



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2017년08월29일
(11) 등록번호 10-1772740
(24) 등록일자 2017년08월23일

- (51) 국제특허분류(Int. Cl.)
HO4N 19/13 (2014.01) *HO4N 19/60* (2014.01)
(21) 출원번호 10-2014-7022636
(22) 출원일자(국제) 2012년12월11일
 심사청구일자 2016년12월26일
(85) 번역문제출일자 2014년08월13일
(65) 공개번호 10-2014-0119736
(43) 공개일자 2014년10월10일
(86) 국제출원번호 PCT/US2012/068997
(87) 국제공개번호 WO 2013/109357
 국제공개일자 2013년07월25일
(30) 우선권주장
 13/710,212 2012년12월10일 미국(US)
 (뒷면에 계속)
(56) 선행기술조사문헌

J. Chen, et al. Non-CE1: throughput improvement on CABAC coefficients level coding. JCT-VC of ITU-T and ISO/IEC. JCTVC-H0554 Ver.2, Jan. 24, 2012, pp.1-6
B. Bross, et al. WD5: Working Draft 5 of High-Efficiency Video Coding. JCT-VC of ITU-T and ISO/IEC. JCTVC-G1103 Ver.1, Dec. 19, 2011, pp.1-214
N. Nguyen, et al. Adaptive Thresholds for Greater-Than-1 and Greater-Than-2 Flags. JCT-VC of ITU-T and ISO/IEC. JCTVC-I0281 Ver.4, Apr. 28, 2012, pp.1-15

KR1020090099236 A

전체 청구항 수 : 총 20 항

심사관 : 이상래

(54) 발명의 명칭 CABAC 계수 레벨 코딩에 대한 처리량 향상

(57) 요 약

본 개시물은 컨텍스트 적응 2진 산술 코딩 (CABAC) 과 함께 적응 컨텍스트 모델을 이용하여 코딩되는 빈들의 개수를 제한하는 여러 기법들을 제안한다. 특히, 본 개시물은 비디오 코딩 프로세스에서 변환 계수들의 레벨 정보를 코딩하기 위해 CABAC 을 이용하는 빈들의 개수를 제한하는 것을 제안한다.

대 표 도 - 도13



(72) 발명자

치엔 웨이-정

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

조쉬 라잔 랙스맨

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

(30) 우선권주장

61/587,624 2012년01월17일 미국(US)

61/589,290 2012년01월20일 미국(US)

61/591,772 2012년01월27일 미국(US)

61/606,347 2012년03월02일 미국(US)

61/622,785 2012년04월11일 미국(US)

명세서

청구범위

청구항 1

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 인코딩하는 방법으로서,

변환 계수들의 청크 (chunk)에서의 변환 계수들에 대해 유의도 맵 플래그 (significant map flag)를 인코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 인코딩하는 단계;

상기 청크에서 변환 계수들에 대한 부호 플래그를 인코딩하는 단계로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 인코딩하는 단계;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 1-보다-큰 플래그 (greater-than-one flag)를 인코딩하는 단계로서, 상기 1-보다-큰 플래그는 상기 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 인코딩하는 단계;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 변환 계수들에 대해 2-보다-큰 플래그 (greater-than-two flag)를 인코딩하는 단계로서, 상기 2-보다-큰 플래그는 상기 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 인코딩하는 단계;

상기 청크에서의 변환 계수들에 대해 레벨 잔존 값 (level remaining value)을 인코딩하는 단계; 및

인코딩된 비디오 비트스트림에서 상기 부호 플래그, 상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 상기 2-보다-큰 플래그 및 상기 레벨 잔존 값을 시그널링하는 단계를 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC)을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 1-보다-큰 플래그는 역 스캔 순서에서의 처음 M1 개의 비-제로 변환 계수들에 대해서만 코딩되고, 상기 M1은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는, 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1을 나타내는 것을 특징으로 하는 변환 계수들을 인코딩하는 방법.

청구항 2

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 디코딩하는 방법으로서,

인코딩된 비디오 비트스트림에서 부호 플래그, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그 및 레벨 잔존 값을 수신하는 단계;

변환 계수들의 청크에서의 변환 계수들에 대해 상기 유의도 맵 플래그를 디코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 디코딩하는 단계;

상기 청크에서 변환 계수들에 대한 상기 부호 플래그를 디코딩하는 단계로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 디코딩하는 단계;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변

환 계수들에 대해 상기 1-보다-큰 플래그를 디코딩하는 단계로서, 상기 1-보다-큰 플래그는 상기 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 디코딩하는 단계;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 변환 계수들에 대해 상기 2-보다-큰 플래그를 디코딩하는 단계로서, 상기 2-보다-큰 플래그는 상기 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 디코딩하는 단계;

상기 청크에서의 변환 계수들에 대해 상기 레벨 잔존 값을 디코딩하는 단계를 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 1-보다-큰 플래그는 역 스캔 순서에서의 처음 M1 개의 비-제로 변환 계수들에 대해서만 코딩되고, 상기 M1 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는, 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타내는 것을 특징으로 하는 변환 계수들을 디코딩하는 방법.

청구항 3

제 1 항에 있어서,

2-보다-큰 플래그는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 역 스캔 순서에서의 처음 N 개의 변환 계수들에 대해 코딩되며, 상기 2-보다-큰 플래그의 개수 (N) 는 상기 1-보다-큰 플래그의 개수 (M1) 이하이며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 1보다 큰 절대값을 갖는 것을 표시하는 코딩된 1-보다-큰 플래그를 갖지만 상기 2-보다-큰 플래그가 코딩되지 않은 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내는, 변환 계수들을 인코딩하는 방법.

청구항 4

제 2 항에 있어서,

2-보다-큰 플래그는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 역 스캔 순서에서의 처음 N 개의 변환 계수들에 대해 코딩되며, 상기 2-보다-큰 플래그의 개수 (N) 는 상기 1-보다-큰 플래그의 개수 (M1) 이하이며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 1보다 큰 절대값을 갖는 것을 표시하는 코딩된 1-보다-큰 플래그를 갖지만 상기 2-보다-큰 플래그가 코딩되지 않은 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내는, 변환 계수들을 디코딩하는 방법.

청구항 5

제 1 항에 있어서,

상기 유의도 맵 플래그를 첫번째로, 상기 1-보다-큰 플래그를 두번째로, 상기 2-보다-큰 플래그를 세번째로, 그리고 상기 레벨 잔존 값을 상기 2-보다-큰 플래그 이후에 인코딩하는 단계를 더 포함하는, 변환 계수들을 인코딩하는 방법.

청구항 6

제 2 항에 있어서,

상기 유의도 맵 플래그를 첫번째로, 상기 1-보다-큰 플래그를 두번째로, 상기 2-보다-큰 플래그를 세번째로, 그리고 상기 레벨 잔존 값을 상기 2-보다-큰 플래그 이후에 디코딩하는 단계를 더 포함하는, 변환 계수들을 디코딩하는 방법.

청구항 7

제 1 항에 있어서,

상기 청크는:

변환 계수들의 블록의 서브블록;

변환 계수들의 블록의 16 개의 변환 계수들의 서브블록;

변환 유닛; 및

변환 계수들의 블록 내의 역 스캔 순서를 따르는 복수의 연속하는 변환 계수들

중 하나인, 변환 계수들을 인코딩하는 방법.

청구항 8

제 2 항에 있어서,

상기 청크는:

변환 계수들의 블록의 서브블록;

변환 계수들의 블록의 16 개의 변환 계수들의 서브블록;

변환 유닛; 및

변환 계수들의 블록 내의 역 스캔 순서를 따르는 복수의 연속하는 변환 계수들

중 하나인, 변환 계수들을 디코딩하는 방법.

청구항 9

제 1 항에 있어서,

상기 M1 의 값은 8 인, 변환 계수들을 인코딩하는 방법.

청구항 10

제 2 항에 있어서,

상기 M1 의 값은 8 인, 변환 계수들을 디코딩하는 방법.

청구항 11

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 인코딩하는 방법으로서,

변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 인코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 인코딩하는 단계;

상기 청크에서 변환 계수들에 대한 부호 플래그를 인코딩하는 단계로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 인코딩하는 단계;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 인코딩하는 단계로서, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 인코딩하는 단계;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 2-보다-큰 플래그를 인코딩하는 단계로서, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 인코딩하는 단계;

상기 청크에서의 변환 계수들에 대해 레벨 잔존 값을 인코딩하는 단계; 및

인코딩된 비디오 비트스트림에서 상기 부호 플래그, 상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 상기 2-보다-큰 플래그 및 상기 레벨 잔존 값을 시그널링하는 단계를 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 2-보다-큰 플래그는 상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 역 스캔 순서에서의 처음 N 개의 변환 계수들에 대해 코딩되고, 상기 N 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 1보다 큰 절대값을 갖는 것을 표시하는 코딩된 1-보다-큰 플래그를 갖지만 상기 2-보다-큰 플래그가 코딩되지 않은 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내는 것을 특징으로 하는 변환 계수들을 인코딩하는 방법.

청구항 12

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 디코딩하는 방법으로서,

인코딩된 비디오 비트스트림에서 부호 플래그, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그 및 레벨 잔존 값을 수신하는 단계;

변환 계수들의 청크에서의 변환 계수들에 대해 상기 유의도 맵 플래그를 디코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 디코딩하는 단계;

상기 청크에서 변환 계수들에 대한 상기 부호 플래그를 디코딩하는 단계로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 디코딩하는 단계;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 변환 계수들의 청크에서의 변환 계수들에 대해 상기 1-보다-큰 플래그를 디코딩하는 단계로서, 상기 1-보다-큰 플래그는 상기 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 디코딩하는 단계;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 상기 2-보다-큰 플래그를 디코딩하는 단계로서, 상기 2-보다-큰 플래그는 상기 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 디코딩하는 단계; 및

상기 청크에서의 변환 계수들에 대해 상기 레벨 잔존 값을 디코딩하는 단계를 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 2-보다-큰 플래그는 상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 역 스캔 순서에서의 처음 N 개의 변환 계수들에 대해 코딩되고, 상기 N 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 1보다 큰 절대값을 갖는 것을 표시하는 코딩된 1-보다-큰 플래그를 갖지만 상기 2-보다-큰 플래그가 코딩되지 않은 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내는 것을 특징으로 하는 변환 계수들을 디코딩하는 방법.

청구항 13

제 11 항에 있어서,

상기 N 의 값은 1 인, 변환 계수들을 인코딩하는 방법.

청구항 14

제 12 항에 있어서,

상기 N 의 값은 1 인, 변환 계수들을 디코딩하는 방법.

청구항 15

명령들을 저장한 컴퓨터 판독가능 저장 매체로서,

상기 명령들은 실행될 때, 비디오 데이터를 코딩하도록 구성되는 디바이스의 하나 이상의 프로세서들로 하여금, 제 1 항, 제 3 항, 제 5 항, 제 7 항, 제 9 항, 제 11 항 또는 제 13 항 중 어느 한 항에 기재된 방법을 수행하게 하는, 명령들을 저장한 컴퓨터 판독가능 저장 매체.

청구항 16

명령들을 저장한 컴퓨터 판독가능 저장 매체로서,

상기 명령들은 실행될 때, 비디오 데이터를 코딩하도록 구성되는 디바이스의 하나 이상의 프로세서들로 하여금, 제 2 항, 제 4 항, 제 6 항, 제 8 항, 제 10 항, 제 12 항 또는 제 14 항 중 어느 한 항에 기재된 방법을 수행하게 하는, 명령들을 저장한 컴퓨터 판독가능 저장 매체.

청구항 17

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 인코딩하도록 구성되는 장치로서,

변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 인코딩하는 수단으로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 인코딩하는 수단;

상기 청크에서 변환 계수들에 대한 부호 플래그를 인코딩하는 수단으로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 인코딩하는 수단;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 1-보다-큰 플래그를 인코딩하는 수단으로서, 상기 1-보다-큰 플래그는 상기 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 인코딩하는 수단;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 인코딩하는 수단으로서, 상기 2-보다-큰 플래그는 상기 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 인코딩하는 수단;

상기 청크에서의 변환 계수들에 대해 레벨 잔존 값을 인코딩하는 수단; 및

인코딩된 비디오 비트스트림에서 상기 부호 플래그, 상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 상기 2-보다-큰 플래그 및 상기 레벨 잔존 값을 시그널링하는 수단을 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값을 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 1-보다-큰 플래그는 역 스캔 순서에서의 처음 M1 개의 비-제로 변환 계수들에 대해서만 코딩되고, 상기 M1 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값을, 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는, 상기 청크에서의 변환 계수들에 대

해 대응하는 계수의 절대값 마이너스 1 을 나타내는 것을 특징으로 하는 변환 계수들을 인코딩하도록 구성되는 장치.

