 있다. (2)에서 (33) 까지의 각도 모드에 대해, 표 9가 사용될 수도 있다. (35)에서 (66) 까지의 각도 모드 또는 그 나머지에 대해, 표 10이 사용될 수도 있다.

[0190] 표 8 - 3 개의 후보들이 있는 MTS 설계의 예

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	11

[0191]

[0192]

표 9 - 5 개의 후보들이 있는 MTS 설계의 예

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	1111

[0193]

[0194]

표 10 - 5 개의 후보가 있는 MTS 설계의 예, 여기서 표 9 에서의 제 4 및 제 5 후보들은 스와핑된다

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DST-7	DCT-8	1110
5	DCT-8	DST-7	1111

[0195]

[0196]

c. 다중 MTS 설계는 블록-형상 및 블록-사이즈에 의존하여 변환 선정을 결정하는데 사용될 수도 있다:

[0197]

i. 상이한 MTS 설계가 상이한 사이즈 및/또는 형상의 블록들에 대해 사용될 수 있다.

[0198]

ii. 예를 들어, 작은 블록들을 코딩하기 위해, 더 적은 후보들을 갖는 MTS 설계가 사용될 수도 있는 한편, 더 큰 블록들에 대해서는 더 많은 변환 후보들을 갖는 다른 MTS 설계가 사용될 수도 있다. 따라서, 작은 블록들에 대한 변환 시그널링 오버헤드가 감소될 수도 있다.

[0199]

iii. 소형 블록들은 그의 폭 및/또는 높이에 기초하여 정의될 수도 있다. 예를 들어, 폭 또는 높이가 8 미만인 블록들이 소형 블록들로 간주될 수도 있는 한편, 나머지 블록들은 대형 블록들로 간주될 수도 있다 (예를 들어, 블록의 최소 폭 및 높이가 16 보다 작으면, 그 블록은 소형으로 분류될 수도 있다.)

[0200]

iv. 블록들은 또한 정사각형/직사각형 형상을 기준으로 분류될 수 있으며, 여기서 폭과 높이 사이의 비율은 상이한 형상의 블록들을 분류하는데 사용될 수 있다 (예를 들어, 4x8 및 8x4 블록들은 하나의 클래스에 속할 수도 있고, 사이즈 4x16 및 16x4 의 블록은 또 다른 클래스에 속할 수도 있다).

[0201]

d. 단일 (통합) MTS 설계가 또한 시그널링을 위해 사용될 수도 있다.

[0202]

6. 변환 후보들을 시그널링하기 위한 컨텍스트 도출은 다음 중 하나 또는 조합들에 의존하여 이루어질 수도 있다.

[0203]

a. 블록 사이즈;

[0204]

b. 블록 형상;

[0205]

c. 인트라-모드;

[0206]

d. 인터-모드.

[0207]

- 인트라-예측된 및 인터-예측된 CU들/TU들에 대해 별도의 컨텍스트들이 정의될 수도 있다.

[0208]

- 블록의 최소 폭 및 높이에 기초하여 별도의 컨텍스트가 정의될 수도 있다.

[0209]

7. 분리가능한 변환들에 부가하여, MTS 설계는 변환 후보들로서 분리불가능한 변환들을 또한 포함할 수도 있다.

예는 표 11에 예시된다.

표 11 - MTS 설계에서 분리가능한 변환들에 부가하여 분리불가능한 변환들을 포함하는 예시의 변환 후보들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	분리불가능한 변환 1		111110
7	분리불가능한 변환 2		111111

[0211]

또한, 분리가능한 변환들에 부가하여 2차 변환들이 MTS 설계에 포함될 수 있다. 표 12는 MTS의 예를 제시하며, 여기서 H: DCT-8, V: DCT-8 조합은 2차 변환들의 세트로 구체된다.

[0213] 2차 변환들은 2016년 9월 20일 출원된 미국 출원 제 15/270,455호 및 2019년 3월 25일에 출원된 제 16/364,007호에 기재된 양태들을 포함할 수도 있다. 구체적으로, 인코더 측에서 2차 변환이 1차 변환 계수들의 서브세트(예를 들어, 2-D DCT-2로부터 획득됨)에 적용될 수 있으며, 여기서 순서는 디코더에서 역전된다(먼저 역 2차 변환이 적용된 후 1차 변환이 적용된다).

[0214] 2차 변환들은 도 8에 도시되고 하기에서 더 상세히 논의되는 바와 같이 다중 2차 변환들 중에서 선택된 변환을 결정하기 위해 부가적인 시그널링을 필요로 할 수도 있다. 단일 2차 변환 후보만 있는 경우(즉, 세트가 단지 단일 2차 변환일 수도 있음), 표 12에서 MTS 시그널링의 상단에 대해서는 부가적인 시그널링이 필요하지 않다.

[0215] 2차 변환들에 대해, 변환 후보들은 또한 예측 모드, 블록 사이즈 및 블록 형상 중 하나 또는 조합들에 의존할 수도 있다.

표 12: MTS 설계에서 분리가능한 변환들에 부가하여 2차 변환들을 포함하는 변환 후보들

후보	수평(H)	수직(V)	코드워드
1	IDT	IDT	0
2	DCT-2	DCT-2	10
3	DST-7	DST-7	110
4	DCT-8	DST-7	1110
5	DST-7	DCT-8	11110
6	2차 변환들		11111

[0217]

[0218] 8. MTS 설계에서 분리가능한 변환들은 IDT, DST-7, DCT-8 및 DCT-2에 부가하여 다른 타입의 DST들 및 DCT들(예를 들어, DST-4 및 DCT-4)의 조합들을 사용하여 구축될 수도 있다.

[0219] 9. 위의 방법들 중 하나 또는 조합들이 인트라-예측된 CU들에 대해 사용될 수 있다.

[0220] 10. 위의 방법들 중 하나 또는 조합들이 인터-예측된 CU들에 대해 사용될 수 있다.

[0221] 11. 위의 방법들 중 하나 또는 조합들이 인트라 및 인터-예측된 CU들 양자 모두에 대해 사용될 수 있다.

[0222] 12. 위의 방법들 중 하나 또는 조합들이 루마 또는 크로마 채널들 또는 양자 모두에 대해 사용될 수 있다.

[0223] 도 8 은 2차 변환들을 지원하는 일 예의 MTS 설계를 도시하는 개념적 다이어그램이다. 2차 변환이 시그널링 /선정되는 경우, N 개의 가능한 2차 변환들 중에서 2차 변환을 표시하기 위해 부가적인 시그널링이 사용될 수도 있다. 즉, 비디오 인코더 (200) 는 1차 변환, 및 이 1차 변환에 부가하여 2차 변환이 적용될 것임을 표시하는 제 1 코드워드를 인코딩하고, 추가로 변환들의 세트의 2차 변환 (예를 들어, 도 8 에 도시된 N 개의 사용 변환들 중 하나) 을 표시하는 제 2 코드워드를 인코딩할 수도 있다. 유사하게, 비디오 디코더 (300) 는 제 1 코드워드를 디코딩하고 제 1 코드워드가 1차 변환을 표시하고 2차 변환이 적용될 것이라고 결정할 수도 있다.

따라서, 비디오 디코더 (300) 는 추가로 제 1 코드워드에 응답하여 제 2 코드워드를 디코딩하고, 제 2 코드워드를 사용하여 2차 변환을 결정할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 1차 및 2차 변환들 양자 모두를 추가로 적용할 수도 있다.

[0224] 일부 예들에서, 하기에서 더 상세히 논의되는 바와 같이, 2차 변환은 저-주파수 분리불가능한 변환 (Low-Frequency Non-separable Transformation; LFNST) 일 수도 있다. 따라서, 제 1 코드워드는 MST 신택스 엘리먼트로 지칭될 수도 있고, 제 2 코드워드는 LFNST 신택스 엘리먼트로 지칭될 수도 있다.

[0225] 도 9 및 도 10 은 저-주파수 분리불가능한 변환 (LFNST) 의 사용을 도시하는 개념적 다이어그램들이다. LFNST 는 JEM-7.0 에서 사용되어 MTS 의 코딩 효율성을 더욱 개선하며, 여기서 LFNST 의 구현은 미국 특허 공개 제 2017/0238013 호, 미국 특허 공개 제 2017/0094313, 2017/0238014 호, 미국 특허 출원 제 16/364,007 호, 및 미국 특허 출원 제 62/668,105 및 62/849,689 호 (대안의 설계 및 추가 상세를 기재함) 에 기재되는, HyGT (Hypercube-Givens Transform) 에 기초한다. LFNST 는 이전에 분리불가능한 2차 변환 (NSST) 또는 2차 변환으로 불렸지만, LFNST, NSST 및 2차 변환은 일반적으로 동일한 기법들을 지칭할 수도 있다. 최근, LFNST 는 Koo 등이 "CE6: Reduced Secondary Transform (RST) (CE6-3.1)," ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 JVET (Joint Video Experts Team), 14th Meeting, Geneva, CH, 19-27 Mar. 2019, document JVET-N0193 에 기재된 바와 같이, 초안 VVC 표준에 채택되었다.

[0226] 도 9 는 비디오 인코더 (200) 및 비디오 디코더 (300) 에 의해 적용된 LFNST 변환들을 도시하는 개념적 다이어그램이다. LFNST 는 코덱에서 분리가능한 변환과 양자화 사이의 새로운 단계를 도입한다. 도 10 은 HxW 블록의 (상단-좌측 부분에서) 계수들의 서브세트에 적용된 LFNST 를 도시하는 개념적 다이어그램이다.

[0227] VVC Draft 5 는 상당한 코딩 이익 없이 일부 인코더/디코더 복잡성을 도입하는 다음의 사양들을 포함한다.

[0228] 1) LFNST 는 변환 스킵 (TS) 모드를 제외한 임의의 MTS 변환으로 사용될 수 있다.

[0229] 2) LFNST 인덱스를 시그널링하기 위해 사용된 컨텍스트 모델은 MTS 인덱스에 의존한다. 예를 들어, 비디오 코더는 MTS 인덱스에 기초하여 LFNST 인덱스를 CABAC 코딩 (또는 다른 코딩 기법) 에서 사용된 컨텍스트 모델을 선택할 수도 있다. 컨텍스트 모델은 특정 값을 갖는 LFNST 인덱스의 제 1 비트의 확률을 표시할 수도 있다.

[0230] 3) LFNST 는 크로마 채널들을 코딩하는데 사용될 수 있지만, MTS 는 크로마에 대해 규범적으로 디스에이블되고,

[0231] 4) 4x4 및 8x8 블록에 적용된 LFNST 는 단일 단계를 사용 (즉, 단일 분리불가능한 변환을 사용) 하여 구현될 수 있지만, 아직 현재 구현은 2 단계 프로세스들에 기초한다.

[0232] 본 개시는 위의 쟁점들을 해결함으로써 LFNST 설계를 단순화할 수도 있는 기법들을 설명한다. 본 개시에서 설명된 LFNST 설계는 개별적으로 또는 임의의 조합으로 사용될 수도 있다.

[0233] VVC Draft 5 에서, LFNST 는 LFNST 인덱스 값들 0, 1 및 2 를 사용하여 시그널링되는, 3 가지 모드들을 포함하며, 여기서:

[0234] • LFNST 인덱스 0 은 LFNST 프로세스를 스kip하는 것에 대응한다 (예를 들어, MTS 만 사용됨).

[0235] • LFNST 인덱스 1 및 2 는 블록 (예를 들어, CU, TU 등) 의 사이즈 및 모드에 의존하여 선정된 2 개의 변환들의 세트로부터 분리불가능한 변환을 결정하는데 사용된다.

[0236] 이러한 설계에 기초하여, LFNST 는 소정의 조건들 하에서 사용되도록 제한될 수 있다:

[0237] • LFNST 는 미리정의된 변환들의 세트 (즉, 소정의 MTS 후보들) 와 함께 적용될 수도 있다. 따라서, 미리 정의된 세트로부터 변환이 선택되는 경우 LFNST 인덱스가 시그널링될 수도 있고, 이 세트는 블록 치수 (폭 및 높이) 에 의존할 수도 있다. 그렇지 않으면, 즉 미리정의된 세트로부터의 변환이 선택되면, LFNST 인덱스는

LFNST 가 스kip되도록 (즉, 적용되지 않도록) 0 으로 추론될 수도 있다.

[0238] ◦ LFNST 의 사용은 변환 타입 및/또는 MTS 인덱스/플래그 및/또는 블록 치수들에 기초하여 제한될 수 있다.

[0239] ■ LFNST 는 미리정의된 변환 타입들 및/또는 MTS 인덱스들/플래그들이 사용될 때 인에이블될 수 있다.

[0240] • LFNST 는 분리가능한 2-D DCT-2 가 사용되는 경우 (즉, DCT-2 가 수평 및 수직으로 적용되는 경우) 인에이블될 수 있다.

[0241] ◦ VVC 에서, 이것은 MTS 인덱스/플래그가 0 인 경우 (즉, 2-D DCT-2 가 사용됨) LFNST 인덱스를 시그널링하는 것에 대응하고, MTS 인덱스 플래그가 0 과 상이한 경우 LFNST 인덱스/플래그는 시그널링되지 않고 비디오 디코더 (300) 에 의해 0 으로 추론된다.

[0242] ◦ 이 경우, LFNST 인덱스/플래그를 코딩하기 위한 컨텍스트 모델은 MTS 인덱스에 의존하지 않는다.

[0243] ■ LFNST 는 변환 스kip 모드에 대해 인에이블될 수 있다.

[0244] • 변환 스kip이 인에이블될 때, LFNST 프로세스가 스kip되고 LFNST 인덱스/플래그는 0 으로 추론된다.

[0245] ◦ LFNST 인덱스의 코딩 시그널링을 위한 컨텍스트 모델은 MTS 인덱스에 의존할 수도 있다. 각각의 MTS 인덱스에 대해, LFNST 인덱스들을 코딩하기 위해 별도의 컨텍스트들이 정의될 수 있다.

[0246] • LFNST 는 루마 블록들에 사용될 수 있고 크로마 채널에 대해 디스에이블될 수 있다. 따라서, LFNST 인덱스는 시그널링되지 않고 크로마 채널에 대해 0 으로 추론된다.