청구항 18

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 인코딩하도록 구성되는 장치로서,

상기 장치는:

변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 인코딩하는 것으로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 인코딩하고;

상기 청크에서 변환 계수들에 대한 부호 플래그를 인코딩하는 것으로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 인코딩하고;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 인코딩하는 것으로서, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 인코딩하고;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 2-보다-큰 플래그를 인코딩하는 것으로서, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 인코딩하고; 그리고

상기 청크에서의 변환 계수들에 대해 레벨 잔존 값을 인코딩하고; 그리고

인코딩된 비디오 비트스트림에서 상기 부호 플래그, 상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 상기 2-보다-큰 플래그 및 상기 레벨 잔존 값을 시그널링하도록 구성되는 비디오 코더를 포함하고,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 2-보다-큰 플래그는 상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 역 스캔 순서에서의 처음 N 개의 변환 계수들에 대해 코딩되고, 상기 N 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 1보다 큰 절대값을 갖는 것을 표시하는 코딩된 1-보다-큰 플래그를 갖지만 상기 2-보다-큰 플래그가 코딩되지 않은 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내는 것을 특징으로 하는 변환 계수들을 인코딩하도록 구성되는 장치.

청구항 19

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 디코딩하도록 구성되는 장치로서,

인코딩된 비디오 비트스트림에서 부호 플래그, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그 및 레벨 잔존 값을 수신하는 수단;

변환 계수들의 청크에서의 변환 계수들에 대해 상기 유의도 맵 플래그를 디코딩하는 수단으로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 디코딩하는 수단;

상기 청크에서 변환 계수들에 대한 상기 부호 플래그를 디코딩하는 수단으로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 디코딩하는 수단;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 상기 1-보다-큰 플래그를 디코딩하는 수단으로서, 상기 1-보다-큰 플래그는 상기 특정의 변환

계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 디코딩하는 수단;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 변환 계수들에 대해 상기 2-보다-큰 플래그를 디코딩하는 수단으로서, 상기 2-보다-큰 플래그는 상기 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 디코딩하는 수단; 및

상기 청크에서의 변환 계수들에 대해 상기 레벨 잔존 값을 디코딩하는 수단을 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 1-보다-큰 플래그는 역 스캔 순서에서의 처음 M1 개의 비-제로 변환 계수들에 대해서만 코딩되고, 상기 M1 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는, 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타내는 것을 특징으로 하는 변환 계수들을 디코딩하도록 구성되는 장치.

청구항 20

비디오 코딩 프로세스에서 공간적 또는 시간적 예측으로부터의 잔차 데이터의 블록-기반 변환에 의해 생성된 변환 계수들을 디코딩하도록 구성되는 장치로서,

상기 장치는:

인코딩된 비디오 비트스트림에서 부호 플래그, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그 및 레벨 잔존 값을 수신하고;

변환 계수들의 청크에서의 변환 계수들에 대해 상기 유의도 맵 플래그를 디코딩하는 것으로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 유의도 맵 플래그를 디코딩하고;

상기 청크에서 변환 계수들에 대한 상기 부호 플래그를 디코딩하는 것으로서, 상기 부호 플래그는 상기 특정의 변환 계수의 부호를 표시하는, 상기 부호 플래그를 디코딩하고;

상기 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 상기 변환 계수들의 청크에서의 변환 계수들에 대해 상기 1-보다-큰 플래그를 디코딩하는 것으로서, 상기 1-보다-큰 플래그는 상기 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 1-보다-큰 플래그를 디코딩하고;

상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 상기 청크에서의 역 스캔 순서에서의 변환 계수들에 대해 상기 2-보다-큰 플래그를 디코딩하는 것으로서, 상기 2-보다-큰 플래그는 상기 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 2-보다-큰 플래그를 디코딩하고; 그리고

상기 청크에서의 변환 계수들에 대해 상기 레벨 잔존 값을 디코딩하도록 구성되는 비디오 코더를 포함하며,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 2보다 큰 절대값을 갖는다는 것을 표시하는 코딩된 2-보다-큰 플래그를 갖는 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내고,

상기 유의도 맵 플래그, 상기 1-보다-큰 플래그, 및 상기 2-보다-큰 플래그는, 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩되고,

상기 2-보다-큰 플래그는 상기 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 역 스캔 순서에서의 처음 N 개의 변환 계수들에 대해 코딩되고, 상기 N 은 상기 청크에서의 변환 계수들의 최대 개수 미만이고,

상기 레벨 잔존 값은, 상기 특정의 변환 계수가 1보다 큰 절대값을 갖는 것을 표시하는 코딩된 1-보다-큰 플래그를 갖지만 상기 2-보다-큰 플래그가 코딩되지 않은 상기 청크에서의 변환 계수들에 대해 대응하는 계수의 절

대값 마이너스 2 를 나타내는 것을 특징으로 하는 변환 계수들을 디코딩하도록 구성되는 장치.

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

발명의 설명

기술 분야

- [0001] 본 출원은 2012년 1월 17일자에 출원된 미국 가출원 번호 제 61/587,624호, 2012년 1월 20일자에 출원된 미국 가출원 번호 제 61/589,290호, 2012년 1월 27일자에 출원된 미국 가출원 번호 제 61/591,772호, 2012년 3월 2일자에 출원된 미국 가출원 번호 제 61/606,347호, 및 2012년 4월 11일자에 출원된 미국 가출원 번호 제 61/622,785호의 이익을 주장하며, 이의 모두는 그들 전체로 참조로 여기에 포함된다.
- [0002] 기술 분야
- [0003] 본 개시물은 비디오 코딩에 관한 것으로, 좀더 구체적으로는, 변환 계수들을 코딩하는 기법들에 관한 것이다.

배경 기술

- [0004] 디지털 비디오 능력들은 디지털 텔레비전, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 휴대정보 단말기들 (PDA들), 랩탑 또는 데스크탑 컴퓨터들, 디지털 카메라들, 디지털 리코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 원격 화상회의 디바이스들 등을 포함한, 광범위한 디바이스들에 포함될 수 있다. 디지털 비디오 디바이스들은 디지털 비디오 정보를 좀더 효율적으로 송수신하고 저장하기 위해, MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, AVC (Advanced Video Coding), 현재 개발중인 HEVC (High Efficiency Video Coding) 표준에 의해 정의된 표준들, 및 이런 표준들의 확장판들에 설명된 것들과 같은 비디오 압축 기법들을 구현한다.
- [0005] 비디오 압축 기법들은 비디오 시퀀스들에 고유한 리던던시를 감소시키거나 또는 제거하기 위해 공간 예측 및/또는 시간 예측을 포함한다. 블록-기반의 비디오 코딩에 있어, 비디오 프레임 또는 슬라이스는 블록들로 파티셔닝될 수도 있다. 비디오 프레임은 대안적으로 화상으로서 지칭될 수도 있다. 각각의 블록은 더 파티셔닝될 수도 있다. 인트라-코딩된 (I) 프레임 또는 슬라이스에서의 블록들은 동일한 프레임 또는 슬라이스에서 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 인터-코딩된 (P 또는 B) 프레임 또는 슬라이스에서의 블록들은 동일한 프레임 또는 슬라이스에서 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측, 또는 다른 참조 프레임들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 공간 또는 시간 예측은 코딩되는 블록에 대한 예측 블록을 초래한다. 잔여 데이터는 코딩되는 원래 블록, 즉, 코딩되는 블록과, 예측 블록 사이의 픽셀 차이들을 나타낸다.

- [0006] 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 가리키는 모션 벡터, 및 코딩된 블록과 예측 블록 사이의 차이를 나타내는 잔여 데이터에 따라서 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔여 데이터에 따라서 인코딩된다. 추가적인 압축을 위해, 잔여 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔여 변환 계수들을 초래할 수도 있으며, 이 잔여 변환 계수는 그후 양자화될 수도 있다. 처음에 2차원 어레이로 배열된, 양자화된 변환 계수들은 엔트로피 코딩을 위한 변환 계수들의 1차원 벡터를 발생하기 위해 특정의 순서로 스캔될 수도 있다.

발명의 내용

과제의 해결 수단

- [0007] 일반적으로, 본 개시물은 비디오 데이터를 코딩하는 기법들을 설명한다. 특히, 본 개시물은 비디오 코딩 프로세스에서 변환 계수들에 관련된 정보를 코딩하는 기법들을 기술한다.
- [0008] 본 개시물은 변환 계수들 대 바이패스 코딩된 빈들의 개수를 시그널링하기 위해 컨텍스트 적응 2진 산술 코딩 (context adaptive binary arithmetic coding; CABAC) 과 함께 적응 컨텍스트 모델을 이용하여 코딩되는 빈들의 개수를 제한하는 여러 기법들을 제안한다. 특히, 본 개시물은 비디오 코딩 프로세스에서 변환 계수들의 레벨 정보를 코딩하기 위해 CABAC 을 이용하는 빈들의 개수를 제한하는 기법들을 기술한다.
- [0009] 본 개시물의 일 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법은, 변환 계수들의 청크 (chunk) 에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그 (greater-

than-one flag) 를 코딩하는 단계로서, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계; 및 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그 (greater-than-two flag) 를 코딩하는 단계로서, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계를 포함한다.

[0010] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법은, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 단계로서, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계; 및 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하는 단계로서, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계를 포함한다.

[0011] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법은 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 단계로서, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 단계로서, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계; 및 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하는 단계로서, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시하는, 상기 코딩하는 단계를 포함한다.

[0012] 본 개시물은 또한 비디오 인코더, 비디오 디코더, 장치들, 및 본원에서 설명되는 변환 계수들을 시그널링하는 기법들을 수행하도록 구성될 수도 있는 명령들을 저장하는 컴퓨터-판독가능 매체들을 기술한다.

[0013] 하나 이상의 예들의 세부 사항들이 첨부도면 및 아래의 상세한 설명에서 개시된다. 다른 특성들, 목적들, 및 이점들은 설명 및 도면들로부터, 그리고 청구항들로부터 명백히 알 수 있을 것이다.

도면의 간단한 설명

[0014] 도 1 은 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2 는 변환 계수 코딩을 위한 예시적인 역 스캔 순서들을 나타내는 개념 도면이다.

도 3 은 예시적인 서브블록-기반의 대각선 스캔을 나타내는 개념 도면이다.

도 4 는 변환 계수들의 4x4 청크의 예시적인 역 대각선 스캔을 나타내는 개념 도면이다.

도 5 는 서브블록 위치에 기초한 컨텍스트-기반의 코딩 임계치들을 예시하는 개념도이다.

도 6 은 서브블록 위치 및 최종 유의 계수 (last significant coefficient) 의 위치에 기초한 컨텍스트-기반의 코딩 임계치들을 예시하는 개념도이다.

도 7 은 서브블록 위치 및 최종 유의 계수의 위치에 기초한 컨텍스트-기반의 코딩 임계치들을 예시하는 개념도이다.

도 8 은 4개의 4x4 서브-블록들로의 8x8 블록의 예시적인 분할 및 컨텍스트 이웃을 나타내는 개념도이다.

도 9 는 예시적인 비디오 인코더를 예시하는 블록도이다.

도 10 은 예시적인 비디오 디코더를 예시하는 블록도이다.

도 11 은 본 개시물의 기법들에 따른 예시적인 방법을 나타내는 플로우차트이다.

도 12 는 본 개시물의 기법들에 따른 또 다른 예시적인 방법을 나타내는 플로우차트이다.