[0247] 이로써, 본 개시의 하나 이상의 기법들에 따른 예에서, 비디오 인코더 (200) 는 LFNST 의 시그널링에 대한 하나 이상의 제한들이 현재 블록에 적용되지 않는 경우 비디오 데이터의 현재 블록에 대한 LFNST 인덱스를, 비디오 데이터의 인코딩된 표현을 포함하는 비트스트림에 부가할 수도 있다. 부가적으로, 이 예에서, 비디오 인코더 (200) 는 현재 블록에 대한 중간 데이터를 생성하기 위해 현재 블록에 대한 잔차 데이터에 변환을 적용할 수도 있다. 이 예에서, LFNST 인덱스의 값에 기초하여, 비디오 인코더 (200) 는 현재 블록에 대한 변환 계수들을 생성하기 위해 중간 데이터에 LFNST 를 적용할 수도 있다. 비디오 인코더 (200) 는 비트스트림에서 현재 블록에 대한 변환 계수들을 나타내는 데이터를 포함할 수도 있다.

[0248] 본 개시의 하나 이상의 기법들에 따른 일 예에서, 비디오 디코더 (300) 는 LFNST 의 시그널링에 대한 하나 이상의 제한들이 현재 블록에 적용되지 않는 경우 비디오 데이터의 현재 블록에 대한 LFNST 인덱스를, 비디오 데이터의 인코딩된 표현을 포함하는 비트스트림으로부터 획득할 수도 있다. 이 예에서, 비디오 디코더 (300) 는 비트스트림에서의 데이터에 기초하여, 변환 계수들의 블록을 결정할 수도 있다. LFNST 인덱스의 값에 기초하여, 비디오 디코더 (300) 는 현재 블록에 대한 중간 데이터를 생성하기 위해 변환 계수들의 블록에 역 LFNST 를 적용할 수도 있다. 비디오 디코더 (300) 는 현재 블록에 대한 잔차 블록을 생성하기 위해 현재 블록에 대한 중간 데이터에 변환의 역을 적용할 수도 있다. 이 예에서, 비디오 디코더 (300) 는 현재 블록에 대한 잔차 데이터에 기초하여 현재 블록의 샘플들을 복원할 수도 있다.

[0249] 도 11a 및 도 11b 는 2019 년 5 월 30 일 VVC 테스트 모델 (VTM) 마다 예시의 2-단계 LFNST 프로세스 구현을 도시하는 개념적 다이어그램들이다. 이 예에서, LFNST 는 상단-좌측 영역에서 더 어둡게 음영처리된 서브블록 내의 분리가능한 변환 계수들 (예를 들어, MTS 계수들) 의 서브세트의 상단에 대해 적용된다. 이러한 2-단계 절차는 도 11a 의 블록 형상들/사이즈들에 대해 회피불가능할 수도 있다. 그러나, 4x4 및 8x8 블록에 대해, 도 11b 에 나타낸 바와 같이, LFNST 및 분리가능한 변환 사이즈가 정렬된다 (즉, LFNST 및 분리가능한 변환들의 지원은 더 어둡게 음영처리된 블록 내에서 동일한 계수 위치들/포지션들을 포함할 수도 있다). 이 경우, 이 변환 프로세스는 다음과 같이 단일-단계의 분리불가능한 변환들로 감소될 수 있다.

[0250] - 2 개의 단계들에 LFNST 를 적용하는 대신 (예를 들어, 분리가능한 변환으로 LFNST 를 적용하는 대신), 비디오 인코더 (200) 및 비디오 디코더 (300) 는 하나의 단계에서 분리불가능한 변환으로부터 직접 계수들을 획득할 수도 있다. 예를 들어:

[0251] o 4×4 경우에 대해, 행렬-벡터 꼽으로 구현될 수 있는, 16-길이 분리불가능한 변환이 사용된다.

[0252] o 8×8 경우에 대해, 행렬-벡터 곱으로 또한 구현될 수 있는, 64-길이 분리불가능한 변환이 사용된다.

[0253] - 더욱이, 제로-아웃 스킴 (예를 들어, 미국 가특허출원 제 62/849,689 호에 기재됨)은 행렬-기반 구현들에 필요한 곱셈의 수를 감소시키는데 사용될 수 있다.

[0254] o 제로-아웃 스킴에서, 첫 번째 K 최저-주파수 계수들이 계산될 필요가 있을 수도 있고, 변환 계수들의 나머지는 규범적으로 제로-아웃될 수도 있다 (즉, 비디오 인코더 (200) 및 비디오 디코더 (300) 양자 모두에서 0 으로 가정됨).

[0255] ■ K 의 값은 블록 사이즈에 의존할 수도 있다. 예를 들어:

[0256] • 4×4 블록들에 대해, K 는 8 일 수 있으므로, 나머지 8 개의 계수들은 규범적으로 제로 아웃된다.

[0257] • 8×8 블록들에 대해, K 는 8 일 수 있으므로, 나머지 56 개의 계수들은 규범적으로 제로 아웃된다.

[0258] • 8×8 블록들에 대해, K 는 16 일 수 있으므로, 나머지 48 개의 계수들은 규범적으로 제로 아웃된다.

[0259] - LFNST 는 4×4 및 8×8 에 대한 단일 단계 분리불가능한 변환으로서 구현될 수 있으며, 다른 경우들에 대해 LFNST 는 미국 가특허 제 62/337,736 호에 기재된 바와 같이 2-단계 프로세스로 구현될 수도 있다.

[0260] 본 개시의 기법들에 따른 예에서, 비디오 인코더 (200)는 비디오 데이터의 제 1 블록에 대한 잔차 데이터를 결정할 수도 있다. 부가적으로, 비디오 인코더 (200)는 비디오 데이터의 제 2 블록에 대한 잔차 데이터를 결정할 수도 있다. 제 1 블록의 폭이 제 1 블록의 높이와 동일한 것에 기초하여: 비디오 인코더 (200)는 제 1 블록에 대한 변환 계수들을 생성하기 위해 제 1 블록에 대한 잔차 데이터에 분리불가능한 변환을 적용하고; 비디오 데이터의 인코딩된 표현을 포함하는 비트스트림에, 제 1 블록에 대한 변환 계수들을 나타내는 데이터를 포함할 수도 있다. 이 예에서, 제 2 블록의 폭이 제 2 블록의 높이와 동일하지 않은 것에 기초하여, 비디오 인코더 (200)는 제 2 블록에 대한 중간 데이터를 생성하기 위해 제 2 블록에 대한 잔차 데이터에 변환을 적용하고; 제 2 블록에 대한 변환 계수를 생성하기 위해 제 2 블록에 대한 중간 데이터에 LFNST 를 적용하며; 그리고 제 2 블록에 대한 변환 계수들을 나타내는 데이터를 비트스트림에 포함할 수도 있다.

[0261] 본 개시의 기법들에 따른 다른 예에서, 비디오 디코더 (300)는 비디오 데이터의 인코딩된 표현을 포함하는 비트스트림에서의 제 1 데이터에 기초하여, 비디오의 제 1 블록에 대한 변환 계수들을 결정할 수도 있다. 부가적으로, 비디오 디코더 (300)는 비트스트림에서의 제 2 데이터에 기초하여, 비디오 데이터의 제 2 블록에 대한 변환 계수들을 결정할 수도 있다. 제 1 블록의 폭이 제 1 블록의 높이와 동일한 것에 기초하여, 비디오 디코더 (300)는 제 1 블록에 대한 잔차 데이터를 생성하기 위해 제 1 블록에 대한 변환 계수들에 분리불가능한 변환의 역을 적용하고; 그리고 제 1 블록에 대한 잔차 데이터에 기초하여 제 1 블록의 샘플들을 복원할 수도 있다. 이 예에서, 제 2 블록의 폭이 제 2 블록의 높이와 동일하지 않은 것에 기초하여, 비디오 디코더 (300)는 제 2 블록에 대한 중간 데이터를 생성하기 위해 제 2 블록에 대한 변환 계수들에 역 변환을 적용하고; 제 2 블록에 대한 잔차 데이터를 생성하기 위해 제 2 블록에 대한 중간 데이터에 LFNST 의 역을 적용하며; 그리고 제 2 블록에 대한 잔차 데이터에 기초하여 제 2 블록의 샘플들을 복원할 수도 있다.

[0262] 도 12 는 본 개시의 기법들을 수행할 수도 있는 예시적인 비디오 인코더 (200)를 도시하는 블록 다이어그램이다. 도 12 는 설명의 목적으로 제공되며 본 개시에 폭넓게 예시되고 기재되는 바와 같이 기법들을 제한하는 것으로 고려되지 않아야 한다. 설명의 목적으로, 본 개시는 HEVC 비디오 코딩 표준 및 개발 중인 H.266/VVC 비디오 코딩 표준과 같은 비디오 코딩 표준들의 콘텍스트에서 비디오 인코더 (200)를 설명한다. 그러나, 본 개시의 기법들은 이를 비디오 코딩 표준들에 제한되지 않으며, 일반적으로 비디오 인코딩 및 디코딩에 적용 가능하다.

[0263] 도 12 의 예에서, 비디오 인코더 (200)는 비디오 데이터 메모리 (230), 모드 선택 유닛 (202), 잔차 생성 유닛 (204), 변환 프로세싱 유닛 (206), 양자화 유닛 (208), 역 양자화 유닛 (210), 역 변환 프로세싱 유닛 (212), 복원 유닛 (214), 필터 유닛 (216), 디코딩된 픽처 버퍼 (DPB)(218), 및 엔트로피 인코딩 유닛 (220)을 포함한다. 도 12 는 상기 도 4 에 나타낸 바와 같이, 변환 프로세싱 유닛 (206) 및 역 변환 프로세싱 유닛 (212)이 본 개시의 기법들에 따른 변환들을 선택하는 변환 뱅크를 더 포함할 수도 있다. 마찬가지로, 도 4 에 나타낸 바와 같이, 변환 프로세싱 유닛 (206)은 엔트로피 인코딩 유닛 (220)에 선택된 변환의 표시를 제공할 수도 있고, 이는 MTS 스킴에 대한 다양한 변환들 중 어느 것이 비디오 데이터의 현재 블록에 대해 선택되는지를

나타내는 본 개시의 기법들에 따라 데이터를 인코딩할 수 있다.

[0264] 비디오 데이터 메모리 (230) 는 비디오 인코더 (200) 의 컴포넌트들에 의해 인코딩될 비디오 데이터를 저장할 수도 있다. 비디오 인코더 (200) 는 예를 들어, 비디오 소스 (104)(도 1)로부터 비디오 데이터 메모리 (230) 에 저장된 비디오 데이터를 수신할 수도 있다. DPB (218) 는 비디오 인코더 (200) 에 의한 후속 비디오 데이터의 예측에 사용하기 위해 참조 비디오 데이터를 저장하는 참조 픽처 메모리로서 작용할 수도 있다. 비디오 데이터 메모리 (230) 및 DPB (218) 는 다양한 메모리 디바이스들, 예컨대 동기식 랜덤 액세스 메모리 (synchronous dynamic random access memory; SDRAM), 자기저항 RAM (magnetoresistive RAM; MRAM), 저항 RAM (resistive RAM; RRAM) 을 포함하는 DRAM, 또는 다른 타입들의 메모리 디바이스들 중 임의의 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (230) 및 DPB (218) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (230) 는 도시된 바와 같이, 비디오 인코더 (200) 의 다른 컴포넌트들과 온-칩이거나 그 컴포넌트들에 대하여 오프-칩일 수도 있다.

[0265] 본 개시에서, 비디오 데이터 메모리 (230) 에 대한 참조는 이처럼 구체적으로 기재되지 않으면 비디오 인코더 (200) 내부의 메모리 또는 이처럼 구체적으로 기재되지 않으면 비디오 인코더 (200) 외부의 메모리로 제한되는 것으로 해석되지 않아야 한다. 오히려, 비디오 데이터 메모리 (230) 에 대한 참조는 비디오 인코더 (200) 가 인코딩을 위해 수신하는 비디오 데이터 (예를 들어, 인코딩될 현재 블록에 대한 비디오 데이터) 를 저장하는 참조 메모리로서 이해되어야 한다. 도 1 의 메모리 (106) 는 또한 비디오 인코더 (200) 의 다양한 유닛들으로부터의 출력들의 일시적 저장을 제공할 수도 있다.

[0266] 도 12 의 다양한 유닛들은 비디오 인코더 (200) 에 의해 수행되는 동작들의 이해를 돋기 위해 도시된다. 이 유닛들은 고정 기능 회로들, 프로그램가능 회로들, 또는 이들의 조합으로서 구현될 수도 있다. 고정 기능 회로들은 특정 기능성을 제공하는 회로들을 지칭하며, 수행될 수 있는 동작들에 대해 미리설정된다. 프로그램가능 회로들은 다양한 태스크들을 수행하도록 프로그램될 수 있는 회로들을 지칭하며, 수행될 동작들에서 유연한 기능성을 제공한다. 예를 들어, 프로그램가능 회로들은 프로그램가능 회로들이 소프트웨어 또는 펌웨어의 명령들에 의해 정의된 방식으로 동작하게 하는 소프트웨어 또는 펌웨어를 실행할 수도 있다. 고정 기능 회로들은 소프트웨어 명령들을 (예를 들어, 파라미터들을 수신하거나 파라미터들을 출력하기 위해) 실행할 수도 있지만, 고정 기능 회로들이 수행하는 동작 타입들은 일반적으로 불변이다. 일부 예들에서, 유닛들의 하나 이상은 별개의 회로 블록들 (고정 기능 또는 프로그램가능) 일 수도 있고, 일부 예들에서, 하나 이상의 유닛들은 집적 회로들일 수도 있다.

[0267] 비디오 인코더 (200) 는 프로그램가능 회로들로부터 형성된, 산술 로직 유닛 (arithmetic logic unit; ALU) 들, 기본 기능 유닛 (elementary function unit; EFU) 들, 디지털 회로들, 아날로그 회로들, 및/또는 프로그램 가능 코어들을 포함할 수도 있다. 비디오 인코더 (200) 의 동작들이 프로그램가능 회로들에 의해 실행되는 소프트웨어를 사용하여 수행되는 예들에서, 메모리 (106)(도 1) 는 비디오 인코더 (200) 가 수신하고 실행하는 소프트웨어의 오브젝트 코드를 저장할 수도 있거나 또는 비디오 인코더 (200) 내의 다른 메모리 (미도시) 가 이러한 명령들을 저장할 수도 있다.

[0268] 비디오 데이터 메모리 (230) 는 수신된 비디오 데이터를 저장하도록 구성된다. 비디오 인코더 (200) 는 비디오 데이터 메모리 (230)로부터 비디오 데이터의 픽처를 취출하고 그 비디오 데이터를 잔차 생성 유닛 (204) 및 모드 선택 유닛 (202) 에 제공할 수도 있다. 비디오 데이터 메모리 (230) 에서의 비디오 데이터는 인코딩될 원시 비디오 데이터일 수도 있다.

[0269] 모드 선택 유닛 (202) 은 모션 추정 유닛 (222), 모션 보상 유닛 (224), 및 인트라-예측 유닛 (226) 을 포함한다. 모드 선택 유닛 (202) 은 다른 예측 모드들에 따라 비디오 예측을 수행하기 위해 부가적인 기능 유닛들을 포함할 수도 있다. 예를 들어, 모드 선택 유닛 (202) 은 팔레트 유닛, 인트라-블록 카피 유닛 (모션 추정 유닛 (222) 및/또는 모션 보상 유닛 (224) 의 일부일 수도 있음), 아핀 유닛, 선형 모델 (LM) 유닛 등을 포함할 수도 있다.

[0270] 모드 선택 유닛 (202) 은 일반적으로 인코딩 파라미터들의 조합들 및 그러한 조합들에 대한 결과의 레이트-왜곡 값들을 테스트하기 위해 다중 인코딩 패스들을 조정한다. 인코딩 파라미터들은 CU들로의 CTU들의 파티셔닝, CU들에 대한 예측 모드들, CU들의 잔차 데이터에 대한 변환 타입들, CU들의 잔차 데이터에 대한 양자화 파라미터들 등을 포함할 수도 있다. 모드 선택 유닛 (202) 은 궁극적으로 다른 테스트된 조합들보다 우수한 레이트-왜곡 값들을 갖는 인코딩 파라미터들의 조합을 선택할 수도 있다.

- [0271] 비디오 인코더 (200) 는 비디오 데이터 메모리 (230) 로부터 취출된 픽처를 일련의 CTU들로 파티셔닝하고, 슬라이스 내에 하나 이상의 CTU들을 캡슐화할 수도 있다. 모드 선택 유닛 (202) 은 상술한 HEVC 의 퀘드-트리 구조 또는 QTBT 구조와 같은, 트리 구조에 따라 픽처의 CTU 를 파티셔닝할 수도 있다. 상술한 바와 같이, 비디오 인코더 (200) 는 트리 구조에 따라 CTU 를 파티셔닝하는 것으로부터 하나 이상의 CU들을 형성할 수도 있다. 이러한 CU 는 또한 일반적으로 "비디오 블록" 또는 "블록" 으로 지칭될 수도 있다.
- [0272] 일반적으로, 모드 선택 유닛 (202) 은 또한 현재 블록 (예를 들어, 현재 CU, 또는 HEVC 에서, PU 및 TU 의 오버랩 부분) 에 대한 예측 블록을 생성하기 위해 그의 컴포넌트들 (예를 들어, 모션 추정 유닛 (222), 모션 보상 유닛 (224) 및 인트라-예측 유닛 (226)) 을 제어한다. 현재 블록의 인터-예측을 위해, 모션 추정 유닛 (222) 은 모션 탐색을 수행하여 하나 이상의 참조 픽처들 (예를 들어, DPB (218) 에 저장된 하나 이상의 이전에 코딩된 픽처들) 에서 하나 이상의 근접하게 매칭하는 참조 블록들을 식별할 수도 있다. 특히, 모션 추정 유닛 (222) 은, 예를 들어 절대차의 합 (SAD), 제곱차의 합 (SSD), 평균 절대차 (MAD), 평균 제곱차 (MSD) 등에 따라, 잠재적 참조 블록이 현재 블록에 얼마나 유사한지를 나타내는 값을 계산할 수도 있다. 모션 추정 유닛 (222) 은 일반적으로 고려되는 참조 블록과 현재 블록 사이의 샘플 별 차이들을 사용하여 이러한 계산들을 수행할 수도 있다. 모션 추정 유닛 (222) 은 현재 블록과 가장 근접하게 매칭하는 참조 블록을 표시하는, 이러한 계산들로부터 야기되는 최저 값을 갖는 참조 블록을 식별할 수도 있다.
- [0273] 모션 추정 유닛 (222) 은 현재 픽처에서의 현재 블록의 포지션에 대한 참조 픽처들에서의 참조 블록들의 포지션들을 정의하는 하나 이상의 모션 벡터 (MV) 들을 형성할 수도 있다. 모션 추정 유닛 (222) 은 그 후 모션 벡터들을 모션 보상 유닛 (224) 에 제공할 수도 있다. 예를 들어, 단방향 인터-예측에 대해, 모션 추정 유닛 (222) 은 단일 모션 벡터를 제공할 수도 있는 반면, 양방향 인터-예측에 대해, 모션 추정 유닛 (222) 은 2 개의 모션 벡터들을 제공할 수도 있다. 그 후, 모션 보상 유닛 (224) 은 모션 벡터들을 사용하여 예측 블록을 생성할 수도 있다. 예를 들어, 모션 보상 유닛 (224) 은 모션 벡터를 사용하여 참조 블록의 데이터를 취출할 수도 있다. 다른 예로서, 모션 벡터가 분수 샘플 정밀도를 갖는다면, 모션 보상 유닛 (224) 은 하나 이상의 보간 필터들에 따라 예측 블록에 대한 값을 보간할 수도 있다. 또한, 양방향 인터-예측에 대해, 모션 보상 유닛 (224) 은 개개의 모션 벡터에 의해 식별된 2 개의 참조 블록들에 대한 데이터를 취출하고, 예를 들어 샘플 별 평균화 또는 가중된 평균화를 통해 취출된 데이터를 결합할 수도 있다.
- [0274] 또 다른 예로서, 인트라-예측 또는 인트라-예측 코딩에 대해, 인트라-예측 유닛 (226) 은 현재 블록에 이웃하는 샘플들로부터 예측 블록을 생성할 수도 있다. 예를 들어, 방향성 모드들에 대해, 인트라-예측 유닛 (226) 은 일반적으로 이웃하는 샘플들의 값을 수학적으로 결합하고 현재 블록에 걸쳐 정의된 방향에서 이들 계산된 값을 파풀레이트하여 예측 블록을 생성할 수도 있다. 또 다른 예로서, DC 모드에 대해, 인트라-예측 유닛 (226) 은 현재 블록에 대한 이웃하는 샘플들의 평균을 계산하고 예측 블록을 생성하여 예측 블록의 각각의 샘플에 대해 이러한 결과의 평균을 포함할 수도 있다.
- [0275] 모드 선택 유닛 (202) 은 예측 블록을 잔차 생성 유닛 (204) 에 제공한다. 잔차 생성 유닛 (204) 은 비디오 데이터 메모리 (230) 로부터의 현재 블록의 원시의, 코딩되지 않은 버전 및 모드 선택 유닛 (202) 으로부터의 예측 블록을 수신한다. 잔차 생성 유닛 (204) 은 현재 블록과 예측 블록 사이의 샘플 별 차이를 계산한다. 결과의 샘플 별 차이는 현재 블록에 대한 잔차 블록을 정의한다. 일부 예들에서, 잔차 생성 유닛 (204) 은 또한 잔차 차분 펄스 코드 변조 (residual differential pulse code modulation; RDPCM) 를 사용하여 잔차 블록을 생성하기 위해 잔차 블록에서의 샘플 값을 사이의 차이를 결정할 수도 있다. 일부 예들에서, 잔차 생성 유닛 (204) 은 이진 감산을 수행하는 하나 이상의 감산 회로들을 사용하여 형성될 수도 있다.
- [0276] 모드 선택 유닛 (202) 이 CU들을 PU들로 파티셔닝하는 예들에서, 각각의 PU 는 루마 예측 유닛 및 대응하는 크로마 예측 유닛들과 연관될 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 다양한 사이즈를 갖는 PU들을 지원할 수도 있다. 위에 표시된 바와 같이, CU 의 사이즈는 CU 의 루마 코딩 블록의 사이즈를 지칭할 수도 있고 PU 의 사이즈는 PU 의 루마 예측 유닛의 사이즈를 지칭할 수도 있다. 특정 CU 의 사이즈가 $2Nx2N$ 이라고 가정하면, 비디오 인코더 (20) 는 인트라-예측을 위해 $2Nx2N$ 또는 NxN 의 PU 사이즈들을 지원하고, 인터-예측을 위해 $2Nx2N$, $2NxN$, $Nx2N$, NxN , 기타 등등의 대칭적인 PU 사이즈들을 지원할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 또한, 인터-예측을 위해 $2NxnU$, $2NxnD$, $nLx2N$, 및 $nRx2N$ 의 PU 사이즈에 대한 비대칭적 파티셔닝을 지원할 수도 있다.
- [0277] 모드 선택 유닛이 CU 를 PU들로 추가로 파티셔닝하지 않는 예들에서, 각각의 CU 는 루마 코딩 블록 및 대응 크로마 코딩 블록들과 연관될 수도 있다. 위에서와 같이, CU 의 사이즈는 CU 의 루마 코딩 블록의 사이즈를

지칭할 수도 있다. 비디오 인코더 (200) 및 비디오 디코더 (300) 는 $2N \times 2N$, $2N \times N$ 또는 $N \times 2N$ 의 CU 사이즈를 지원할 수도 있다.

[0278] 인트라 블록 카피 모드 코딩, 아핀 모드 코딩 및 선형 모델 (LM) 모드 코딩과 같은 다른 비디오 코딩 기법들에 대해, 몇몇 예들에서와 같이, 모드 선택 유닛 (202) 은 코딩 기술과 연관된 개개의 유닛들을 통해, 인코딩될 현재 블록에 대한 예측 블록을 생성한다. 팔레트 모드 코딩과 같은 일부 예에서, 모드 선택 유닛 (202) 은 예측 블록을 생성하지 않을 수도 있고, 대신에 선택된 팔레트에 기초하여 블록을 복원하는 방식을 표시하는 신택스 엘리먼트들을 생성할 수도 있다. 이러한 모드들에서, 모드 선택 유닛 (202) 은 이들 신택스 엘리먼트들을 인코딩될 엔트로피 인코딩 유닛 (220) 에 제공할 수도 있다.

[0279] 상술한 바와 같이, 잔차 생성 유닛 (204) 은 현재 블록 및 대응하는 예측 블록에 대해 비디오 데이터를 수신한다. 잔차 생성 유닛 (204) 은 그 후 현재 블록에 대한 잔차 블록을 생성한다. 잔차 블록을 생성하기 위해, 잔차 생성 유닛 (204) 은 현재 블록과 예측 블록 사이의 샘플 별 차이들을 계산한다.

[0280] 변환 프로세싱 유닛 (206) 은 잔차 블록에 하나 이상의 변환들을 적용하여 변환 계수들의 블록 (본 명세서에서 "변환 계수 블록" 으로 지칭됨) 을 생성한다. 변환 프로세싱 유닛 (206) 은 다양한 변환들을 잔차 블록에 적용하여 변환 계수 블록을 형성할 수도 있다. 예를 들어, 변환 프로세싱 유닛 (206) 은 이산 코사인 변환 (DCT), 방향성 변환, Karhunen-Loeve 변환 (KLT) 또는 개념적으로 유사한 변환을 잔차 블록에 적용할 수도 있다. 일부 예들에서, 변환 프로세싱 유닛 (206) 은 잔차 블록에 대한 다중 변환들, 예를 들어 1차 변환 및 2차 변환, 예컨대 회전 변환을 수행할 수도 있다. 일부 예들에서, 변환 프로세싱 유닛 (206) 은 잔차 블록에 변환들을 적용하지 않는다.

[0281] 본 개시의 기법들에 따라, 변환 프로세싱 유닛 (206) 은 1차 변환 및 2차 변환 양자 모두를 포함하는 변환 스킵 (예를 들어, MTS 스킵) 을 선택할 수도 있다. 1차 변환은 다양한 DCT들 및/또는 DST들 중 하나와 같은 수평 변환 및 수직 변환을 포함하는 분리가능한 변환일 수도 있다. 2차 변환은 LFNST 일 수도 있다. 변환 프로세싱 유닛 (206) 은 또한 선택된 변환 스킵의 표시를 제공할 수도 있고, 선택된 변환 스킵이 2차 변환을 포함하는 경우, 선택된 2차 변환의 표시를 엔트로피 인코딩 유닛 (220) 에 제공할 수도 있다. 엔트로피 인코딩 유닛 (220) 은 결국 선택된 변환 스킵을 나타내는 제 1 코드워드 (선택된 변환 스킵이 2차 변환을 포함하는지 여부를 또한 표시할 수도 있음) 를 인코딩할 수도 있다. 선택된 변환 스킵이 LFNST 와 같은 2차 변환을 포함하는 경우, 엔트로피 인코딩 유닛 (220) 은 추가로 선택된 2차 변환을 나타내는 제 2 코드워드를 인코딩할 수도 있다. 비디오 인코더 (200) 는, 예를 들어 위에 논의된 바와 같이, 1차 변환이 DCT-2 수평 변환 및 DCT-2 수직 변환을 포함하는 경우, 선택된 변환 스킵이 2차 변환을 포함한다고 결정할 수도 있다. 더욱이, 변환 프로세싱 유닛 (206) 은 1차 변환을 잔차 블록에 적용할 수도 있다. 선택된 변환 스킵이 2차 변환을 포함하는 경우, 변환 프로세싱 유닛 (206) 은 또한 1차 변환에 후속하여 2차 변환을 적용할 수 있다.