도 13 은 본 개시물의 기법들에 따른 또 다른 예시적인 방법을 나타내는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

- [0015] 일반적으로, 본 개시물은 비디오 데이터를 코딩하는 기법들을 설명한다. 특히, 본 개시물은 비디오 인코딩 및/또는 디코딩 프로세스에서 변환 계수들을 코딩하는 기법들을 설명한다.
- [0016] 변환 계수들에 대한 레벨 정보 (예컨대, 절대값 및 부호)는 일반적으로 2 가지 방법들 중 하나로 엔트로피 코딩된다. 레벨 정보의 일부 빈들은 (예컨대, 컨텍스트 적응 2진 산술 코딩 (CABAC) 과 함께) 적응 컨텍스트 모델을 이용하여 코딩된다. 레벨 정보의 다른 빈들은 (예컨대, 지수 Golomb 코더 또는 Golomb-Rice 코더를 이용하여) 고정 동등 확률 모델들 (fixed equal probability models) 과 함께 바이패스 모드를 통해서 코딩된다. 적응 컨텍스트-기반의 빈 코딩은, 대역폭 효율을 증진하지만, 엔트로피 코딩의 주요 병목들 중 하나인 것으로 관찰되었다.
- [0017] 이 단점의 관점에서, 본 개시물은 CABAC 처리량을 향상시키는 기법들을 제시한다. 특히, 일부 예들에서, 본 개시물은 적응 컨텍스트-코딩된 빈들의 개수를 감소시키고 바이패스-코딩된 빈들의 개수를 증가시킴으로써, CABAC 처리량을 향상시키는 기법들을 제안한다.
- [0018] 도 1은 본 개시물의 예들에 따른, 변환 계수들을 코딩하는 기법들을 이용하도록 구성될 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10)을 예시하는 블록도이다. 도 1에 나타낸 바와 같이, 시스템 (10)은 인코딩된 비디오를 통신 채널 (16)을 통해서 목적지 디바이스 (14)로 송신하는 소스 디바이스 (12)를 포함한다. 인코딩된 비디오 데이터는 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있으며, 원하는 바에 따라 목적지 디바이스 (14)에 의해 액세스될 수도 있다. 저장 매체 또는 파일 서버에 저장될 때, 비디오 인코더 (20)는 코딩된 비디오 데이터를 저장 매체에 저장하기 위해, 코딩된 비디오 데이터를 네트워크 인터페이스, 컴팩트 디스크 (CD), 블루-레이 또는 디지털 비디오 디스크 (DVD) 베너 또는 스템핑 기능 디바이스, 또는 다른 디바이스들과 같은, 또 다른 디바이스에 제공할 수도 있다. 이와 유사하게, 네트워크 인터페이스, CD 또는 DVD 리더 등과 같은, 비디오 디코더 (30)와는 별개인 디바이스가 저장 매체로부터 코딩된 비디오 데이터를 취출하고, 그 취출된 데이터를 비디오 디코더 (30)에 제공할 수도 있다.
- [0019] 소스 디바이스 (12) 및 목적지 디바이스 (14)는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋-탑 박스들, 전화기 핸드셋들, 예컨대 소위 스마트폰들, 텔레비전, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들 등을 포함한, 매우 다양한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 많은 경우에, 이런 디바이스들은 무선 통신용으로 탑재될 수도 있다. 그러므로, 통신 채널 (16)은 인코딩된 비디오 데이터의 송신에 적합한, 무선 채널, 유선 채널, 또는 무선 채널과 유선 채널의 조합을 포함할 수도 있다. 이와 유사하게, 파일 서버 (36)는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서 목적지 디바이스 (14)에 의해 액세스될 수도 있다. 이것은 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양쪽의 조합을 포함할 수도 있다.
- [0020] 본 개시물의 예들에 따른, 변환 계수들을 코딩하는 기법들은, 오버-디-에어 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예컨대, 인터넷을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상에의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은, 다양한 멀티미디어 애플리케이션들 중 임의의 애플리케이션의 지원 하에서, 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 시스템 (10)은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화 통신과 같은, 지원 애플리케이션들로의 1-방향 또는 2-방향 비디오 송신을 지원하도록 구성될 수도 있다.
- [0021] 도 1의 예에서, 소스 디바이스 (12)는 비디오 소스 (18), 비디오 인코더 (20), 변조기/복조기 (모뎀; 22) 및 송신기 (24)를 포함한다. 소스 디바이스 (12)에서, 비디오 소스 (18)는 비디오 캡처 디바이스, 예컨대 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 공급 인터페이스, 및/또는 컴퓨터 그래픽스 데이터를 소스 비디오로서 발생하는 컴퓨터 그래픽스 시스템과 같은 소스, 또는 이런 소스들의 조합을 포함할 수도 있다. 일 예로서, 비디오 소스 (18)가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14)는 예컨대, 스마트폰들 또는 태블릿 컴퓨터들 내에 제공될 수도 있는, 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 그러나, 본 개시물에서 설명하는 기법들은 일반적으로는 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션들, 또는 인코딩된 비디오 데이터가 로컬 디스크 상에 저장되는 애플리케이션에 적용될 수도 있다.
- [0022] 캡처되거나, 사전-캡처되거나, 또는 컴퓨터-발생된 비디오는 비디오 인코더 (20)에 의해 인코딩될 수도 있다. 인코딩된 비디오 정보는 유선 또는 무선 통신 프로토콜과 같은 통신 표준에 따라서 모뎀 (22)에 의해 변조

되어, 송신기 (24) 를 통해서 목적지 디바이스 (14) 로 송신될 수도 있다. 모뎀 (22) 은 여러 맵서들, 필터들, 증폭기들 또는 신호 변조용으로 설계된 다른 구성요소들을 포함할 수도 있다. 송신기 (24) 는 증폭기들, 필터들, 및, 무선 통신의 경우, 하나 이상의 안테나들을 포함한, 데이터를 송신할 목적으로 설계된 회로들을 포함할 수도 있다.

[0023] 비디오 인코더 (20) 에 의해 인코딩된, 캡처된, 사전-캡처된, 또는 컴퓨터-발생된 비디오는 또한 추후 소비를 위해 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있다. 저장 매체 (34) 는 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 또는 인코딩된 비디오를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들을 포함할 수도 있다. 저장 매체 (34) 상에 저장된 인코딩된 비디오는 그후 디코딩 및 플레이백을 위해 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 도 1 에 나타내지는 않지만, 일부 예들에서, 저장 매체 (34) 및/또는 파일 서버 (36) 는 송신기 (24) 의 출력을 저장할 수도 있다.

[0024] 파일 서버 (36) 는 인코딩된 비디오를 저장하고 그 인코딩된 비디오를 목적지 디바이스 (14) 로 송신하는 것이 가능한 임의 종류의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예컨대, 웹사이트용), FTP 서버, NAS (network attached storage) 디바이스들, 로컬 디스크 드라이브, 또는 인코딩된 비디오 데이터를 저장하고 이를 목적지 디바이스로 송신할 수 있는 임의의 다른 유형의 디바이스를 포함한다. 파일 서버 (36) 로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이 양쪽의 조합일 수도 있다. 파일 서버 (36) 는 목적지 디바이스 (14) 에 의해, 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서 액세스될 수도 있다. 이것은 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀, 이더넷, USB 등), 또는 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한 이들 양쪽의 조합을 포함할 수도 있다.

[0025] 목적지 디바이스 (14) 는, 도 1 의 예에서, 수신기 (26), 모뎀 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 목적지 디바이스 (14) 의 수신기 (26) 는 채널 (16) 을 통해서 정보를 수신하고, 모뎀 (28) 은 그 정보를 복조하여, 비디오 디코더 (30) 에 대한 복조된 비트스트림을 발생한다. 채널 (16) 을 통해서 통신된 정보는 비디오 데이터를 디코딩할 때에 비디오 디코더 (30) 에 의한 사용을 위해서 비디오 인코더 (20) 에 의해 발생되는 다양한 구문 정보를 포함할 수도 있다. 이런 구문은 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장된 인코딩된 비디오 데이터와 함께 포함될 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 비디오 데이터를 인코딩 또는 디코딩할 수 있는 각각의 인코더-디코더 (코덱) 의 일부를 형성할 수도 있다.

[0026] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합되거나 또는 그 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함하며, 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 그 디코딩된 비디오 데이터를 사용자에게 디스플레이하며, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 또 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0027] 도 1 의 예에서, 통신 채널 (16) 은 임의의 무선 또는 유선 통신 매체, 예컨대 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적인 송신 라인들, 또는 무선 매체와 유선 매체들의 임의의 조합을 포함할 수도 있다. 통신 채널 (16) 은 근거리 네트워크, 광역 네트워크, 또는 글로벌 네트워크, 예컨대 인터넷과 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 채널 (16) 은 일반적으로 유선 또는 무선 매체들의 임의의 적합한 조합을 포함한, 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 송신하기 위한, 임의의 적합한 통신 매체, 또는 상이한 통신 매체들의 결합을 나타낸다. 통신 채널 (16) 은 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0028] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 ITU-T 비디오 코딩 전문가 그룹 (VCEG) 과 ISO/IEC 동화상 전문가 그룹 (MPEG) 의 비디오 코딩에 관한 합동 작업팀 (JCT-VC) 에 의해 현재 개발중인 HEVC (High Efficiency Video Coding) 표준과 같은 비디오 압축 표준에 따라서 동작할 수도 있다. HEVC 의 가장 최근의 작업 초안 (WD) 이 ITU-T SG16 WP3 와 ISO/IEC JTC1/SC29/WG11 의 JCT-VC (Joint Collaborative Team on Video Coding), 문서 JCTVC-I1003, Bross 등, "High efficiency video coding (HEVC) text specification draft 9" 에 설명되어 있으며, 이하에 HEVC WD7 로서 지칭되며, http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v6.zip 으로부터 입수 가능하다.

- [0029] 이의 대안으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG 4, 파트 10, AVC (Advanced Video Coding) 로서 대안적으로 지정되는 ITU-T H.264 표준, 또는 이런 표준들의 확장판들과 같은 다른 사유 (proprietary) 또는 산업 표준들에 따라서 동작할 수도 있다. 본 개시물의 기법들은, 그러나, 임의의 특정의 코딩 표준에 한정되지 않는다. 다른 예들은 MPEG-2 및 ITU-T H.263 를 포함한다.
- [0030] 도 1 에 나타내지는 않지만, 일부 양태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 공통 데이터 스트림 또는 별개의 데이터 스트림들에서의 오디오와 비디오 양쪽의 인코딩을 처리하는 데 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대, 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.
- [0031] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적회로들 (ASIC들), 펠드 프로그래밍가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 임의의 이들의 조합들과 같은, 다양한 적합한 인코더 회로 중 임의의 회로로 구현될 수도 있다. 이 기법들이 소프트웨어로 부분적으로 구현되는 경우, 디바이스는 본 개시물의 기법들을 수행하기 위해 소프트웨어용 명령들을 적합한 비일시성 컴퓨터-관독가능 매체에 저장하고, 그 명령들을 하드웨어에서 하나 이상의 프로세서들을 이용하여 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이를 중 어느 쪽이든 각각의 디바이스에서 결합된 인코더/디코더 (코덱) 의 일부로서 통합될 수도 있다.
- [0032] 비디오 인코더 (20) 는 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 본 개시물의 기법들 중 임의의 기법 또는 모두를 구현할 수도 있다. 이와 유사하게, 비디오 디코더 (30) 는 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 이들 기법들 중 임의의 기법 또는 모두를 구현할 수도 있다. 비디오 코더는, 본 개시물에서 설명하는 바와 같이, 비디오 인코더 또는 비디오 디코더를 지칭할 수도 있다. 이와 유사하게, 비디오 코딩 유닛은 비디오 인코더 또는 비디오 디코더를 지칭할 수도 있다. 이와 유사하게, 비디오 코딩은 비디오 인코딩 또는 비디오 디코딩을 지칭할 수도 있다.
- [0033] 본 개시물의 일 예에서, 비디오 인코더 (20) 는 변환 계수들의 청크에서의 각각의 변환 계수에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시한다.
- [0034] 이와 유사하게, 비디오 디코더 (30) 는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그들을 디코딩하고; 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 디코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 디코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시한다.
- [0035] 본 개시물의 또 다른 예에서, 비디오 인코더 (20) 는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.
- [0036] 이와 유사하게, 비디오 디코더 (30) 는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 디코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 디코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 디코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변

환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.

[0037] 본 개시물의 또 다른 예에서, 비디오 인코더 (20)는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.

[0038] 이와 유사하게, 비디오 디코더 (30)는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그들을 디코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 디코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 디코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.

[0039] 디지털 비디오 디바이스들은 디지털 비디오 정보를 좀더 효율적으로 인코딩하고 디코딩하기 위해 비디오 압축 기법들을 구현한다. 비디오 압축은 비디오 시퀀스들에 고유한 리턴던시를 감소시키거나 또는 제거하기 위해 공간 (인트라-프레임) 예측 및/또는 시간 (인터-프레임) 예측 기법들을 적용할 수도 있다.

[0040] JCT-VC 는 예컨대, 위에서 설명한 HEVC WD7 에 기술되어 있는 바와 같은, HEVC 표준의 발달에 노력을 들이고 있다. HEVC 표준화 노력들은 HEVC 테스트 모델 (HM) 로서 지칭되는 비디오 코딩 디바이스의 진화 모델 (evolving model) 에 기초한다. HM은 예컨대, ITU-T H.264/AVC 에 따른 기준 디바이스들에 관련된 비디오 코딩 디바이스들의 여러 추가적인 능력들을 가정한다. 예를 들어, H.264 는 9개의 인트라-예측 인코딩 모드들을 제공하는 반면, HM은 33개 만큼이나 많은 인트라-예측 인코딩 모드들을 제공할 수도 있다. 다음 섹션은 HM의 어떤 양태들을 좀더 자세하게 설명한다.