[0282] 양자화 유닛 (208) 은 양자화된 변환 계수 블록을 생성하기 위해 변환 계수 블록에서의 변환 계수들을 양자화할 수도 있다. 양자화 유닛 (208) 은 현재 블록과 연관된 양자화 파라미터 (QP) 값에 따라 변환 계수 블록의 변환 계수들을 양자화할 수도 있다. 비디오 인코더 (202) 는 (예를 들어, 모드 선택 유닛 (202) 을 통해) CU 와 연관된 QP 값을 조정함으로써 현재 블록과 연관된 계수 블록들에 적용된 양자화도를 조정할 수도 있다. 양자화는 정보의 손실을 도입할 수도 있으며, 따라서 양자화된 변환 계수들은 변환 프로세싱 유닛 (206) 에 의해 생성된 원래의 변환 계수들보다 더 낮은 정확도를 가질 수도 있다.

[0283] 역 양자화 유닛 (210) 및 역 변환 프로세싱 유닛 (212) 은 각각 양자화된 변환 계수 블록에 역 양자화 및 역 변환들을 적용하여, 변환 계수 블록으로부터 잔차 블록을 복원할 수도 있다. 본 개시의 기법들에 따라, 역 변환 프로세싱 유닛 (212) 은 변환 계수들에 역 2차 변환을 적용한 다음 역 1차 변환을 적용할 수도 있다. 복원 유닛 (214) 은 모드 선택 유닛 (202) 에 의해 생성된 예측 블록 및 복원된 잔차 블록에 기초하여 (잠재적으로 어느 정도의 왜곡을 가짐에도 불구하고) 현재 블록에 대응하는 복원된 블록을 생성할 수도 있다. 예를 들어, 복원 유닛 (214) 은 복원된 잔차 블록의 샘플들을, 모드 선택 유닛 (202) 에 의해 생성된 예측 블록으로부터의 대응하는 샘플들에 가산하여 복원된 블록을 생성할 수도 있다.

[0284] 필터 유닛 (216) 은 복원된 블록에 대해 하나 이상의 필터 동작들을 수행할 수도 있다. 예를 들어, 필터 유닛 (216) 은 CU들의 에지들을 따라 블록크니스 아티팩트 (blockiness artifacts) 를 감소시키기 위해 디블록킹 동작들을 수행할 수도 있다. 필터 유닛 (216) 의 동작들은 일부 예들에서 스kip될 수도 있다.

[0285] 비디오 인코더 (200) 는 DPB (218) 에 복원된 블록들을 저장한다. 예를 들어, 필터 유닛 (216) 의 동작들

이 필요하지 않은 예들에서, 복원 유닛 (214) 은 복원된 블록들을 DPB (218) 에 저장할 수도 있다. 필터 유닛 (216) 의 동작들이 필요한 예들에서, 필터 유닛 (216) 은 필터링된 복원된 블록들을 DPB (218) 에 저장할 수도 있다. 모션 추정 유닛 (222) 및 모션 보상 유닛 (224) 은 복원된 (및 잠재적으로 필터링된) 블록들로부터 형성된 DPB (218) 로부터 참조 픽처를 취출하여, 후속하여 인코딩된 픽처들의 블록들을 인터-예측할 수도 있다. 또한, 인트라-예측 유닛 (226) 은 현재 픽처에서의 다른 블록들을 인트라-예측하기 위해 현재 픽처의 DPB (218) 에서 복원된 블록들을 사용할 수도 있다.

[0286] 일반적으로, 엔트로피 인코딩 유닛 (220) 은 비디오 인코더 (200) 의 다른 기능성 컴포넌트들로부터 취출된 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (220) 은 양자화 유닛 (208) 으로부터 양자화된 변환 계수 블록들을 엔트로피 인코딩할 수도 있다. 또 다른 예로서, 엔트로피 인코딩 유닛 (220) 은 모드 선택 유닛 (202) 으로부터 예측 신택스 엘리먼트들 (예를 들어, 인트라-예측에 대한 인트라-모드 정보 또는 인터-예측에 대한 모션 정보) 를 엔트로피 인코딩할 수도 있다. 엔트로피 인코딩 유닛 (220) 은 엔트로피 인코딩된 데이터를 생성하기 위해, 비디오 데이터의 또 다른 예인, 신택스 엘리먼트들에 대해 하나 이상의 엔트로피 인코딩 동작들을 수행할 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (220) 은 컨텍스트 적응 가변 길이 코딩 (CAVLC) 동작, CABAC 동작, V2V (variable-to-variable) 길이 코딩 동작, 신택스 기반 컨텍스트 적응 바이너리 산술 코딩 (SBAC) 동작, 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩 동작, 지수-골름 인코딩 동작, 또는 다른 타입의 엔트로피 인코딩 동작을 데이터에 대해 수행할 수도 있다. 일부 예들에서, 엔트로피 인코딩 유닛 (220) 은 신택스 엘리먼트들이 엔트로피 인코딩되지 않은 바이패스 모드에서 동작할 수도 있다.

[0287] 비디오 인코더 (200) 는 픽처 또는 슬라이스의 블록들을 복원하는데 필요한 엔트로피 인코딩된 신택스 엘리먼트들을 포함하는 비트스트림을 출력할 수도 있다. 특히, 엔트로피 인코딩 유닛 (220) 이 비트스트림을 출력할 수도 있다.

[0288] 상술한 동작들은 블록과 관련하여 설명된다. 이러한 설명은 루마 코딩 블록 및/또는 크로마 코딩 블록들에 대한 동작들인 것으로 이해되어야 한다. 상술한 바와 같이, 일부 예들에서, 루마 코딩 블록 및 크로마 코딩 블록들은 CU 의 루마 및 크로마 성분들이다. 일부 예들에서, 루마 코딩 블록 및 크로마 코딩 블록들은 PU 의 루마 및 크로마 성분들이다.

[0289] 일부 예들에서, 루마 코딩 블록에 대해 수행되는 동작들은 크로마 코딩 블록에 대해 반복될 필요가 없다. 일 예로서, 크로마 블록들에 대한 모션 벡터 (MV) 및 참조 픽처를 식별하기 위해 루마 코딩 블록에 대한 MV 및 참조 픽처를 식별하는 동작들이 반복될 필요는 없다. 오히려, 루마 코딩 블록에 대한 MV 는 크로마 블록들에 대한 MV 를 결정하도록 스케일링될 수도 있고, 참조 픽처는 동일할 수도 있다. 다른 예로서, 인트라-예측 프로세스는 루마 코딩 블록들 및 크로마 코딩 블록들에 대해 동일할 수도 있다.

[0290] 비디오 인코더 (200) 는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로부에서 구현되는 하나 이상의 프로세싱 유닛들을 포함하는, 비디오 데이터를 인코딩하도록 구성된 디바이스의 예를 나타내며, 하나 이상의 프로세싱 유닛들은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스킴을 나타내는 제 1 코드워드를 코딩하는 것으로서, 선택된 변환 스킴은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하고, 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하며; 그리고 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하도록 구성된다.

[0291] 도 13 은 본 개시의 기법들을 수행할 수도 있는 예시적인 비디오 디코더 (300) 를 도시하는 블록 다이어그램이다. 도 13 은 설명의 목적을 위해 제공되고 본 개시에 폭넓게 예시되고 설명된 기법들에 대해 한정하지 않는다. 설명의 목적으로, 본 개시는 비디오 디코더 (300) 가 VVC 및 HEVC 의 기법들에 따라 기술되는 것을 설명한다. 그러나, 본 개시의 기법들은 다른 비디오 코딩 표준들로 구성되는 비디오 코딩 디바이스들에 의해 수행될 수도 있다.

[0292] 도 13 의 예에서, 비디오 디코더 (300) 는 코딩된 픽처 버퍼 (CPB)(320), 엔트로피 디코딩 유닛 (302), 예측 프로세싱 유닛 (304), 역 양자화 유닛 (306), 역 변환 프로세싱 유닛 (310), 복원 유닛 (310), 필터 유닛 (312), 및 디코딩된 픽처 버퍼 (DPB)(314) 를 포함한다. 도 13 은 상기 도 4 에 나타낸 바와 같이, 역 변환 프로세싱 유닛 (308) 이 본 개시의 기법들에 따른 변환들을 선택하는 변환 뱅크를 더 포함할 수도 있다. 마찬가지로, 도 4 에 나타낸 기법들과 반대로, 엔트로피 디코딩 유닛 (302) 은 MTS 스킴에 대한 다양한 변환들 중 어느 것이 비디오 데이터의 현재 블록에 대해 선택되는지를 나타내는 본 개시의 기법들에 따라 데이터를 디코딩하고

역 변환 프로세싱 유닛(308)에 변환의 표시를 제공할 수도 있다.

[0293] 예측 프로세싱 유닛(304)은 모션 보상 유닛(316) 및 인트라-예측 유닛(318)을 포함한다. 예측 프로세싱 유닛(304)은 다른 예측 모드들에 따라 예측을 수행하기 위해 부가적인 유닛들을 포함할 수도 있다. 예를 들어, 예측 프로세싱 유닛(304)은 팔레트 유닛, 인트라-블록 카피 유닛(모션 보상 유닛(316)의 일부를 형성할 수도 있음), 아핀 유닛, 선형 모델(LM) 유닛 등을 포함할 수도 있다. 다른 예들에서, 비디오 디코더(300)는 더 많거나, 더 적거나, 또는 상이한 기능성 컴포넌트들을 포함할 수도 있다.

[0294] CPB 메모리(320)는 비디오 디코더(300)의 컴포넌트들에 의해 디코딩된 인코딩된 비디오 비트스트림과 같은 비디오 데이터를 저장할 수도 있다. CPB 메모리(320)에 저장된 비디오 데이터는, 예를 들어 컴퓨터 판독 가능 매체(110)(도 1)로부터 획득될 수도 있다. CPB 메모리(320)는 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터(예를 들어, 신팩스 엘리먼트들)를 저장하는 CPB를 포함할 수도 있다. 또한, CPB 메모리(320)는 비디오 디코더(300)의 다양한 유닛들로부터의 출력들을 나타내는 일시적인 데이터와 같은, 코딩된 픽처의 신팩스 엘리먼트들 이외의 비디오 데이터를 저장할 수도 있다. DPB(314)는 일반적으로, 인코딩된 비디오 비트스트림의 후속 데이터 또는 픽처들을 디코딩할 때, 참조 비디오 데이터로서 비디오 디코더(300)가 출력하고 및/또는 사용할 수도 있는 디코딩된 픽처들을 저장한다. CPB 메모리(320) 및 DPB(314)는 다양한 메모리 디바이스들, 예컨대 동기식 동적 랜덤 액세스 메모리(SDRAM), 자기저항 RAM(MRAM), 저항 RAM(RRAM)을 포함하는 DRAM, 또는 다른 타입들의 메모리 디바이스들 중 임의의 것에 의해 형성될 수도 있다. CPB 메모리(320) 및 DPB(314)는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, CPB 메모리(320)는 비디오 디코더(300)의 다른 컴포넌트들과 온-칩이거나 그 컴포넌트들에 대하여 오프-칩일 수도 있다.

[0295] 부가적으로 또는 대안으로, 일부 예들에서, 비디오 디코더(300)는 메모리(120)(도 1)로부터 코딩된 비디오 데이터를 취출할 수도 있다. 즉, 메모리(120)는 CPB 메모리(320)로 위에서 논의된 바와 같이 데이터를 저장할 수도 있다. 마찬가지로, 메모리(120)는 비디오 디코더(300)의 기능성의 일부 또는 전부가 비디오 디코더(300)의 프로세싱 회로부에 의해 실행되는 소프트웨어에서 구현될 때, 비디오 디코더(300)에 의해 실행될 명령들을 저장할 수도 있다.

[0296] 도 13에 나타낸 다양한 유닛들은 비디오 인코더(300)에 의해 수행되는 동작들의 이해를 돋기 위해 도시된다. 이 유닛들은 고정 기능 회로들, 프로그램가능 회로들, 또는 이들의 조합으로서 구현될 수도 있다. 도 12와 유사하게, 고정 기능 회로들은 특정 기능성을 제공하는 회로들을 지칭하며, 수행될 수 있는 동작들에 대해 미리설정된다. 프로그램가능 회로들은 다양한 태스크들을 수행하도록 프로그램될 수 있는 회로들을 지칭하며, 수행될 동작들에서 유연한 기능성을 제공한다. 예를 들어, 프로그램가능 회로들은 프로그램가능 회로들이 소프트웨어 또는 펌웨어의 명령들에 의해 정의된 방식으로 동작하게 하는 소프트웨어 또는 펌웨어를 실행할 수도 있다. 고정 기능 회로들은 소프트웨어 명령들을(예를 들어, 파라미터들을 수신하거나 파라미터들을 출력하기 위해) 실행할 수도 있지만, 고정 기능 회로들이 수행하는 동작 타입들은 일반적으로 불변이다. 일부 예들에서, 유닛들의 하나 이상은 별개의 회로 블록들(고정 기능 또는 프로그램가능)일 수도 있고, 일부 예들에서, 하나 이상의 유닛들은 집적 회로들일 수도 있다.

[0297] 비디오 디코더(300)는 프로그램가능 회로들로부터 형성된, ALU들, EFU들, 디지털 회로들, 아날로그 회로들, 및/또는 프로그램가능 코어들을 포함할 수도 있다. 비디오 디코더(300)의 동작들이 프로그램가능 회로들 상에서 실행하는 소프트웨어에 의해 수행되는 예들에서, 온-칩 또는 오프-칩 메모리는 비디오 디코더(300)가 수신하고 실행하는 소프트웨어의 명령들(예를 들어, 오브젝트 코드)을 저장할 수도 있다.

[0298] 엔트로피 디코딩 유닛(302)은 인코딩된 비디오 데이터를 CPB로부터 수신하고, 그 비디오 데이터를 엔트로피 디코딩하여 신팩스 엘리먼트들을 재생할 수도 있다. 예측 프로세싱 유닛(304), 역 양자화 유닛(306), 역 변환 프로세싱 유닛(308), 복원 유닛(310), 및 필터 유닛(312)은 비트스트림으로부터 추출된 신팩스 엘리먼트들에 기초하여 디코딩된 비디오 데이터를 생성할 수도 있다.