[0041] 현재 개발중인 HEVC 표준에 따른 비디오 코딩에 있어, 비디오 프레임은 코딩 유닛들로 파티셔닝될 수도 있다. 코딩 유닛 (CU)은 일반적으로 비디오 압축을 위해 여러 코딩 툴들이 적용되는 기본 유닛으로서 역할을 하는 이미지 영역을 지칭한다. CU는 대개 Y로 표기되는 흑도 성분, 및 U와 V로 표기되는 2개의 크로마 성분들을 갖는다. 비디오 샘플링 포맷에 따라서, U 및 V 성분들의 사이즈는, 샘플들의 개수의 관점에서, Y 성분의 사이즈와 동일하거나 또는 상이할 수도 있다.

[0042] CU는 일반적으로 정사각형이며, 예컨대, ITU-T H.264 와 같은, 다른 비디오 코딩 표준들 하에서 소위 매크로블록과 유사한 것으로 간주될 수도 있다. 개발중인 HEVC 표준의 현재 제안된 양태들의 일부에 따른 코딩이 본 출원 예시의 목적을 위해 설명될 것이다. 그러나, 본 개시물에서 설명하는 기법들은 H.264 또는 다른 표준 또는 사유 비디오 코딩 프로세스들에 따라 정의된 것들과 같은, 다른 비디오 코딩 프로세스들에 유용할 수도 있다.

[0043] HM에 따르면, CU는 하나 이상의 예측 유닛들 (PU들) 및/또는 하나 이상의 변환 유닛들 (TU들)을 포함할 수도 있다. 비트스트림 내 구문 데이터는 픽셀들의 개수의 관점에서 최대 CU인 최대 코딩 유닛 (LCU)을 정의한다. 일반적으로, CU가 사이즈 구별을 갖지 않는다는 점을 제외하고는, CU는 H.264의 매크로블록과 유사한 목적을 갖는다. 따라서, CU는 서브-CU들로 분할될 수도 있다. 일반적으로, 본 개시물에서 CU에 대한 언급들은 화상의 최대 코딩 유닛 또는 LCU의 서브-CU를 지칭할 수도 있다. LCU는 서브-CU들로 분할될 수도 있으며, 각각의 서브-CU는 서브-CU들로 추가로 분할될 수도 있다. 비트스트림에 대한 구문 데이터는 LCU가 분할되는 최대 횟수를 정의할 수도 있으며, 이는 CU 깊이로 지칭된다. 따라서, 비트스트림은 또한 최소 코딩 유닛 (SCU)을 정의할 수도 있다. 본 개시물은 또한 CU, PU, 또는 TU 중 임의의 것을 지칭하기 위해 용어 "블록" 또는 "부분"을 사용한다. 일반적으로, "부분"은 비디오 프레임의 임의의 서브-세트를 지칭할 수도 있다.

[0044] LCU는 퀘드트리 데이터 구조와 연관될 수도 있다. 일반적으로, 퀘드트리 데이터 구조는 CU 당 하나의 노드를 포함하며, 여기서, 루트 노드는 LCU에 대응한다. CU가 4개의 서브-CU들로 분할되면, CU에 대응하는

노드는 4개의 잎 노드들을 포함하며, 그 잎 노드 각각은 서브-CU들 중 하나에 대응한다. 큐드트리 데이터 구조의 각각의 노드는 대응하는 CU에 대해 구문 데이터를 제공할 수도 있다. 예를 들어, 큐드트리에서 노드는 그 노드에 대응하는 CU가 서브-CU들로 분할되는지 여부를 표시하는 분할 플래그를 포함할 수도 있다. CU에 대한 구문 엘리먼트들은 회귀적으로 정의될 수도 있으며, CU가 서브-CU들로 분할되는지 여부에 의존할 수도 있다. CU가 추가로 분할되지 않으면, 잎-CU로서 지칭된다. 본 개시물에서, 잎-CU의 4개의 서브-CU들은 또한 원래 잎-CU의 명시적인 분할이 존재하지 않더라도, 잎-CU들로서 지칭될 것이다. 예를 들어, 16x16 사이즈에서 CU가 추가로 분할되지 않으면, 4개의 8x8 서브-CU들이 또한, 16x16 CU가 전혀 분할되지 않았더라도, 잎-CU들로서 지칭될 것이다.

[0045] 잎-CU는 하나 이상의 예측 유닛들 (PU들)을 포함할 수도 있다. 일반적으로, PU는 대응하는 CU의 모두 또는 부분을 나타내며, PU에 대한 참조 샘플을 취출하기 위한 데이터를 포함할 수도 있다. 예를 들어, PU가 인터-모드 인코딩될 때, PU는 그 PU에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 성분, 모션 벡터의 수직 성분, 모션 벡터에 대한 해상도 (예컨대, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 가리키는 참조 프레임, 및/또는 모션 벡터에 대한 참조 리스트 (예컨대, 리스트 0 또는 리스트 1)를 기술할 수도 있다. PU(들)을 정의하는 잎-CU에 대한 데이터는 또한 예를 들어, 하나 이상의 PU들로의 CU의 파티셔닝을 기술할 수도 있다. 파티셔닝 모드들은 CU가 인코딩되지 않거나, 인트라-예측 모드 인코딩되거나, 또는 인터-예측 모드 인코딩되는지 여부에 따라서 상이할 수도 있다. 인트라 코딩에 있어, PU는 아래에서 설명되는 잎 변환 유닛과 동일하게 처리될 수도 있다.

[0046] 차기 HEVC 표준은 변환 유닛들 (TU들)에 따른 변환들을 가능하게 하며, 이 TU들은 상이한 CU들에 대해 상이할 수도 있다. TU들은 일반적으로 파티셔닝된 LCU에 대해 정의된 주어진 CU 내 PU들의 사이즈에 기초하여 사이징되지만, 이것이 항상 그런 것은 아니다. TU들은 일반적으로 PU들과 동일한 사이즈이거나 또는 그보다 작다. 일부 예들에서, CU에 대응하는 잔여 샘플들은 "잔여 큐드 트리" (RQT)로서 알려진 큐드트리 구조를 이용하여 더 작은 유닛들로 세분될 수도 있다. RQT의 잎 노드들은 변환 유닛들 (TU들)로서 지칭될 수도 있다. TU들과 연관되는 픽셀 차이 값들은 변환 계수들을 발생하기 위해 변환될 수도 있으며, 그 변환 계수들은 양자화될 수도 있다.

[0047] 일반적으로, PU는 예측 프로세스에 관련된 데이터를 지칭한다. 예를 들어, PU가 인트라-모드 인코딩될 때, PU는 그 PU에 대한 인트라-예측 모드를 기술하는 데이터를 포함할 수도 있다. 또 다른 예로서, PU가 인터-모드 인코딩될 때, PU는 그 PU에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다.

[0048] 일반적으로, TU는 변환 및 양자화 프로세스들에 사용된다. 하나 이상의 PU들을 갖는 주어진 CU는 또한 하나 이상의 변환 유닛들 (TU들)을 포함할 수도 있다. 예측 이후, 비디오 인코더 (20)는 PU에 따라서 코딩 노드에 의해 식별되는 비디오 블록으로부터 잔여 값들을 계산할 수도 있다. 코딩 노드는 그후 원래 비디오 블록 대신, 잔여 값들을 참조하기 위해 업데이트된다. 잔여 값들은 엔트로피 코딩을 위한 직렬화된 변환 계수들을 발생하기 위해 TU들에 규정된 변환들 및 다른 변환 정보를 이용하여, 변환 계수들로 변환되고, 양자화되고, 그리고 스캔될 수도 있는 픽셀 차이 값들을 포함한다. 코딩 노드는 또 다시 이를 직렬화된 변환 계수들을 참조하기 위해 업데이트될 수도 있다. 본 개시물은 일반적으로 CU의 코딩 노드를 지칭하기 위해 용어 "비디오 블록"을 사용한다. 일부 특정의 경우들에서, 본 개시물은 또한 트리블록, 즉, LCU, 또는 코딩 노드 및 PU들 및 TU들을 포함하는 CU를 지칭하기 위해 용어 "비디오 블록"을 이용할 수도 있다.

[0049] 비디오 시퀀스는 일반적으로 비디오 프레임들 또는 화상들의 시리즈를 포함한다. 화상들의 그룹 (GOP)은 일반적으로 비디오 화상들의 하나 이상의 시리즈를 포함한다. GOP는 GOP의 헤더, 화상들의 하나 이상의 헤더, 또는 다른 곳에, GOP에 포함된 다수의 화상들을 기술하는 구문 데이터를 포함할 수도 있다. 화상의 각각의 슬라이스는 각각의 슬라이스에 대한 인코딩 모드를 기술하는 슬라이스 구문 데이터를 포함할 수도 있다.

비디오 인코더 (20)는 일반적으로 비디오 데이터를 인코딩하기 위해 개개의 비디오 슬라이스들 내 비디오 블록들에 대해 동작한다. 비디오 블록은 CU 내 코딩 노드에 대응할 수도 있다. 비디오 블록들은 고정 또는 가변 사이즈들을 가질 수도 있으며, 규정된 코딩 표준에 따라서 사이즈가 상이할 수도 있다.

[0050] 블록 (예컨대, 비디오 데이터의 예측 유닛)을 코딩하기 위해, 그 블록에 대한 예측자 (predictor)가 먼저 유도된다. 예측 블록으로 또한 지칭되는 예측자는 인트라 (I) 예측 (즉, 공간 예측) 또는 인터 (P 또는 B) 예측 (즉, 시간 예측)을 통해서 유도될 수 있다. 그러므로, 일부 예측 유닛들은 동일한 프레임 (또는, 슬라이스)에서 이웃하는 참조 블록들에서의 참조 샘플들에 대해 공간 예측을 이용하여 인트라-코딩될 수도 있으며

(I), 다른 예측 유닛들은 다른 이전에-코딩된 프레임들 (또는, 슬라이스들) 에서의 참조 샘플들의 블록들에 대해 단방향 인터-코딩되거나 (P) 또는 양방향 인터-코딩될 수도 있다 (B). 각 경우에, 참조 샘플들은 코딩되는 블록에 대한 예측 블록을 형성하는데 사용될 수도 있다.

[0051] 예측 블록의 식별 시, 원래 비디오 데이터 블록에서의 픽셀들과 그의 예측 블록에서의 픽셀들 사이의 차이가 결정된다. 이 차이는 예측 잔여 데이터로서 지칭될 수도 있으며, 코딩되는 블록에서의 픽셀 값들과, 코딩된 블록을 나타내기 위해 선택된 예측 블록에서의 픽셀 값들 사이의 픽셀 차이들을 나타낸다. 더 나은 압축을 달성하기 위해, 예측 잔여 데이터는 변환 계수들을 발생하기 위해 예컨대, 이산 코사인 변환 (DCT), 정수 변환, Karhunen-Loeve (K-L) 변환, 또는 또 다른 변환을 이용하여 변환될 수도 있다.

[0052] TU 와 같은 변환 블록에서의 잔여 데이터는 공간, 픽셀 도메인에 존재하는 픽셀 차이 값들의 2차원 (2D) 어레이로 배열될 수도 있다. 변환은 잔여 픽셀 값들을 주파수 도메인과 같은 변환 도메인에서의 변환 계수들의 2차원 어레이로 변환한다.

[0053] 추가적인 압축을 위해, 변환 계수들은 엔트로피 코딩 이전에 양자화될 수도 있다. 엔트로피 코더는 그후 그 양자화된 변환 계수들에, 컨텍스트 적응 가변 길이 코딩 (Context Adaptive Variable Length Coding; CA VLC), 컨텍스트 적응 2진 산술 코딩 (CABAC), 확률 간격 파티셔닝 엔트로피 코딩 (Probability Interval Partitioning Entropy Coding ; PIPE) 등과 같은 엔트로피 코딩을 적용한다. 일부 예들에서, 비디오 인코더 (20)는 엔트로피 인코딩될 수 있는 직렬화된 벡터를 발생하기 위해, 미리 정의된 스캔 순서를 이용하여, 양자화된 변환 계수들을 스캔할 수도 있다. 다른 예들에서, 비디오 인코더 (20)는 적응적 스캔을 수행할 수도 있다. 양자화된 변환 계수들을 스캐닝하여 1차원 벡터를 형성한 후, 비디오 인코더 (20)는 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (20)는 또한 비디오 데이터를 디코딩할 때에 비디오 디코더 (30)에 의해 사용하기 위한 인코딩된 비디오 데이터와 연관되는 구문 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0054] 본 개시물은 컨텍스트 적응 2진 산술 코딩 (CABAC) 엔트로피 코더들 또는 다른 엔트로피 코더들, 예컨대, 확률 간격 파티셔닝 엔트로피 코딩 (PIPE) 또는 관련되는 코더들에 대한 기법들과 관련된다. 산술 코딩은 심볼들을 비-정수 길이 코드워드들에 맵핑하는 것이 가능하기 때문에 높은 코딩 효율을 가진 많은 압축 알고리즘들에서 사용되는 엔트로피 코딩의 유형이다. 산술 코딩 알고리즘의 일 예는 H.264/AVC 에서 이용되는 컨텍스트 기반 2진 산술 코딩 (CABAC) 이다.

[0055] 일반적으로, CABAC 을 이용하여 데이터 심볼들을 엔트로피 코딩하는 것은 다음 단계들 중 하나 이상을 수반한다:

[0056] (1) 2진화: 코딩되는 심볼이 비-2진수이면, 소위 "빈들"의 시퀀스에 맵핑된다. 각각의 빈은 "0" 또는 "1"의 값을 가질 수 있다.

[0057] (2) 컨텍스트 할당: (정상 모드에서) 각각의 빈이 컨텍스트에 할당된다. 컨텍스트 모델은 주어진 빈에 대한 컨텍스트가 빈에 대해 이용 가능한 정보, 예컨대, 이전에 인코딩된 심볼들의 값들 또는 빈 개수에 기초하여 어떻게 계산되는지를 결정한다.

[0058] (3) 빈 인코딩: 빈들은 산술 인코더로 인코딩된다. 빈을 인코딩하기 위해, 산술 인코더는 빈의 확률, 즉, 빈의 값이 "0" 과 동일할 확률, 및 빈의 값이 "1" 과 동일할 확률을 입력으로서 필요로 한다. 각각의 컨텍스트의 (추정된) 확률은 "컨텍스트 상태"로서 불리는 정수 값으로 표현된다. 각각의 컨텍스트는 상태를 가지며, 따라서 그 상태 (즉, 추정된 확률)는 하나의 컨텍스트에 할당되는 빈들에 대해 동일하며, 컨텍스트들 사이에는 상이하다.

[0059] (4) 상태 업데이트: 선택된 컨텍스트에 대한 확률 (상태) 이 빈의 실제 코딩된 값에 기초하여 업데이트된다 (예컨대, 빈 값이 "1" 이었으면, "1들"의 확률이 증가된다).

[0060] 확률 간격 파티셔닝 엔트로피 코딩 (PIPE) 이 산술 코딩의 원리들과 유사한 원리들을 이용하며, 따라서, 또한 본 개시물의 기법들을 이용할 수 있다는 점에 유의해야 한다.

[0061] H.264/AVC 및 HEVC 에서의 CABAC 은 상태들을 이용하며, 각각의 상태는 확률에 내재적으로 (implicitly) 관련된다. 심볼 ("0" 또는 "1")의 확률이 직접 사용되는, 즉, 확률 (또는, 그의 정수 버전) 이 상태인, CABAC 의 변형들이 존재한다. 예를 들어, CABAC 의 이런 변형들은, 이하에 "JCTVC-A114" 로서 지칭되는, 2010년 4월, 독일, 드레스덴, JCTVC-A114, 1차 JCT-VC 회의, "Description of video coding technology proposal by

France Telecom, NTT, NTT DOCOMO, Panasonic and Technicolor", 및 이하에 "JCTVC-F254" 로서 지칭되는, 2011년 7월, 이탈리아, 토리노, JCTVC-F254, 6차 JCT-VC 회의, A. Alshin 및 E. Alshina, "Multi-parameter probability update for CABAC" 에 설명되어 있다.

[0062] 양자화된 변환 계수들의 블록을 엔트로피 코딩하기 위해, 스캐닝 프로세스는, 보통, 블록에서의 양자화된 변환 계수들의 2차원 (2D) 어레이가 특정의 스캔 순서에 따라서, 변환 계수들의 순서정렬된, 1차원 (1D) 어레이, 즉, 벡터로 재정렬되도록, 수행된다. 엔트로피 코딩이 그후 변환 계수들의 벡터에 적용된다. 변환 유닛에서의 양자화된 변환 계수들의 스캔은 엔트로피 코더에 대한 변환 계수들의 2D 어레이를 직렬화한다. 유의도 맵은 유의한 (즉, 비-제로) 계수들의 위치들을 나타내기 위해 발생될 수도 있다. 스캐닝은 유의 (즉, 비제로) 계수들의 레벨들을 스캔하거나, 및/또는 유의 계수들의 부호들을 코딩하는데 적용될 수도 있다.

[0063] 차기 HEVC 표준에서, 유의한 변환의 위치 정보 (예컨대, 유의도 맵)는 스캔 순서에서 최종 비-제로 계수의 로케이션을 나타내기 위해 TU에 대해 먼저 코딩된다. 유의도 맵 및 레벨 정보 (계수들의 절대값들 및 부호들)는 각각의 계수에 대해 역 스캔 순서로 코딩된다.

[0064] 현재, HEVC 에서는 4 가지 상이한 스캔들이 존재한다: 대각선 스캔, 수평 스캔, 수직 스캔 및 서브블록 기반의 대각선 스캔. 도 2 는 변환 계수들의 블록에 대한 역 스캔 순서들의 예들을 나타낸다. 역 대각선 패턴 (35), 역 지그-재그 패턴 (29), 역 수직 패턴 (31), 및 역 수평 패턴 (33)의 각각이 변환 블록의 하부 우측 코너에서의 더 높은 빈도의 계수들로부터 변환 블록의 상부 좌측 코너에서의 낮은 빈도의 계수들로 진행한다는 점에 유의한다.

[0065] 대각선 스캔, 수평 스캔, 및 수직 스캔이 4×4 및 8×8 TU들에 대해 적용된다. 서브블록 기반의 대각선 스캔이 현재의 HEVC 테스트 모델에서의 16×16 및 32×32 TU들에 채용된다. 일부 예들에서, 서브블록 기반의 대각선 스캔이 또한 8×8 TU에 채용될 수 있을 것이다. 서브블록 기반의 스캔에서, 더 큰 TU의 하나의 4×4 서브블록이 더 큰 TU를 가진 또 다른 4×4 서브블록으로 진행하기 전에 스캔된다. 다른 예들에서, "서브블록"은 사용되는 스캔 순서에 따라서 다수의 연속적으로 스캔된 계수들로 구성될 수도 있다. 예를 들어, "서브블록"은 대각선 스캔 순서를 따라서 16 개의 연속으로 스캔된 계수들로 구성될 수도 있다.

[0066] 도 3 은 8×8 TU에 대한 서브블록 기반의 스캔의 일 예를 도시한다. 8×8 TU (38)는 4개의 4×4 서브블록들 (36A, 36B, 36C, 36D)로 구성된다. 도 3에 나타낸 바와 같이, 서브블록 (36A)에서의 변환 계수들이 서브블록 (36B)에서의 변환 계수들을 스캐닝하기 전에 스캔된다. 스캔은 그후 서브블록 (36B)으로부터 서브블록 (36C)으로, 그리고 최종적으로, 서브블록 (36D)으로 진행한다. 도 3은 각각의 서브블록에서의 순방향 대각선 스캐닝 순서를 도시하지만; 그러나, 임의의 스캐닝 순서가 이용될 수도 있다 (예컨대, 수평, 수직, 지그재그 등). 다른 예들에서, 도 2에 나타낸 것들과 같은, 역 스캐닝 순서들이 각각의 서브블록 내에서 이용된다.

[0067] 차기 HEVC 표준에서, 계수들은 청크 (chunk)로 그룹화될 수도 있다. 변환 계수들의 유의도 맵 및 레벨 정보 (절대값 및 부호)는 각각의 청크에 대해 코딩된다. 일 예에서, 청크는 4×4 TU 및 8×8 TU에 대해 스캔 순서 (예컨대, 순방향 또는 역 대각선, 수평, 또는 수직 스캔 순서)를 따라서 16 개의 연속되는 계수들로 구성된다. 16×16 및 32×32 TU들에 있어, 더 큰 TU 내 변환 계수들의 4×4 서브블록은 청크로서 취급된다. 청크 내 계수들 레벨 정보를 나타내기 위해 다음 심볼들이 코딩되어 시그널링된다. 일 예에서, 모든 심볼들은 역 스캔 순서로 인코딩된다.

[0068] **significant_coeff_flag** (축약형 *sigMapFlag*): 이 플래그는 청크에서의 각각의 계수의 유의도를 나타낸다. 1 이상의 절대값을 가진 계수는 유의한 것으로 간주된다. 일 예로서, 0의 *sigMapFlag* 값은 그 계수가 유의하지 않음을 나타내지만, 1의 값은 그 계수가 유의함을 나타낸다. 이 플래그는 일반적으로 유의도 플래그로서 지칭될 수도 있다.

[0069] **coeff_abs_level_greater1_flag** (축약형 *gr1Flag*): 이 플래그는 계수의 절대값이 임의의 비-제로 계수들 (즉, 1로서 *sigMapFlag*를 가진 계수들)에 대해 1 보다 큰지 여부를 표시한다. 일 예로서, 0의 *gr1Flag* 값은 그 계수가 1 보다 큰 절대값을 갖지 않는다는 것을 표시하지만, *gr1Flag*에 있어 1의 값은 그 계수가 1 보다 큰 절대값을 갖는다는 것을 표시한다. 이 플래그는 일반적으로 1-보다-큰 플래그로서 지칭될 수도 있다.

[0070] **coeff_abs_level_greater2_flag** (축약형 *gr2Flag*): 이 플래그는 계수의 절대값이 1 보다 큰 절대값을 가진 임의의 계수들 (즉, 1로서 *gr1Flag* 를 가진 계수들)에 대해 2 보다 큰지 여부를 표시한다. 일 예로서, 0의 *gr2Flag* 값은 그 계수가 2 보다 큰 절대값을 갖지 않는다는 것을 표시하지만, *gr2Flag*에 있어 1의 값은 그 계수가 2 보다 큰 절대값을 갖는다는 것을 표시한다. 이 플래그는 일반적으로 2-보다-큰 플래그로서 지칭될 수도 있다.

[0071] **coeff_sign_flag** (축약형 *signFlag*): 이 플래그는 임의의 비-제로 계수들 (즉, 1로서 *sigMapFlag* 를 가진 계수들)에 대한 부호 정보를 표시한다. 예를 들어, 이 플래그에 있어, 0은 양의 부호를 표시하지만, 1은 음의 부호를 표시한다.

[0072] **coeff_abs_level_remaining** (축약형 *levelRem*): 이 구문 엘리먼트는 잔존 계수들의 절대 레벨 값을 표시한다. 이 플래그에 대해, 계수의 절대값 마이너스 3은 2 보다 큰 절대값을 가진 각각의 계수 (즉, 1로서 *gr2Flag* 를 가진 계수들)에 대해 (abs(레벨)-3) 이다.

[0073] 도 4는 4×4 블록 (100)에서의 양자화된 계수들의 일 예를 나타낸다. 블록 (100)은 4×4 TU 일 수도 있거나 또는 8×8 , 16×16 또는 32×32 TU에서의 4×4 서브블록 (청크) 일 수도 있다. 역 스캔 순서로 스캔되는, 도 4에 나타낸 계수들에 대한 인코딩된 심볼들이, 테이블 1에 요약된다. 테이블 1에서, *scan_pos* 는 도 4에 나타낸 역 대각선 스캔에 따른 계수의 위치를 지칭한다. *scan_pos* 15는 스캔되는 첫 번째 계수이며 블록 (100)의 하부 좌측 코너에 로케이트된다. *scan_pos* 15에서의 양자화된 계수는 0의 절대값을 갖는다. *scan_pos* 0은 스캔되는 최종 계수이며 블록 (100)의 상부 우측 코너에 로케이트된다. *scan_pos* 0에서의 양자화된 계수는 10의 절대값을 갖는다. 4×4 TU 또는 더 큰 TU에서의 최종 4×4 서브블록의 경우, 최종 비-제로 계수의 위치가 알려져 있으므로, 처음 4개의 *sigMapFlag*들은 코딩될 필요가 없다. 즉, *sigMapFlag*의 코딩은 최종 비-제로 계수 (이 예에서, *scan_pos* 11에서의 계수)에서 시작할 수도 있다.