[0299] 본 개시의 기법들에 따라, 엔트로피 디코딩 유닛(302)은 비디오 데이터의 현재 블록에 대한 디코딩된 변환 계수들에 적용될 변환 스킴을 나타내는 제 1 코드워드를 디코딩할 수도 있다. 엔트로피 디코딩 유닛(302)은 추가로, 선택된 변환 스킴이 1차 변환에 부가하여 적용될 2차 변환(예를 들어, LFNST)을 포함하는지 여부를 결정할 수도 있다. 예를 들어, 1차 변환이 DCT-2 수평 변환 및 DCT-2 수직 변환을 포함하는 경우, 엔트로피 디코딩 유닛(302)은 2차 변환이 또한 적용되어야 한다고 결정할 수도 있다. 또한, 2차 변환이 적용되어야 한다고 결정하는 것에 응답하여, 엔트로피 디코딩 유닛(302)은 또한 가용 2차 변환들의 세트의 2차 변환을 나

타내는 제 2 코드워드를 디코딩할 수도 있다.

[0300] 일반적으로, 비디오 디코더 (300) 는 블록 별 (block-by-block) 단위로 꾹처를 복원한다. 비디오 디코더 (300) 는 개별적으로 (현재 복원되고 있는, 즉 디코딩되는 블록이 "현재 블록"으로 지정될 수도 있는 경우) 각각의 블록에 대해 복원 동작을 수행할 수도 있다.

[0301] 엔트로피 디코딩 유닛 (302) 은 양자화 파라미터 (QP) 및/또는 변환 모드 표시(들)과 같은 변환 정보 뿐만 아니라, 양자화된 변환 계수 블록의 양자화된 변환 계수들을 정의하는 선택스 엘리먼트들을 엔트로피 디코딩할 수도 있다. 역 양자화 유닛 (306)은 양자화된 변환 계수 블록과 연관된 QP 를 사용하여, 양자화도 및 유사하게, 적용할 역 양자화 유닛 (306)에 대한 역 양자화도를 결정할 수도 있다. 역 양자화 유닛 (306)은 예를 들어, 양자화된 변환 계수들을 역 양자화하기 위해 비트단위 (bitwise) 좌측-시프트 동작을 수행할 수도 있다. 따라서, 역 양자화 유닛 (306)은 변환 계수들을 포함하는 변환 계수 블록을 형성할 수도 있다.

[0302] 역 양자화 유닛 (306)이 변환 계수 블록을 형성한 후, 역 변환 프로세싱 유닛 (308)은 현재 블록과 연관된 잔차 블록을 생성하기 위해 변환 계수 블록에 하나 이상의 역 변환들을 적용할 수도 있다. 예를 들어, 역 변환 프로세싱 유닛 (308)은 역 DCT, 역 정수 변환, 역 Karhunen-Loeve 변환 (KLT), 역 회전 변환, 역 방향성 변환, 또는 다른 역 변환을 계수 블록에 적용할 수도 있다. 변환 스킴이 2차 변환을 포함하는 경우, 역 양자화 유닛 (306)은 1차 변환을 적용하기 전에 2차 변환을 적용할 수도 있다.

[0303] 또한, 예측 프로세싱 유닛 (304)은 엔트로피 디코딩 유닛 (302)에 의해 엔트로피 디코딩된 예측 정보 선택스 엘리먼트들에 따라 예측 블록을 생성한다. 예를 들어, 예측 정보 선택스 엘리먼트들이 현재 블록이 인터-예측된 것을 표시하면, 모션 보상 유닛 (316)은 예측 블록을 생성할 수도 있다. 이 경우, 예측 정보 선택스 엘리먼트들은 참조 블록을 취출할 DPB (314)에서의 참조 꾹처뿐만 아니라 현재 꾹처에서의 현재 블록의 위치에 대한 참조 꾹처에서의 참조 블록의 위치를 식별하는 모션 벡터를 표시할 수도 있다. 모션 보상 유닛 (316)은 일반적으로 모션 보상 유닛 (224)(도 12)과 관련하여 설명된 것과 실질적으로 유사한 방식으로 인터-예측 프로세스를 수행할 수도 있다.

[0304] 다른 예로서, 예측 정보 선택스 엘리먼트가 현재 블록이 인트라-예측되는 것을 표시하면, 인트라-예측 유닛 (318)은 예측 정보 선택스 엘리먼트들에 의해 표시된 인트라-예측 모드에 따라 예측 블록을 생성할 수도 있다. 다시, 인트라-예측 유닛 (318)은 일반적으로 인트라-예측 유닛 (226)(도 12)과 관련하여 설명된 것과 실질적으로 유사한 방식으로 인트라-예측 프로세스를 수행할 수도 있다. 인트라-예측 유닛 (318)은 DPB (314)로부터 현재 블록에 이웃하는 샘플들의 데이터를 취출할 수도 있다.

[0305] 복원 유닛 (310)은 예측 블록 및 잔차 블록을 사용하여 현재 블록을 복원한다. 예를 들어, 복원 유닛 (310)은 잔차 꾹셀 블록의 샘플들을 예측 블록의 대응하는 샘플들에 가산하여 현재 블록을 복원할 수도 있다.

[0306] 필터 유닛 (312)은 복원된 블록들에 대해 하나 이상의 필터 동작들을 수행할 수도 있다. 예를 들어, 필터 유닛 (312)은 복원된 블록들의 에지들을 따라 블록크니스 아티팩트를 감소시키기 위해 디블록킹 동작들을 수행할 수도 있다. 필터 유닛 (312)의 동작들이 모든 예들에서 반드시 수행되지는 않는다.

[0307] 비디오 디코더 (300)는 DPB (314)에 복원된 블록들을 저장할 수도 있다. 위에 논의된 바와 같이, DPB (314)는 예측 프로세싱 유닛 (304)에 인트라-예측을 위한 현재 꾹처의 샘플들 및 후속 모션 보상을 위해 이전에 디코딩된 꾹처들과 같은 참조 정보를 제공할 수도 있다. 또한, 비디오 디코더 (300)는 도 1의 디스플레이 디바이스 (118)와 같은 디스플레이 디바이스 상으로의 후속 프리젠테이션을 위해 DPB로부터 디코딩된 꾹처들을 출력할 수도 있다.

[0308] 비디오 디코더 (300)는 비디오 데이터를 저장하도록 구성된 메모리, 및 회로부에서 구현되는 하나 이상의 프로세싱 유닛들을 포함하는, 비디오 디코딩 디바이스의 예를 나타내며, 하나 이상의 프로세싱 유닛들은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스킴을 나타내는 제 1 코드워드를 코딩하는 것으로서, 선택된 변환 스킴은 1차 변환에 부가하여 적용될 가능 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하고; 가능 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하며; 그리고 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하도록 구성된다.

[0309] 도 14는 본 개시의 기법들에 따른 현재 블록을 인코딩하기 위한 예시의 방법을 도시하는 플로우차트이다. 현재 블록은 현재 CU 를 포함할 수도 있다. 비디오 인코더 (200)(도 1 및 도 9)와 관련하여 설명되지만, 도 14 와 유사한 방법을 수행하도록 다른 디바이스들이 구성될 수도 있음을 이해해야 한다.

[0310] 본 예에서, 비디오 인코더 (200) 는 초기에 현재 블록을 예측한다 (350). 예를 들어, 비디오 인코더 (200) 는 현재 블록에 대한 예측 블록을 형성할 수도 있다. 비디오 인코더 (200) 는 그 후 현재 블록에 대한 잔차 블록을 계산할 수도 있다 (352). 잔차 블록을 계산하기 위해, 비디오 인코더 (200) 는 원래의, 코딩되지 않은 블록과 현재 블록에 대한 예측 블록 사이의 차이를 계산할 수도 있다. 비디오 인코더 (200) 는 그 후 변환을 선택하고 선택된 변환을 사용하여 잔차 블록의 계수들을 양자화할 수도 있다 (354). 선택된 변환은 1차 변환 및/또는 2차 변환, 예컨대 LFNST 를 포함할 수도 있다. 비디오 인코더 (200) 는 선택된 변환에 따라 1차 변환 및/또는 2차 변환 중 하나 또는 양자 모두를 적용할 수도 있다.

[0311] 다음으로, 비디오 인코더 (200) 는 잔차 블록의 양자화된 변환 계수들을 스캔할 수도 있다 (356). 스캔 동안, 또는 스캔에 후속하여, 비디오 인코더 (200) 는 선택된 변환을 나타내는 데이터 뿐만 아니라, 계수들을 엔트로피 인코딩할 수도 있다 (358). 예를 들어, 비디오 인코더 (200) 는 위에 논의된 바와 같이 본 개시의 다양한 기법들 중 임의의 것을 사용하여 변환을 나타내는 데이터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (200) 는 CAVLC 또는 CABAC 를 사용하여 계수들을 인코딩할 수도 있다. 특히, 비디오 인코더 (200) 는 본 개시의 기법들에 따라 변환 스킵을 선택하고 본 개시의 기법들 중 임의의 것에 따라 선택된 변환을 나타내는 코드워드를 엔트로피 인코딩할 수도 있다. 선택된 변환 스킵이 2차 변환을 포함하는 경우, 비디오 인코더 (200) 는 추가로, 예를 들어 도 8 과 관련하여 위에 논의된 바와 같이, 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 인코딩할 수도 있다. 비디오 인코더 (200) 는 그 후 블록의 계수들 및 변환(들) 을 나타내는 엔트로피 인코딩된 데이터를 출력할 수도 있다 (360).

[0312] 이러한 방식으로, 도 14 의 방법은 비디오 데이터를 인코딩하는 방법의 예를 나타내며, 방법은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 단계로서, 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 단계; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하는 단계; 및 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하는 단계를 포함한다.

[0313] 도 15 는 본 개시의 기법들에 따른 비디오 데이터의 현재 블록을 디코딩하기 위한 예시의 방법을 도시하는 플로우차트이다. 현재 블록은 현재 CU 를 포함할 수도 있다. 비디오 디코더 (300)(도 1 및 도 13) 와 관련하여 설명되지만, 도 15 와 유사한 방법을 수행하도록 다른 디바이스들이 구성될 수도 있음을 이해해야 한다.

[0314] 비디오 디코더 (300) 는 현재 블록에 대응하는 잔차 블록의 계수들에 대한 엔트로피 코딩된 예측 정보 및 엔트로피 코딩된 데이터와 같은, 현재 블록에 대한 엔트로피 코딩된 데이터를 수신할 수도 있다 (370). 비디오 디코더 (300) 는 엔트로피 코딩된 데이터를 엔트로피 디코딩하여 현재 블록에 대한 예측 정보, 현재 블록에 대한 변환을 결정하고, 잔차 블록의 계수들을 재생할 수도 있다 (372). 비디오 디코더 (300) 는 본 개시의 다양한 기법들 중 임의의 것에 따라 변환 정보를 엔트로피 디코딩할 수도 있다. 비디오 디코더 (300) 는 현재 블록에 대한 예측 블록을 계산하기 위해, 예를 들어 현재 블록에 대한 예측 정보에 의해 표시된 바와 같이 인트라- 또는 인터-예측 모드를 사용하여, 현재 블록을 예측할 수도 있다 (374). 비디오 디코더 (300) 는 그 후 양자화된 변환 계수들의 블록을 생성하기 위해 재생된 계수들을 역 스캔할 수도 있다 (376). 비디오 디코더 (300) 는 그 후 표시된 변환을 사용하여 계수들을 역 양자화 및 역 변환하여 잔차 블록을 생성할 수도 있다 (378). 예를 들어, 비디오 디코더 (300) 는 본 개시의 기법들 중 임의의 것에 따라 적용될 변환을 나타내는 코드워드를 디코딩할 수도 있다. 비디오 디코더 (300) 는 예측 블록 및 잔차 블록을 조합함으로써 결국 현재 블록을 디코딩할 수도 있다 (380).

[0315] 이러한 방식으로, 도 15 의 방법은 비디오 데이터를 디코딩하는 방법의 예를 나타내며, 방법은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킵의 변환 후보들의 세트의 선택된 변환 스킵을 나타내는 제 1 코드워드를 코딩하는 단계로서, 선택된 변환 스킵은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 단계; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하는 단계; 및 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하는 단계를 포함한다.

[0316] 도 16 은 본 개시의 기법들에 따른 예시의 비디오 인코딩 방법을 도시하는 플로우차트이다. 예시의 목적으로, 도 16 의 방법은 도 1 및 도 12 의 비디오 인코더 (200) 와 관련하여 설명되지만, 다른 비디오 인코더들이 이를 수행하거나 유사한 방법을 수행하도록 구성될 수도 있음을 이해해야 한다.

[0317] 초기에, 비디오 인코더 (200) 는 1차 및 2차 변환을 포함하는 변환 스킵을 선택할 수도 있다 (400). 모드

선택 유닛 (202) 은 또한 가용 2차 변환들의 세트로부터 2차 변환을 선택할 수도 있다 (402). 예를 들어, 모드 선택 유닛 (202) 은 비디오 인코더 (200) 의 다양한 컴포넌트들로 하여금 다양한 변환 스킴들을 테스팅하는 것을 포함하는, 다양한 인코딩 패스들을 수행하게 할 수도 있다. 모드 선택 유닛 (202) 은 레이트-왜곡 메트릭을 계산하고 1차 및 선택된 2차 변환을 포함하는 선택된 변환 스킴이 최상의 테스트된 레이트-왜곡 특징들을 산출한다고 결정할 수도 있다.

[0318] 비디오 인코더 (200) 는 그 후 선택된 변환 스킴을 나타내는 제 1 코드워드를 인코딩할 수도 있다 (404). 부가적으로, 비디오 인코더 (200) 는 선택된 2차 변환 스킴을 나타내는 제 2 코드워드를 인코딩할 수도 있다 (406). 특히, 엔트로피 인코딩 유닛 (220) 은 제 1 및 제 2 코드워드들을 엔트로피 인코딩할 수도 있다.