<i>Scan_pos</i>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
<i>sigMapFlag</i>	0	0	0	0	1	1	0	1	0	1	1	1	0	1	1	1
<i>gr1Flag</i>					0	0		1		1	1	0		1	1	1
<i>gr2Flag</i>								0		1	0			1	1	1
<i>signFlag</i>							0	1		0	0	1	1	0	1	0
<i>levelRem</i>										0			2	4	7	

테이블 1. 4×4 TU 또는 4×4 청크의 계수들에 대한 코딩된 심볼들

[0074] 이들 심볼들 중에서, *sigMapFlag*, *gr1Flag* 및 *gr2Flag*의 빈들이 적응 컨텍스트 모델들을 이용하여 인코딩된다. *signFlag* 및 *levelRem*의 2진화된 빈들이 고정 동등 확률 모델들 (예컨대, 지수 Golomb 코드)을 이용하여 바이패스 모드를 통해서 인코딩된다. 현재의 HEVC 설계에서, 3개의 플래그 빈들이 0 보다 큰 진폭을 가진 계수에 대해 적응 컨텍스트 모델들을 이용하여 인코딩된다. 컨텍스트-기반의 빈 코딩이 엔트로피 코딩의 주요 병목들 중 하나인 것으로 판찰되었다.

[0076] 이 단점의 관점에서, 본 개시물은 CABAC 처리량을 향상시키는 기법들을 제시한다. 특히, 본 개시물은 컨텍스트-코딩되는 빈들의 개수를 감소시키고 바이패스-코딩되는 빈들의 개수를 증가시킴으로써, CABAC 처리량을 향상시키는 기법들을 제안한다.

[0077] 차기 HEVC 표준에서, 계수 레벨 정보는 역 스캔 순서로 코딩된다. 이것은 일반적으로 더 높은 빈도의 변환 계수들 (블록의 하부 우측 코너에 더 가까운 계수들)이 먼저 스캔된다는 것을 의미한다. 이런 설계에서,

역 스캔 순서에서의 초기 계수들은 작은 절대값들을 갖는 경향이 있다. 이런 계수들이 유의하면, 이들은 1 또는 2 의 절대값을 가질 더 높은 가능성을 갖는 경향이 있다. 그들 계수들에 대해, 명시적인 심볼들 gr1Flag 및 gr2Flag 을 이용하는 것은 계수 레벨을 나타내는데 2진화된 빈들의 길이를 감소시킬 수 있으며, gr1Flag 및 gr2Flag 가, 할당된 컨텍스트를 이용하여 이전에 코딩된 콘텐츠에 따라서 산술 코딩 엔진에 의해 효율적으로 인코딩될 수 있다.

[0078] 그러나, 역 스캔 순서에서 잔존 계수들의 경우, 심볼들 gr1Flag 및 gr2Flag 을 이용하는 것은 그 계수들이 더 큰 절대 레벨 값들을 갖는 경향이 있어, 압축 성능을 향상시키지 않을 수도 있다. gr1Flag 및 gr2Flag 를 이용하는 것이 오히려 코딩 효율을 저하시킬 수 있다.

[0079] 본 개시물의 하나의 목표는, 명시적인 gr1Flag 및 gr2Flag 를, 고정 확률 모델과 함께 바이패스 모드를 이용하여 코딩되는 levelRem 구문으로 적응적으로 스위칭함으로써, 컨텍스트-코딩되는 빈들의 개수를 감소시키는 것이다. 바이패스 모드는 CABAC 엔진과는 상이하게 동작하는 바이패스 코딩 엔진에 의해 처리된다. 바이패스 코딩 엔진은 예를 들어, Golomb 또는 지수 Golomb 코드를 이용할 수도 있다. levelRem 구문은 levelRem 값 (예컨대, 2 보다 큰 잔존 레벨) 을 Golomb 코드로 2진화하고 2진화된 값을 동등 확률 모델을 이용하여 바이패스 모드에서 인코딩함으로써 일반적으로 코딩된다. 요약하면, 본 개시물은 오직 청크 또는 TU 의 계수들의 서브세트에 대해서만, 명시적인 sigMapFlag, gr1Flag 및/또는 gr2Flag 를 인코딩하는 여러 예시적인 기법들을 제안한다.

[0080] 일 예에서, 본 개시물은 gr2Flag 로 명시적으로 인코딩되는 청크에서의 계수들의 개수를 제한하는 것을 제안한다. HEVC 에 대한 일부 제안들에서, 청크에서 gr2Flag 로 명시적으로 인코딩되는 계수들의 개수는 16 개의 계수들 (예컨대, 모든 계수들의 절대값들이 1보다 더 클 경우 그 청크의 모든 계수들) 까지 일 수 있다. 본 개시물은 스캔 순서를 따라서 1 보다 큰 절대값을 가진 청크에서의 단지 처음 N 개의 계수들 (즉, 1 로서 gr1Flag 를 가진 계수들) 에 대해 명시적인 gr2Flag 를 인코딩하는 것을 제안한다. 값 N 은 비디오 인코더에 의해 선택가능할 수도 있으며, 0 으로부터 16 까지의 임의의 값으로 설정될 수도 있다. 또 다른 예에서, N 은 16 보다 작은 임의의 값으로, 예컨대, 16 개의 계수들의 세트에서의 계수들 모두보다 작은 계수들이 gr2Flag 구문 엘리먼트로 코딩되도록, 선택된다. 0 과 동일한 N 은 심볼 gr2Flag 가 거의 코딩되지 않는다는 것을 의미한다.

[0081] 일 예로서, 본 개시물은 모든 청크들에 대해 고정된 N 값을 적용하는 것을 제안한다. 하나의 구체적인 예에서, N 은 1 로 설정된다. 이와 같이, gr2Flag 는 gr2Flag 코딩이 수행되는 첫번째 계수에 대해서만 (즉, 1 보다 큰 절대값을 갖는 첫번째 계수 (1 의 gr1Flag) 에 대해서만) 코딩된다. gr2Flag 는 변환 계수가 1보다 더 크지 않은 것을 나타내기 위해 gr1Flag 가 코딩되었으면 코딩될 필요가 없다. 이 경우, 변환 계수가 2 보다 더 크지 않은 것으로 현재 알려져 있으므로, gr2Flag 를 코딩하는 것이 불필요하다. N 의 값으로서 1 을 이용하는 것은 코딩 효율과 컨텍스트-기반의 빈들의 개수 사이에 상충관계를 제공한다. 임의의 N 의 값이 선택될 수도 있지만, 2 또는 4 의 값들이 또한 N 에 대해 적합할 수도 있다.

[0082] 테이블 2 는 N 이 1 과 동일할 때 도 4 에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 1 에서의 코딩된 심볼들과 비교할 때, X 로 마크된 위치들에서 5개의 gr2Flags 가 스킵된다 (scan_pos 6, 5, 2, 1, 0). 대응하여, 스캔 위치들 6, 5, 2, 1, 및 0 에서의 계수들의 levelRem 이, 테이블 1 에 나타낸 것과 비교하여, 변경된다. 이들 위치들에 대해, levelRem 빈의 값은 (abs(레벨)-3) 대신, (abs(레벨)-2) 로 계산된다.

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
<i>sigMapFlag</i>	0	0	0	0	1	1	0	1	0	1	1	1	0	1	1	1
<i>gr1Flag</i>					0	0		1		1	1	0		1	1	1
<i>gr2Flag</i>							0		X	X			X	X	X	
<i>signFlag</i>					0	1		0	0	0	1	1	0	1	0	
<i>levelRem</i>								1	0			3	5	8		

[0083] **테이블 2. N 이 1 과 동일할 때 청크에 대한 코딩된 심볼들**

[0084] 위에서 설명한 바와 같이, 명시적인 *gr2Flag* 의 개수를 N 으로 제한할 때, 청크에서의 계수 레벨들에 대한 코딩된 심볼들은 다음과 같이 요약될 수 있다:

[0085] *sigMapFlag*: 계수의 절대값이 0 보다 큰지 여부를 표시한다.

[0086] *gr1Flag*: 계수의 절대값이 값 1 의 *sigMapFlag* 를 가진 계수들에 대해 1 보다 큰지 여부를 표시한다.

[0087] *gr2Flag*: 계수의 절대값이 값 1 의 *gr1Flag* 를 가진 처음 N 개의 계수들에 대해 2 보다 큰지 여부를 표시한다.

[0088] *signFlag*: 1 로서 *sigMapFlag* 를 가진 계수들의 부호를 표시한다.

[0089] *levelRem*: 잔존 계수들에 대한 절대 레벨값을 나타내며; 값 1 의 *gr2Flag* 를 가진 계수들에 대해 값 (abs(레벨)-3) 이 사용되며; 명시적인 *gr2Flag* 가 코딩되지 않은 (테이블 2 에서 X 로 표시된) 계수들 및 값 1 의 *gr1Flag* 에 대해 값 (abs(레벨)-2) 이 시그널링된다.

[0090] 위에서 설명한 기법들에 따르면, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그들 (significance map flags) 을 코딩하고; 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하고; 그리고 청크에서의 변환 계수들에 대해 레벨 잔존 값 (level remaining value) 을 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시한다. 잔존 계수들 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내며, 잔존 계수들 값은 코딩된 1-보다-큰 플래그를 1 로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타낸다.

[0091] 또 다른 예에서, 본 개시물은 *gr1Flag* 로 명시적으로 인코딩되는 청크에서의 계수들의 개수를 제한하는 것을 제안한다. HEVC 에 대한 일부 제안들에서, 청크에서의 코딩된 *gr1Flag* 구문 엘리먼트들의 개수는 그 계수들 모두가 유의한 경우 16 개에 이를 수 있다. 일 예로서, 본 개시물은 청크에서의 역 스캔 순서를 따라서 오직 처음 M1 개의 비-제로 계수들에 대해 명시적인 *gr1Flag* 를 인코딩하는 것을 제안한다. M1 은 0 으로부터 16 까지의 임의의 값으로 선택될 수도 있다. 또 다른 예에서, M1 은 16 미만인 임의의 값으로, 예컨대, 16 개의 계수들의 세트에서의 모든 계수들보다 적은 수의 계수가 *gr1Flag* 구문 엘리먼트로 코딩되도록, 선택된다. 0 과 동일한 M1 은 심볼 *gr1Flag* 가 거의 인코딩되지 않는다는 것을 의미한다.

[0092] 일 예에서, 본 개시물은 모든 청크들에 대해 고정된 M1 값을 적용하는 것을 제안한다. 하나의 구체적인 예에서, M1 은 8 로 설정된다. 이와 같이, *gr1Flag* 는 *gr1Flag* 코딩이 수행되는 처음 8개의 계수들에 대해 (즉, 0 보다 큰 절대값을 갖는 처음 8개의 계수들 (1 의 *sigMapFlag*) 에 대해) 오직 코딩된다. M1 의 값으로 8 을 이용하는 것은 코딩 효율과 컨텍스트-기반의 빈들의 개수 사이에 상충관계를 제공한다. 2 또는 4 가 또한 청크에서의 계수들에 대한 M1 의 적합한 값들일 수도 있다. 테이블 3 은 M1 이 4 와 동일할 때 도 4 에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 4 는 M1 이 4 와 동일할 때 도 8 에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 1 에서의 심볼들과 비교할 때, M1 이 4 와

동일할 때, X로 마크된 위치들에서 5개의 gr1Flag 및 4개의 gr2Flag 가 스kip된다. 테이블 1에서의 심볼들과 비교할 때, M1이 8과 동일할 때, X로 마크된 위치들에서 하나의 gr1Flag 및 하나의 gr2Flag 가 스kip된다. 테이블들 3 및 4에서 X로 마킹된 위치들에 대해, levelRem 빈의 값은 $(\text{abs(레벨)}-3)$ 대신, $(\text{abs(레벨)}-1)$ 을 이용하여 계산된다.

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
sigMapFlag	0	0	0	0	1	1	0	1	0	1	1	1	0	1	1	1
gr1Flag					0	0		1		1	X	X		X	X	X
gr2Flag							0		1	X			X	X	X	
signFlag							0	1		0	0	1	1	0	1	0
levelRem									0	1			4	6	9	

[0093] 테이블 3. M1이 4와 동일할 때 청크에 대한 코딩된 심볼들

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
sigMapFlag	0	0	0	0	1	1	0	1	0	1	1	1	0	1	1	1
gr1Flag					0	0		1		1	1	0		1	1	X
gr2Flag							0		1	0			1	1	X	
signFlag							0	1		0	0	1	1	0	1	0
levelRem									0	1			2	4	9	

[0094] 테이블 4. M1이 8와 동일할 때 청크에 대한 코딩된 심볼들

[0095] gr1Flags의 최대 개수를 M1 까지 제한할 때, 청크에서의 계수 레벨들의 코딩된 심볼들은 다음과 같이 요약될 수 있다:

[0096] **sigMapFlag**: 계수의 절대값이 0 보다 큰지 여부를 표시한다.

[0097] **gr1Flag**: 계수의 절대값이 처음 M1 개의 비-제로 계수들 (값 1의 sigMapFlag)에 대해 1 보다 큰지 여부를 표시한다.