[0319] 비디오 인코더 (200) 는 그 후 잔차 블록에 1차 변환을 적용할 수도 있다 (408). 특히, 변환 프로세싱 유닛 (206) 은 1차 변환을 잔차 블록에 적용하여, 변환 계수들의 변환 블록을 생성할 수도 있다. 비디오 인코더 (200)(특히, 변환 프로세싱 유닛 (206)) 는 또한 2차 변환을 변환 블록에 적용할 수도 있다 (410).

[0320] 이러한 방식으로, 도 16 의 방법은 비디오 데이터를 인코딩하는 방법의 예를 나타내며, 방법은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스킴을 나타내는 제 1 코드워드를 코딩하는 단계로서, 선택된 변환 스킴은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 단계; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하는 단계; 및 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하는 단계를 포함한다.

[0321] 도 17 은 본 개시의 기법들에 따른 예시의 비디오 디코딩 방법을 도시하는 플로우차트이다. 예시의 목적으로, 도 17 의 방법은 도 1 및 도 13 의 비디오 디코더 (300) 와 관련하여 설명되지만, 다른 비디오 인코더들이 이를 수행하거나 유사한 방법을 수행하도록 구성될 수도 있음을 이해해야 한다.

[0322] 비디오 디코더 (300) 는 1차 변환 및 2차 변환 양자 모두를 포함하는 변환 스킴을 나타내는 제 1 코드워드를 초기에 디코딩할 수도 있다 (420). 특히, 엔트로피 디코딩 유닛 (302) 은 제 1 코드워드를 엔트로피 디코딩할 수도 있다. 비디오 디코더 (300) 의 엔트로피 디코딩 유닛 (302) 은 또한 가용 2차 변환들의 세트에서 2차 변환을 나타내는 제 2 코드워드를 엔트로피 디코딩할 수도 있다 (422). 예를 들어, 제 2 코드워드는 가용 2차 변환들의 세트로의 인덱스로서 작용할 수도 있다.

[0323] 비디오 디코더 (300) 는 그 후 중간 변환 블록을 생성하기 위해 변환 블록의 디코딩된 변환 계수들에 2차 변환을 적용할 수도 있다 (424). 비디오 디코더 (300) 는 또한 잔차 블록을 생성하기 위해 중간 변환 블록에 1차 변환을 적용할 수도 있다 (426). 특히, 역 변환 프로세싱 유닛 (308) 은 2차 및 1차 변환들을 적용할 수도 있다.

[0324] 이러한 방식으로, 도 17 의 방법은 비디오 데이터를 디코딩하는 방법의 예를 나타내며, 방법은 비디오 데이터의 현재 블록에 대한 다중 변환 선택 (MTS) 스킴의 변환 후보들의 세트의 선택된 변환 스킴을 나타내는 제 1 코드워드를 코딩하는 단계로서, 선택된 변환 스킴은 1차 변환에 부가하여 적용될 가용 2차 변환들의 세트의 2차 변환인, 상기 제 1 코드워드를 코딩하는 단계; 가용 2차 변환들의 세트로부터의 2차 변환을 나타내는 제 2 코드워드를 코딩하는 단계; 및 현재 블록에 대한 잔차 데이터의 코딩 동안 1차 변환 및 2차 변환을 적용하는 단계를 포함한다.

[0325] 예시에 의존하여, 본 명세서에서 설명된 기법들 중 임의의 것의 소정의 액트들 또는 이벤트들은 상이한 시퀀스로 수행될 수 있고, 전체적으로 부가되거나 병합되거나 또는 제거될 수도 있음 (예를 들어, 설명된 모든 액트들 또는 이벤트들이 그 기법들의 실시를 위해 필수적인 것은 아님) 이 인식되어야 한다. 더욱이, 소정의 예들에서, 액트들 또는 이벤트들은 순차적인 것보다는, 예를 들어 다중-스레딩된 프로세싱, 인터럽트 프로세싱, 또는 다중의 프로세서들을 통해 동시에 수행될 수도 있다.

[0326] 하나 이상의 예들에 있어서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합에서 구현될 수도 있다. 소프트웨어로 구현되는 경우, 그 기능들은 컴퓨터 판독가능 매체 상의 하나 이상의 명령들 또는 코드로서 저장되거나 이를 통해 송신될 수도 있고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체는 데이터 저장 매체와 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체, 또는 예를 들어, 통신 프로토콜에 따라, 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 가능하게 하는 임의의 매체를 포함하는 통신 매체를 포함할 수도 있다. 이러한 방식으로, 컴퓨터 판독가능 매체는 일반적으로, (1) 비일시적인 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대

응할 수도 있다. 데이터 저장 매체들은 본 개시에서 설명된 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 취출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체들일 수도 있다. 컴퓨터 프로그램 제품이 컴퓨터 판독가능 매체를 포함할 수도 있다.

[0327] 제한이 아닌 예로서, 이러한 컴퓨터 판독가능 저장 매체들은 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장, 자기 디스크 저장, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 명령들 또는 데이터 구조들의 형태로 원하는 프로그램 코드를 저장하기 위해 이용될 수 있으며 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 커넥션이 컴퓨터 판독가능 매체로 적절히 명명된다. 예를 들어, 동축 케이블, 광섬유 케이블, 꼬임쌍선, 디지털가입자 라인 (DSL), 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들을 이용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 명령들이 송신된다면, 동축 케이블, 광섬유 케이블, 꼬임쌍선, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 매체의 정의에 포함된다.

하지만, 컴퓨터 판독가능 저장 매체들 및 데이터 저장 매체들은 커넥션들, 캐리어파들, 신호들, 또는 다른 일시적 매체들을 포함하지 않지만 대신 비일시적인 유형의 저장 매체들로 지향됨이 이해되어야 한다. 본 명세서에서 사용된 바와 같이 디스크 (disk) 와 디스크 (disc) 는, 컴팩트 디스크(CD), 레이저 디스크, 광학 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크, 및 블루레이 디스크를 포함하며, 여기서 디스크 (disk) 들은 통상 자기적으로 데이터를 재생하는 한편, 디스크(disc) 들은 레이저들을 이용하여 광학적으로 데이터를 재생한다. 또한, 상기의 조합은 컴퓨터 판독 가능 매체의 범위 내에 포함되어야 한다.

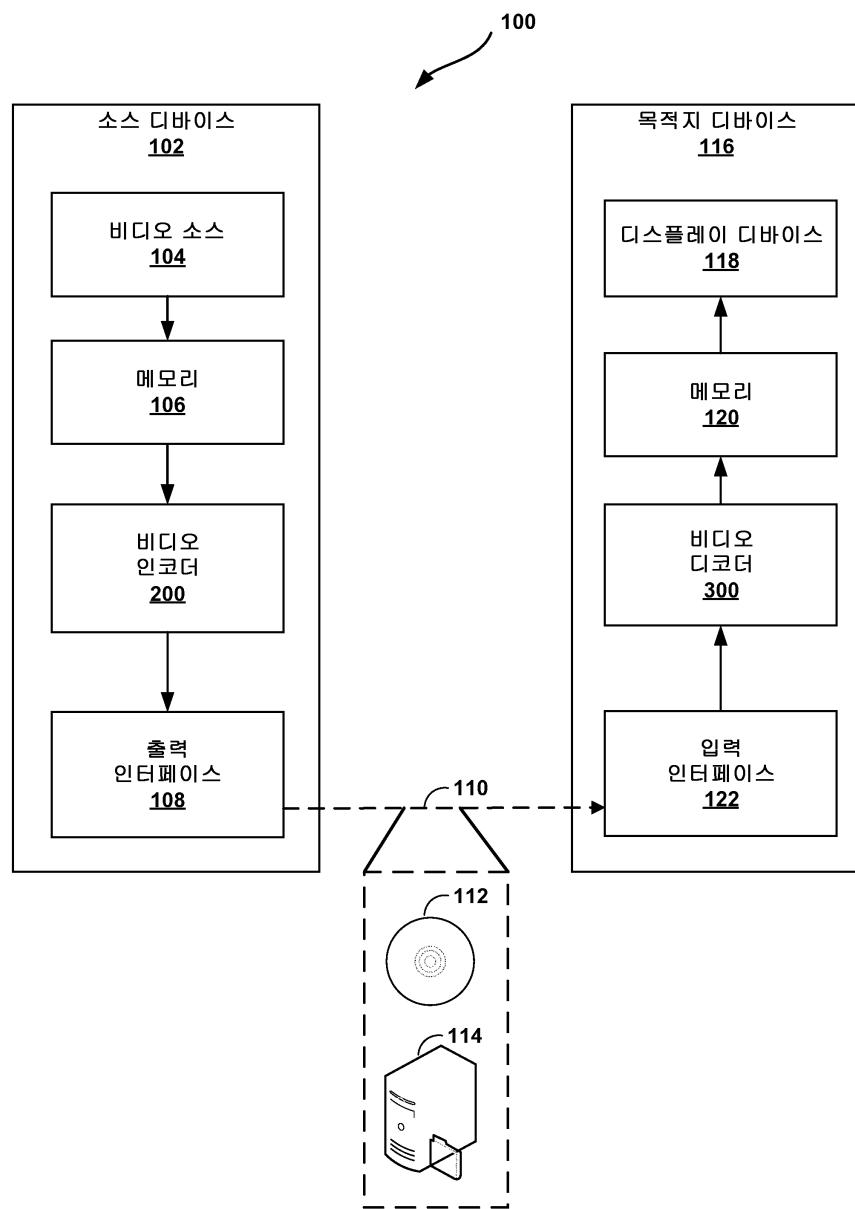
[0328] 명령들은 하나 이상의 프로세서, 이를 테면 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 또는 다른 등가의 집적 또는 이산 로직 회로부에 의해 실행될 수도 있다. 따라서, 본원에 사용된 용어 "프로세서"는 전술한 구조 중 임의의 것 또는 본원에 설명된 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 부가적으로, 일부 양태들에 있어서, 본 명세서에서 설명된 기능성은 인코딩 및 디코딩을 위해 구성되거나 또는 결합된 코덱에서 통합된 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 그 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

[0329] 본 개시의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC 들의 세트 (예를 들어, 칩 세트) 를 포함하여, 광범위하게 다양한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들 또는 유닛들이, 개시된 기술들을 수행하도록 구성된 디바이스들의 기능적인 양태들을 강조하기 위하여 본 개시에 설명되었지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하는 것은 아니다. 오히려, 상술한 바와 같이, 다양한 유닛들이 코덱 하드웨어 유닛에 결합될 수도 있거나, 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 상술한 하나 이상의 프로세서들을 포함하는 상호동작 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

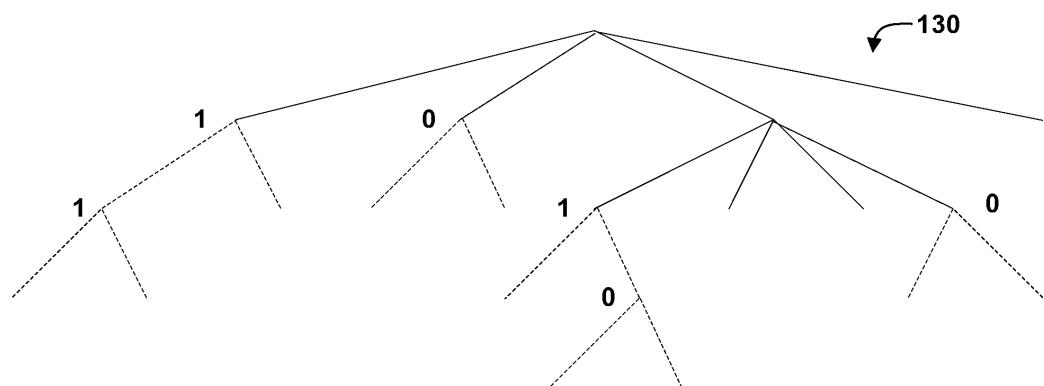
[0330] 다양한 예들이 설명되었다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