[0098] **gr2Flag**: 계수의 절대값이 값 1의 gr1Flag 를 가진 계수들에 대해 2 보다 큰지 여부를 표시한다.

[0099] **signFlag**: 1로서 sigMapFlag 를 가진 계수들의 부호를 표시한다.

[0100] **levelRem**: 잔존 계수들에 대한 절대 레벨 값을 나타내며; gr2Flag 가 1의 값을 갖는 계수들에 대해 값 $(\text{abs(레벨)}-3)$ 이 사용되며; 명시적인 gr1Flag 가 코딩되지 않은 비-제로 계수들에 대해 값 $(\text{abs(레벨)}-1)$ 이 사용된다.

[0101] 위에서 설명한 기법들에 따르면, 비디오 인코더 (20) 및/또는 비디오 디코더 (30)는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다. 비디오 인코더 (20)

및/또는 비디오 디코더 (30)는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성될 수도 있다. 잔존 계수들 값을 코딩된 2-보다-큰 플래그를 1로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3을 나타내며, 잔존 계수들 값을 코딩된 유의도 맵 플래그를 갖지만 1-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1을 나타낸다.

[0102] 명시적으로 코딩된 gr1Flags 및 g2Flags의 개수를 제한하는 위에서 설명한 방법들이 결합될 수 있다. 이 예에서, 청크에서의 계수 레벨들에 대한 코딩된 심볼들은 다음과 같이 요약될 수 있다 (원칙적으로, N은 M1과 동일하거나 또는 더 작을 것이다):

[0103] **sigMapFlag**: 계수의 절대값이 0보다 큰지 여부를 표시한다.

[0104] **gr1Flag**: 계수의 절대값이 처음 M1개의 비-제로 계수들 (값 1의 sigMapFlag)에 대해 1보다 큰지 여부를 표시한다.

[0105] **gr2Flag**: 계수의 절대값이 값 1의 gr1Flag를 가진 처음 N개의 계수들에 대해 2보다 큰지 여부를 표시한다.

[0106] **signFlag**: 1로서 sigMapFlag를 가진 계수들의 부호를 표시한다.

[0107] **levelRem**: 잔존 계수들에 대한 절대 값을 나타내며; gr2Flag가 1의 값을 갖는 계수들에 대해 값 (abs(레벨)-3)이 사용되며; 명시적인 gr2Flag가 코딩되지 않고 gr1Flag가 1의 값을 갖는 계수들에 대해 값 (abs(레벨)-2)이 사용되며; 명시적인 gr1Flag가 코딩되지 않은 비-제로 계수들에 대해 값 (abs(레벨)-1)이 사용된다.

[0108] 아래 테이블 5는 명시적인 gr2Flag (N은 1과 동일함) 및 명시적인 gr1Flags (M1은 8과 동일함) 양쪽이 제한될 때 도 4에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 1에서의 심볼들과 비교할 때, M1이 8과 동일하고 N이 1과 동일할 때, X로서 마크된 위치들에서 하나의 gr1Flag 및 5개의 gr2Flags가 스kip된다.

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
sigMapFlag	0	0	0	0	1	1	0	1	0	1	1	1	0	1	1	1
gr1Flag					0	0		1		1	1	0		1	1	X
gr2Flag							0		X	X			X	X	X	
signFlag							0	1	0	0	0	1	1	0	1	0
levelRem									1	0			3	5	9	

테이블 5. M1이 8과 동일하고 N이 1과 동일할 때 청크에 대한 코딩된 심볼들

[0109] 테이블 5는 명시적인 gr2Flag (N은 1과 동일함) 및 명시적인 gr1Flags (M1은 8과 동일함) 양쪽이 제한될 때 도 4에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 1에서의 심볼들과 비교할 때, M1이 8과 동일하고 N이 1과 동일할 때, X로서 마크된 위치들에서 하나의 gr1Flag 및 5개의 gr2Flags가 스kip된다.

[0110] 위에서 설명한 기법들에 따르면, 비디오 인코더 (20) 및/또는 비디오 디코더 (30)는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2보다 큰 절대값을 갖는지 여부를 표시한다. 이 예에서, 비디오 인코더 (20) 및/또는 비디오 디코더 (30)는 비-제로인 청크에서의 처음 M1개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하도록 구성된다.

[0111] 비디오 인코더 (20) 및/또는 비디오 디코더 (30)는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성될 수도 있다. 레벨 잔존 값을 코딩된 2-보다-큰 플래그를 1로서 갖는 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3을 나타내며, 레벨 잔존 값을 코딩된 1-보다-큰 플래그를 1로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2를 나타내며, 레벨 잔존 값을 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는 청크에서의 변환 계수들에 대해

대응하는 계수의 절대값 마이너스 1를 나타낸다.

[0112] 또 다른 예로서, 본 개시물은, 1보다 큰 절대값을 가진 청크(예컨대, 1로서의 코딩된 gr1Flag)에서의 이전 계수들의 개수가 어떤 양 M2로 누적될 때 청크에서의 잔존 계수들에 대해 명시적인 gr1Flag를 인코딩하는 것을 스위칭 오프하는 것을 제안한다. M2는 0으로부터 16까지의 임의의 값으로 설정될 수 있다.

[0113] 일 예에서, 본 개시물은 모든 청크들에 대해 고정된 M2값을 적용하는 것을 제안한다. 1의 M2값은 코딩 효율과 컨텍스트-기반의 빈들의 개수 사이에 우수한 상충관계를 제공하는 것으로 나타났다. 2 또는 4가 또한 청크에서의 계수들에 대한 M2의 예시적인 값들이다. 테이블 6은 M2가 1과 동일할 때 도 4에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 1에서의 코딩된 심볼들과 비교할 때, X로 마크된 위치들에서 6개의 gr1Flag가 코딩되지 않는다. 이들 위치들에 대해, levelRem 코딩이 사용된다. LevelRem는 테이블 1에서와 같은 (abs(레벨)-3) 대신, (abs(레벨)-1)로 계산된다.

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
sigMapFlag	0	0	0	0	1	1	0	1	0	1	1	1	0	1	1	1
gr1Flag					0	0		1		X	X	X		X	X	X
gr2Flag								0		X	X			X	X	X
signFlag							0	1		0	0	1	1	0	1	0
levelRem									2	1			4	6	9	

테이블 6. M2가 1과 동일할 때 청크에 대한 코딩된 심볼들

[0114] 역 스캔 순서에서 1의 값을 가진 이전에 인코딩된 gr1Flags의 개수가 M2로 누적될 때 잔존 계수들에 대해 명시적인 gr1Flag 코딩을 스위칭 오프하는 방법을 이용하는 경우, 청크에서의 계수 레벨들에 대한 코딩된 심볼들은 다음과 같이 요약될 수 있다:

[0115] **sigMapFlag**: 계수의 절대값이 0보다 큰지 여부를 표시한다.

[0116] **gr1Flag**: 1과 동일한 값을 가진 이전에 코딩된 gr1Flags의 누적된 양이 M2 미만일 때 계수의 절대값이 비-제로 계수들에 대해 1보다 큰지 여부를 표시한다.

[0117] **gr2Flag**: 계수의 절대값이 1로서 코딩된 gr1Flag를 가진 계수들에 대해 2보다 큰지 여부를 표시한다.

[0118] **signFlag**: 1로서 sigMapFlag를 가진 계수들의 부호를 표시한다.

[0119] **levelRem**: 잔존 계수들에 대한 절대 레벨값을 나타내며; gr2Flag가 1로서 코딩되는 계수들에 대해 값 (abs(레벨)-3)이 코딩되며; 명시적인 gr1Flag가 코딩되지 않은 비-제로 계수들에 대해 값 (abs(레벨)-1)이 코딩된다.

[0120] 위에서 설명한 바와 같이, gr1Flag에 대해 코딩되는 빈들의 개수는 그 청크에서의 이전에 코딩된 gr1Flags의 값에 의존한다. 이것은, 비디오 코더가 청크에서의 이전에 코딩된 gr1Flags를 추적해야 하기 때문에, 빈 코딩 처리량을 방해할 수도 있다. 이 의존성의 영향을 경감시키기 위해, 비디오 코더는 gr1Flags의 그룹을 함께 인코딩하고, 그후 하나의 그룹에서의 모든 gr1Flag를 인코딩한 후 잔존 계수들에 대해 명시적인 gr1Flag를 인코딩하는 것을 스위칭 오프할지 여부를 결정하도록 구성될 수도 있다. gr1Flags의 그룹은 고정된 개수의 gr1Flags로 구성될 수도 있다. 예시적인 그룹 사이즈는 2, 3 또는 4개의 gr1Flags일 수도 있다.

[0121] 이 의존성의 영향을 경감할 수도 있는 또 다른 예는, 1보다 큰 진폭을 가진 이전에 코딩된 계수들(즉, 1로서의 코딩된 gr1Flag)의 개수가 어떤 양 M2로 누적된 후 추가적인 gr1Flags의 그룹을 인코딩하는 것을 포함한다. 이 추가적인 그룹이 코딩된 후, 명시적인 gr1Flags의 코딩이 스위칭 오프된다(즉, gr1Flag가 코딩되지 않지만, 대신에 levelRem 값이 바이패스 모드에서 코딩된다). 추가적인 그룹의 예시적인 사이즈들은 1, 2 또는 3개의 계수들일 수도 있다.

- [0123] 청크에서의 이전에 코딩된 gr1Flags 의 개수가 값 M2 로 누적될 때 명시적으로 코딩된 gr1Flags 의 개수를 한정하는 이 제안된 방법은 또한 gr2Flags 의 최대 개수를 N 으로 제한하는 방법과 결합될 수도 있다. 청크에서의 계수 레벨들에 대한 코딩된 심볼들은 다음과 같이 요약될 수도 있다 (원칙적으로, N 은 M2 보다 작을 것이다):
- [0124] ***sigMapFlag***: 계수의 절대값이 0 보다 큰지 여부를 표시한다.
- [0125] ***gr1Flag***: 1 의 값을 가진 이전에-코딩된 gr1Flags 의 누적된 개수가 M2 미만일 때 계수의 절대값이 계수들에 대해 1 보다 큰지 여부를 표시한다.
- [0126] ***gr2Flag***: 계수의 절대값이 1 로서 코딩된 gr1Flag 를 가진 처음 N 개의 계수들에 대해 2 보다 큰지 여부를 표시한다.
- [0127] ***signFlag***: 1 로서 sigMapFlag 를 가진 계수들의 부호를 표시한다.
- [0128] ***levelRem***: 잔존 계수들에 대한 절대 레벨 값을 나타내며; gr2Flag 가 1 로서 코딩되는 계수들에 대해 값 (abs(레벨)-3) 이 코딩되며; 명시적인 gr2Flag 가 코딩되지 않고 gr1Flag 가 1 로서 코딩되는 계수들에 대해 값 (abs(레벨)-2) 가 코딩되며; 명시적인 gr1Flag 가 코딩되지 않은 비-제로 계수들에 대해 값 (abs(레벨)-1) 이 코딩된다.
- [0129] 본 개시물의 또 다른 예에서, 위에서 설명한 기법들의 모두가 명시적으로 코딩된 gr1Flags 및 gr2Flags 의 개수를 감소시키기 위해 결합될 수도 있다. 결합된 방법에서 청크에서의 계수 레벨들에 대한 코딩된 심볼들은 다음과 같이 요약될 수 있다 (원칙적으로, M1 은 M2 보다 더 클 것이다):
- [0130] ***sigMapFlag***: 계수의 절대값이 0 보다 큰지 여부를 표시한다.
- [0131] ***gr1Flag***: 계수의 절대값이 처음 M1 개의 비-제로 계수들 (값 1의 sigMapFlag) 에 대해 1 보다 큰지의 여부 및 1 의 값을 가진 이전에 코딩된 gr1flags 의 누적된 개수가 M2 미만인 때를 표시한다. 이들 조건 중 어느 것도 만족되지 않으면, gr1Flag 가 코딩되지 않는다.
- [0132] ***gr2Flag***: 계수의 절대값이 1 로서 코딩된 gr1flag 를 가진 처음 N 개의 계수들에 대해 2 보다 큰지 여부를 표시한다.
- [0133] ***signFlag***: 1 로서 sigMapFlag 를 가진 계수들의 부호를 표시한다.
- [0134] ***levelRem***: 잔존 계수들의 절대값을 나타내며; gr2flag 가 1 로서 코딩되는 계수들에 대해 값 (abs(레벨)-3) 이 코딩되며; 명시적인 gr2flag 가 코딩되지 않고 gr1flag 가 1 로서 코딩되는 계수들에 대해 값 (abs(레벨)-2) 이 코딩되며; 명시적인 gr1flag 가 코딩되지 않은 비-제로 계수들에 대해 값 (abs(레벨)-1) 이 코딩된다.
- [0135] 본 개시물은 또한 청크에서의 계수들에 대해 명시적으로 인코딩된 sigMapFlags 의 개수를 제한하는 것을 제안한다. HEVC 에 대한 이전 제안들에서, 청크에서의 코딩된 sigMapFlags 의 개수는 큰 TU 에서의 비-최종 청크의 경우에 16 개 만큼이나 많은 sigMapFlags 가 있을 수도 있다.
- [0136] 또 다른 예에서, 본 개시물은 청크에서의 역 스캔 순서에서 처음 K1 개의 계수들에 대해 명시적인 sigMapFlag 를 인코딩하는 것을 제안한다. K1 은 비디오 인코더에 의해 0 으로부터 16 까지의 임의의 값으로 설정되도록 선택될 수도 있다. 또 다른 예에서, K1 은 16 미만인 임의의 값으로 설정된다. 0 과 동일한 K1 은 심볼 gr1Flag 가 거의 명시적으로 인코딩되지 않는다는 것을 의미한다.
- [0137] 본 개시물은 모든 청크들에 대해 고정된 K1 값을 적용하는 것을 제안한다. K1 에 대한 8 의 값은 코딩 효율과 컨텍스트-기반의 빈들의 개수 사이에 허용가능한 상충관계를 제공하는 것으로 나타났다. 12 가 또한 K1 의 적합한 값이다. 테이블 7 은 K1 이 8 과 동일할 때 도 4 에서의 예시적인 청크에 대해 코딩되는 심볼들을 나타낸다. 테이블 1 에서의 심볼들과 비교할 때, X 로 마크된 위치들에서 8개의 sigMapFlags, 6개의 gr1Flags, 및 5개의 gr2Flags 가 스kip된다. 대신, levelRem 심볼이 그들 위치들에 대해 코딩된다. 이 경우, levelRem 코딩은 테이블 1 에서와 같은 (abs(레벨)-3) 대신, 진폭 값 (절대값) 을 바로 이용한다.