도면1

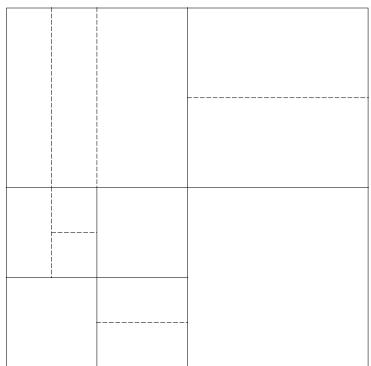


도면2a

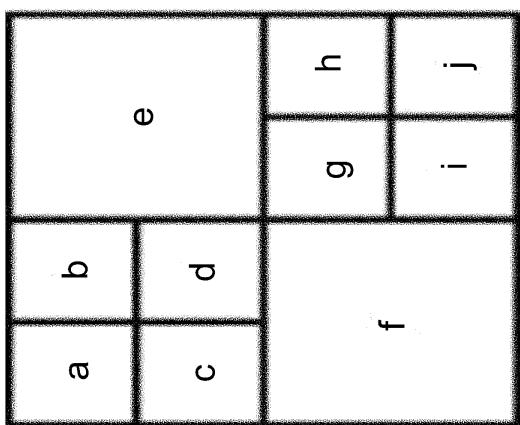


도면2b

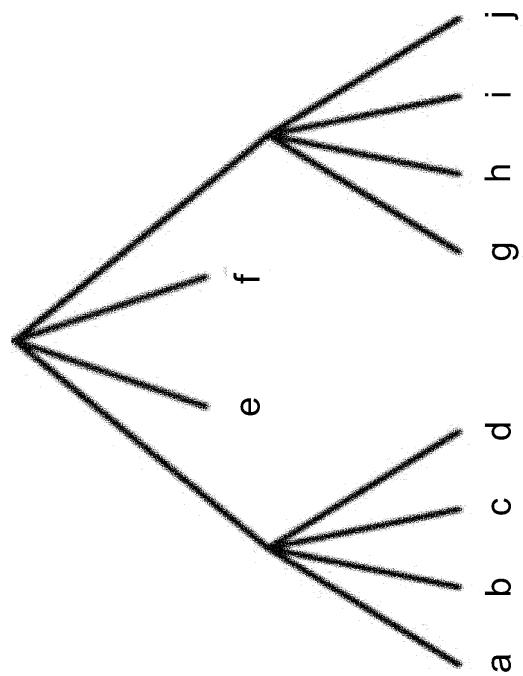
132



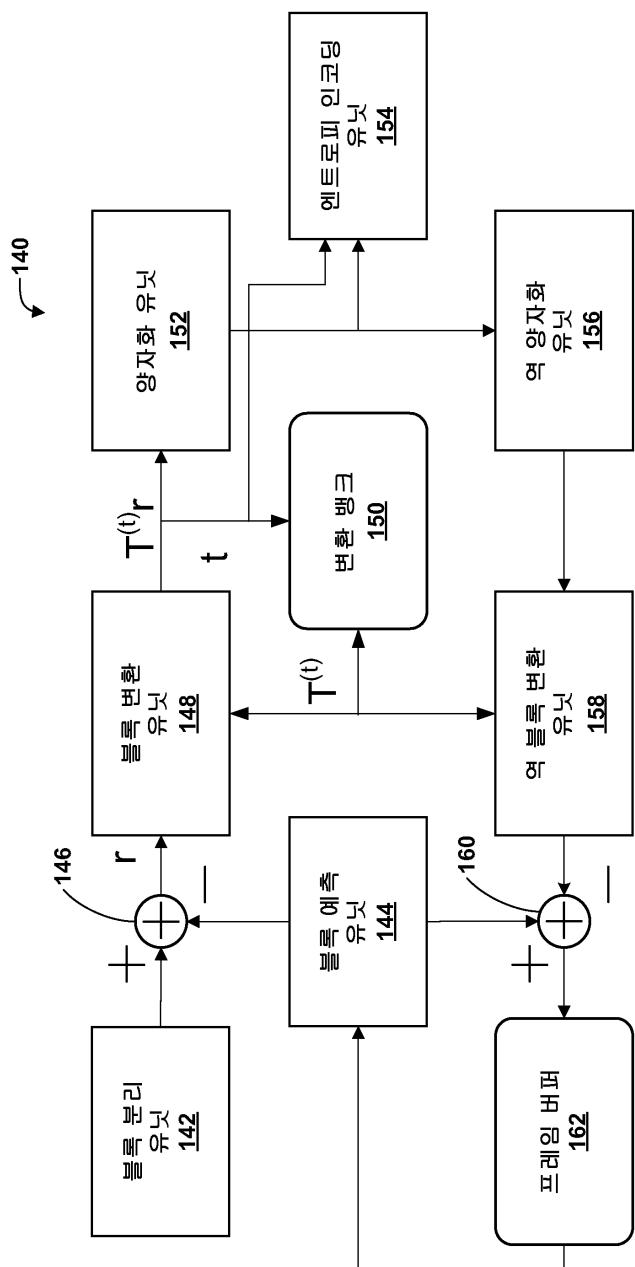
도면3a



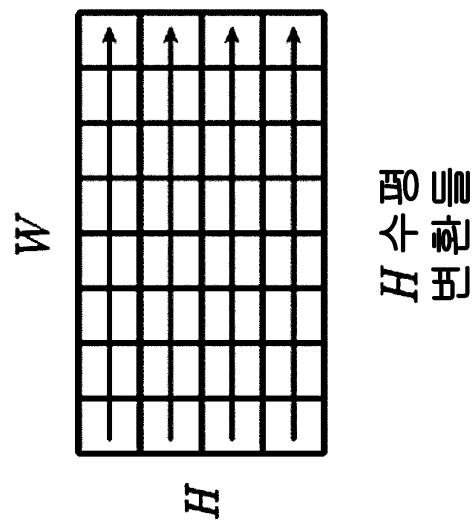
도면3b



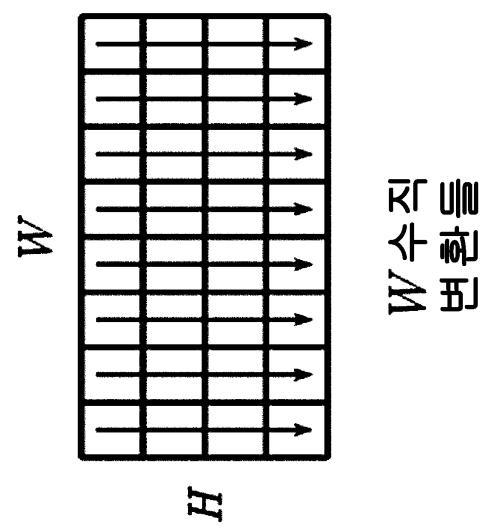
도면4



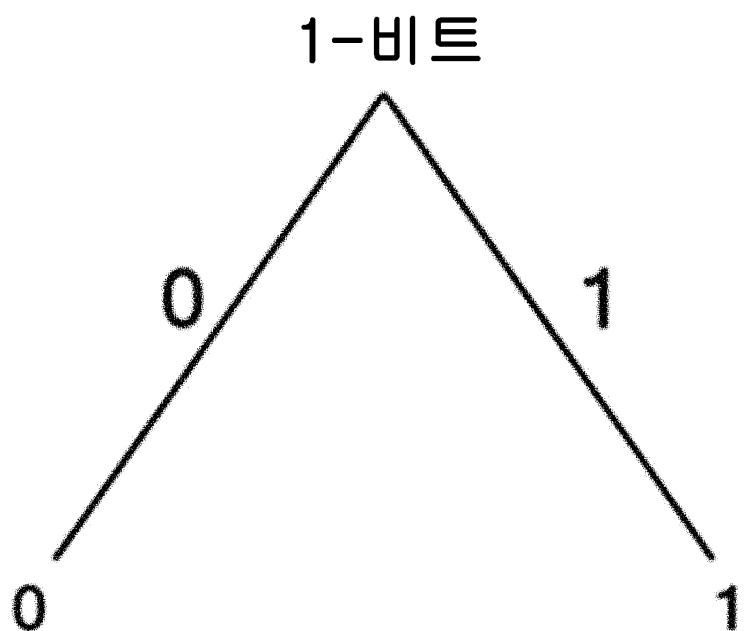
도면5a



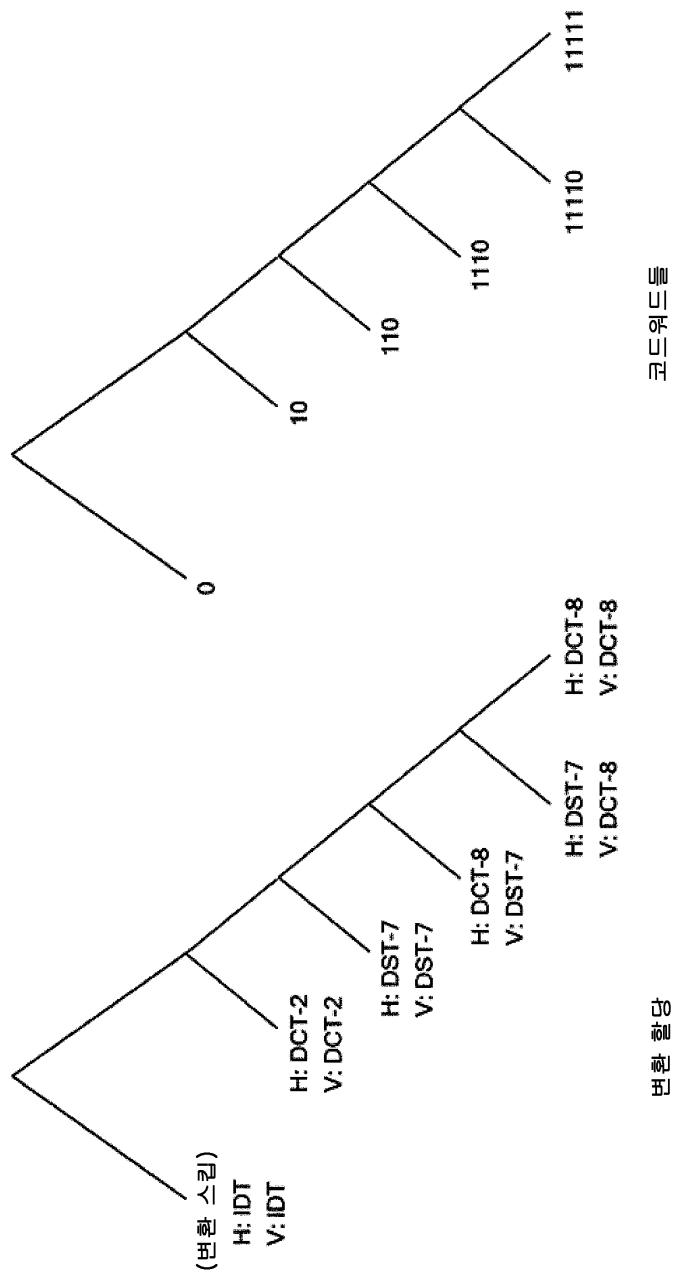
도면5b



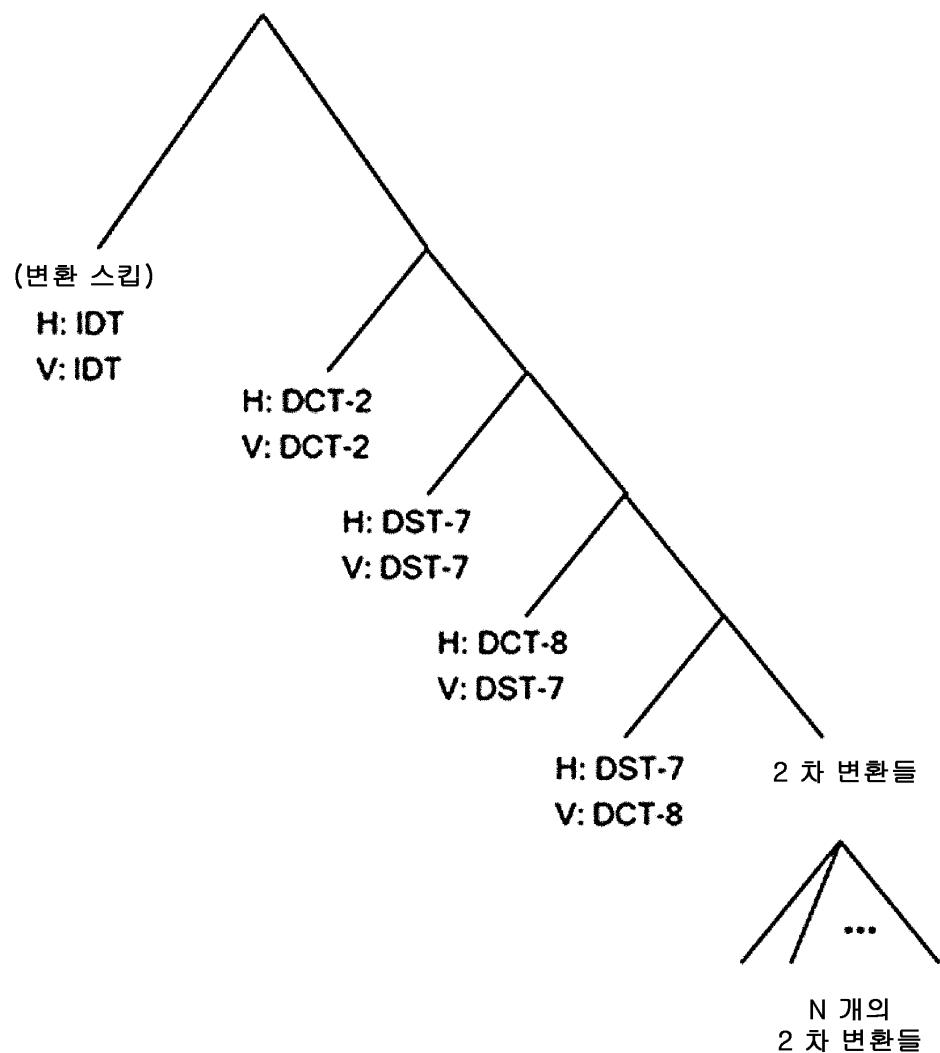
도면6



도면7



도면8



도면9

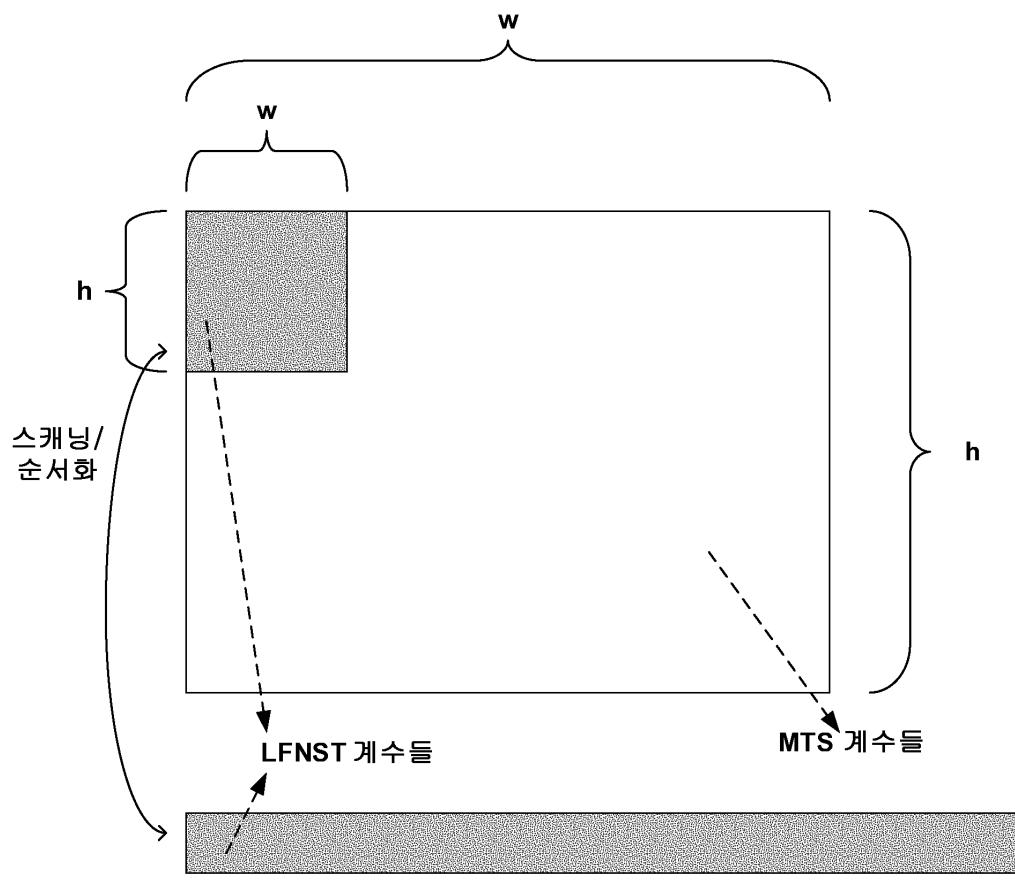


인코더 측

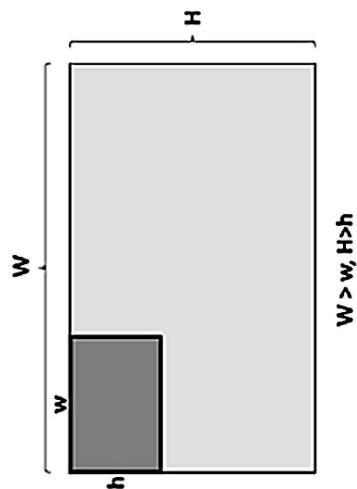
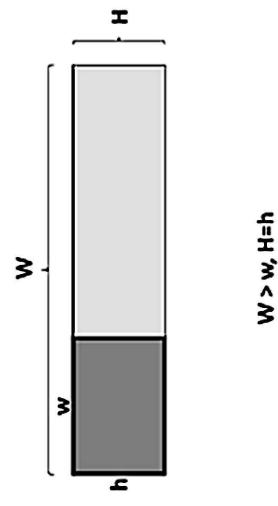
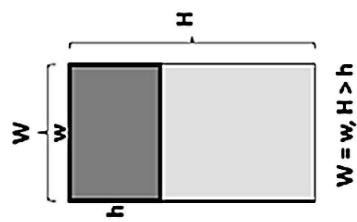


디코더 측

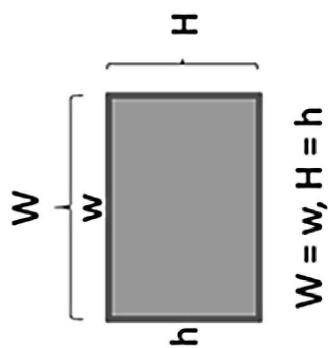
도면10



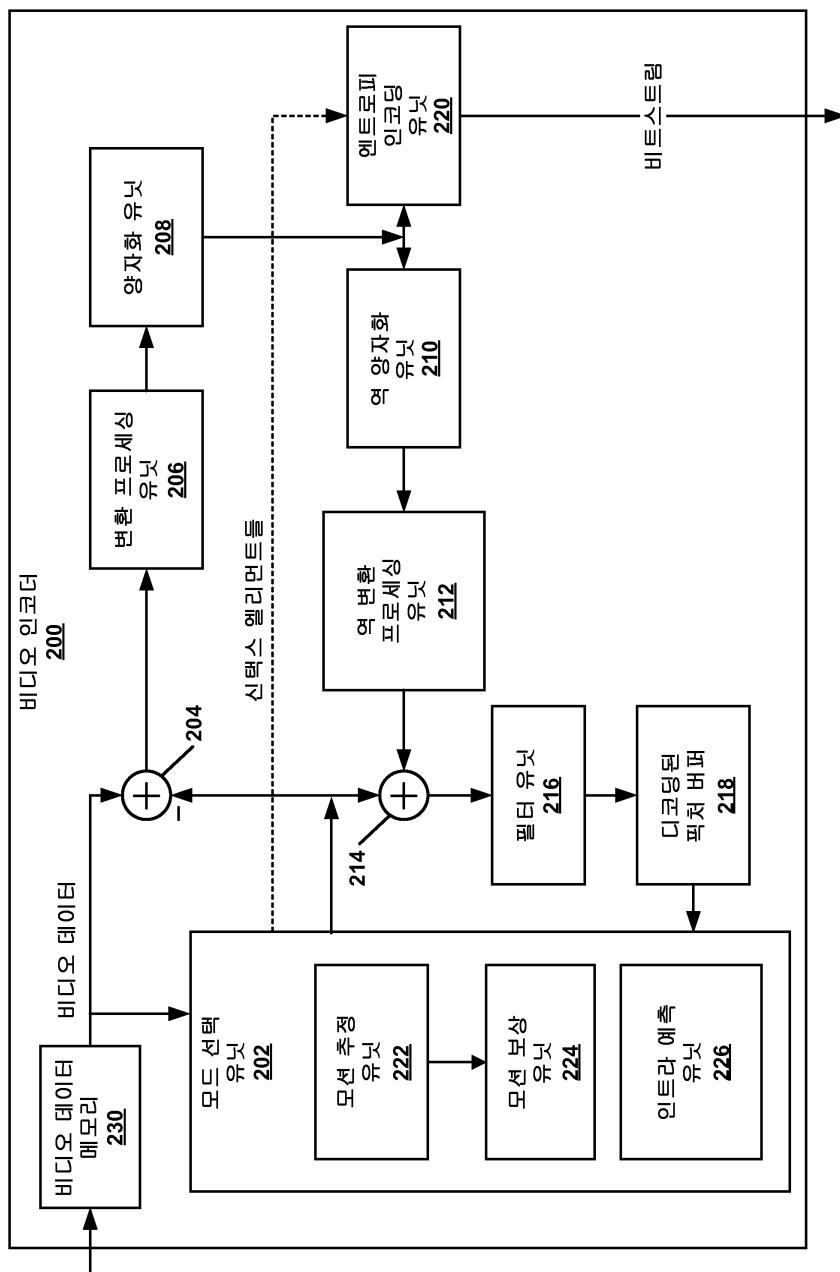
도면11a



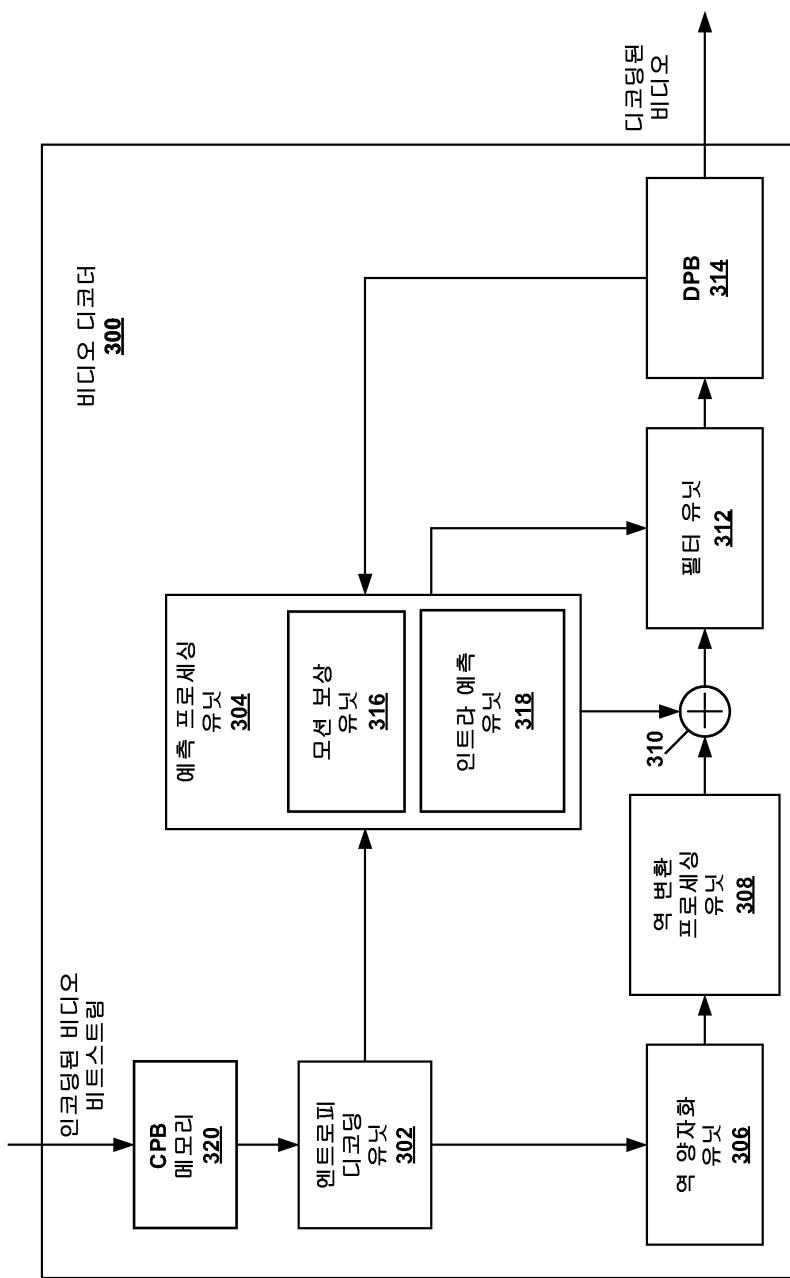
도면11b



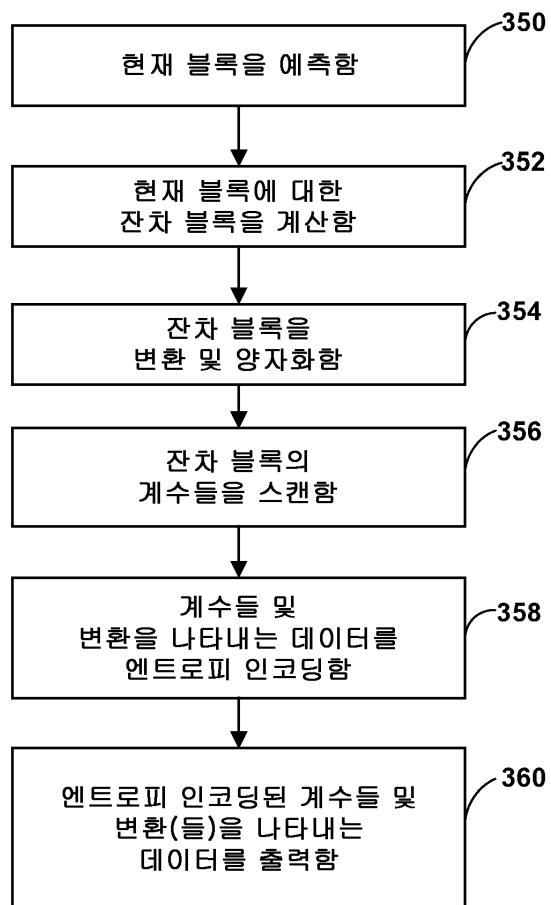
도면12



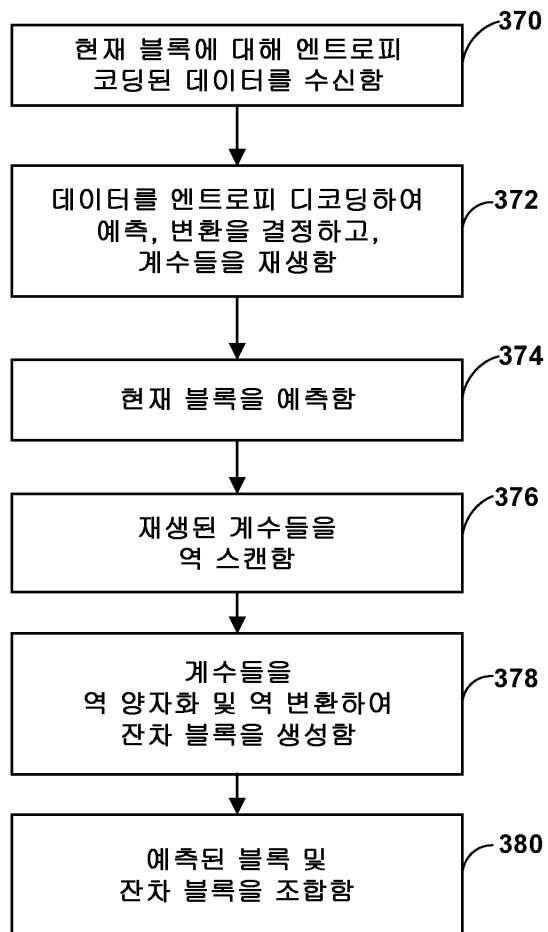
도면13



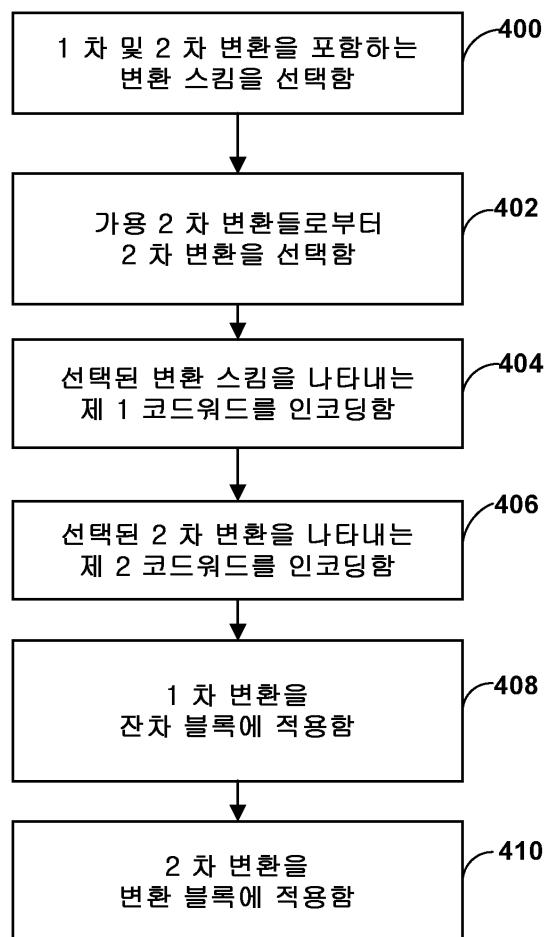
도면14



도면15



도면16



도면17