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
<i>sigMapFlag</i>	0	0	0	0	1	1	0	1	X	X	X	X	X	X	X	X
<i>gr1Flag</i>					0	0		1		X	X	X		X	X	X
<i>gr2Flag</i>								0		X	X			X	X	X
<i>signFlag</i>							0	1		0	0	1		0	1	0
<i>levelRem</i>									3	2	1	0	5	7	10	

[0138] **테이블 7. K1 이 8 과 동일할 때 청크에 대한 코딩된 심볼들**

[0139] *sigMapFlag* 의 최대 개수를 개수 K1 까지 제한할 때, 청크에서의 계수 레벨들의 코딩된 심볼들은 다음과 같이 요약될 수 있다:

[0140] *sigMapFlag*: 청크에서의 처음 K1 개의 계수들에 대한 각각의 계수의 유의도를 나타낸다.

[0141] *gr1Flag*: 계수의 절대값이 1 의 값을 가진 *sigMapFlag* 를 갖는 계수들에 대해 1 보다 큰지 여부를 표시한다.

[0142] *gr2Flag*: 계수의 절대값이 1 로서 코딩된 *gr1flag* 를 가진 계수들에 대해 2 보다 큰지 여부를 표시한다.

[0143] *signFlag*: 1 로서 *sigMapFlag* 를 가진 계수들의 부호를 표시한다.

[0144] *levelRem*: 잔존 계수들의 절대 레벨 값을 나타내며; *gr2flag* 가 1 로서 코딩되는 계수들에 대해 값 (abs(레벨)-3) 이 코딩되며; 명시적인 *sigMapFlag* 가 코딩되지 않은 계수들에 대해 값 (abs(레벨)) 이 코딩된다.

[0145] 명시적으로 코딩된 *sigMapFlags* 의 개수를 제한하는 기법은 또한 명시적으로 코딩된 *gr1Flags* 및 *gr2Flags* 의 개수를 감소시키는 위에서 설명한 제안된 방법들과 결합될 수도 있다. 이 예에서, *levelRem* 및 *signFlag* 의 코딩 순서는 플립 (flip) 될 수도 있다. 즉, *levelRem* 코딩은 *signFlag* 코딩 이전에 수행될 수도 있다.

[0146] 또 다른 예로서, 본 개시물은 이전에 코딩된 비-제로 계수들의 개수 (즉, 1 로서의 코딩된 *sigMapFlag*) 가 어떤 양 K2 로 누적될 때 잔존 계수들에 대해 명시적인 *sigMapFlags* 를 인코딩하는 것을 스위칭 오프하는 (즉, 코딩하지 않는) 것을 제안한다. K2 는 비디오 인코더에 의해 0 으로부터 16 까지의 임의의 값으로 설정되도록 선택될 수도 있다. 또 다른 예에서, K2 는 16 미만의 임의의 값으로 설정될 수도 있다. 일 예에서, 본 개시물은 모든 청크들에 대해 고정된 K2 값을 적용하는 것을 제안한다. 8 의 K2 값은 코딩 효율과 컨택스트-기반의 빈들의 개수 사이에 허용가능한 상충관계를 제공하는 것으로 나타났다. 4 또는 12 가 또한 청크에서의 계수들에 대해 적합한 K2 의 값들이다.

[0147] 이전에 코딩된 비-제로 계수들의 개수가 어떤 양 K2 로 누적될 때 *sigMapFlag* 를 명시적으로 코딩하는 것이 스위칭 오프되는 (즉, 코딩되지 않는) 경우, 청크에서의 계수 레벨들에 대한 코딩된 심볼들은 다음과 같이 요약될 수 있다:

[0148] *sigMapFlag*: 1 과 동일한 값을 가진 이전에-코딩된 *sigMapFlags* 의 누적된 양이 K2 미만일 때 각각의 계수의 유의도를 나타낸다.

[0149] *gr1Flag*: 계수의 절대값이 *sigMapFlag* 가 1 로서 코딩되는 계수들에 대해 1 보다 큰지 여부를 표시한다.

[0150] *gr2Flag*: 계수 진폭의 절대값이 1 로서 코딩된 *gr1flag* 를 가진 계수들에 대해 2 보다 큰지 여부를 표시한다.

[0151] *signFlag*: 1 로서 *sigMapFlag* 를 가진 계수들의 부호를 표시한다.

[0152] *levelRem*: 잔존 계수들에 대한 절대 레벨 값을 나타내며; *gr2flag* 가 1 로서 코딩되는 계수들에 대해 값 (abs(레벨)-3) 이 코딩되며; 명시적인 *sigMapFlag* 가 코딩되지 않은 계수들에 대해 값 (abs(레벨)) 이 코딩된다.

[0153] 이 제안된 기법은 명시적으로 코딩된 *gr1Flags* 및 *gr2Flags* 의 개수를 감소시키는 위에서 설명된 다른 기법들과 결합될 수 있다. 일 예에서, *levelRem* 과 *signFlag* 의 코딩 순서는 플립될 수도 있다. 즉, *levelRem*

코딩은 signFlag 코딩 이전에 수행될 수도 있다.

[0154]

위에서 설명된 제안된 기법들은 명시적으로 코딩된 sigMapFlags, gr1Flags 및 gr2Flags의 개수를 제한하기 위해 고정된 임계치들 (N, M1, M2, K1, 및 K2) 을 적용한다. 또 다른 예로서, 본 개시물은 또한 임계치들 K1, K2, N, M1 및 M2 을, 인코딩된 비디오 비트스트림에서의 하이 레벨 구문에서, 예컨대, 시퀀스 파라미터 세트 (SPS), 화상 파라미터 세트 (PPS), 적응 파라미터 세트 (APS) 또는 슬라이스 헤더 (SliceHeader) 구문에서, 시그널링하는 것을 제안한다. 인코더는 인코딩된 비디오 콘텐츠의 계수 통계치들에 따라서 그들 임계치들의 최상의 값을 선택할 수도 있다. 디코더는 그후 엔트로피 디코딩 프로세스에서 그 임계치를 적용할 수도 있다.

[0155]

프로파일들 및 레벨들은 비트스트림들에 대한 제한 사항들을 규정하며, 따라서 비트스트림들을 디코딩하는데 필요로 하는 능력들을 제한한다. 프로파일들 및 레벨들은 또한 개개의 디코더 구현 예들 사이의 상호운용 포인트들 (interoperability points) 을 나타내는데 이용될 수도 있다. 각각의 프로파일은 그 프로파일에 따르는 모든 디코더들에 의해 지원될 알고리즘적 특성들 및 한계들의 서브세트를 규정한다. 각각의 레벨은 비디오 압축 표준의 구문 엘리먼트들에 의해 취해질 수도 있는 값들에 대한 한계들의 세트를 규정한다. 동일한 레벨 정의들의 세트는 모든 프로파일들과 함께 사용되지만, 개개의 구현 예들은 각각의 지원되는 프로파일에 대해 상이한 레벨을 지원할 수도 있다. 임의의 주어진 프로파일에 대해, 레벨들은 일반적으로 디코더 프로세싱 부하 및 메모리 성능에 대응한다. 또 다른 예로서, 하이 레벨 구문에서의 위에서 설명된 상이한 임계치들 N, M1, M2, K1 및 K2 는 상이한 레벨들 및 프로파일에 대해 적용될 수 있다. 예를 들어, 베이스 프로파일은 낮은 임계값을 이용할 수 있는 반면, 메인 프로파일은 높은 임계값을 이용할 수 있다.

[0156]

또 다른 예로서, 임계치들 K1, K2, N, M1 및 M2 는 코딩중인 비디오 블록의 관련되는 비디오 코딩 특성들에 따라서 적응적으로 변경될 수도 있다. 이런 특성들은 양자화 파라미터, TU 사이즈, TU 심도, 예측 모드, 칼라 성분 유형, 이웃하는 청크들 또는 블록들의 계수 통계치들 등을 포함할 수도 있다.

[0157]

예를 들어, 임계치들 N, M1, M2, K1 및 K2 는 이전에 코딩된 청크들에서의 계수 통계치들에 따라서 결정될 수 있다. 큰 TU에서의 4×4 서브블록의 경우, 임계치들 N, M1, M2, K1 및 K2 는 이전에 인코딩되어진 둘러싸는 서브블록들의 계수들의 통계치들 (예컨대, 현재 코딩된 서브블록에 대해, 우측 및 하측 서브블록) 에 따라서 결정될 수 있다. 다음의 예시적인 기법들이 적용될 수도 있다.

[0158]

- 이전에 코딩된 (또는, 둘러싸는) 청크들에서의 계수들이 일부 임계치보다 더 높은 평균 절대값을 가지면, 더 작은 값 (심지어, 제로) 이 N, M1, M2, K1 및 K2 에 대해 적용된다. 반대로, 이전에 코딩된 청크들에서의 계수들이 일부 임계치보다 더 낮은 평균 절대값을 가질 때 N, M1, M2, K1 및 K2 에 대해 더 높은 값이 이용될 수도 있다.

[0159]

- 이전에 코딩된 (또는, 둘러싸는) 청크들에서의 계수들이 일부 임계치보다 더 높은 최대 절대값을 가지면, 더 작은 값 (심지어, 제로) 이 N, M1, M2, K1 및 K2 에 대해 적용된다. 반대로, 이전에 코딩된 청크들에서의 계수들이 일부 임계치보다 더 낮은 최대 절대값을 가지면, N, M1, M2, K1 및 K2 에 대해 더 높은 값이 이용될 수도 있다.

[0160]

- 이전에 코딩된 (또는, 둘러싸는) 청크들에서의 어떤 값보다 큰 절대값을 가진 계수들의 개수가 일부 임계치보다 더 높으면, 더 작은 값 (심지어, 제로) 이 N, M1, M2, K1 및 K2 에 대해 적용된다. 반대로, 이전에 코딩된 청크들에서의 어떤 값보다 큰 절대값을 가진 계수들의 개수가 일부 임계치보다 더 낮을 때, N, M1, M2, K1 및 K2 에 대해 더 높은 값이 이용될 수도 있다.

[0161]

일 예로서, 1 보다 큰 절대값 계수들을 가진 계수들의 양이 일부 임계치 X 보다 더 높으면, gr1Flags 및 gr2Flags 코딩의 명시적인 코딩이 스위칭 오프될 수도 있다 (즉, N, M1 및 M2 를 0 으로 설정할 수도 있다). 임계치 X 의 예시적인 값들은 0, 1, 및 2 이다.

[0162]

또 다른 예로서, 임계치들 N, M1, M2, K1, 및 K2 는 현재의 청크 또는 블록에서의 이미 인코딩된 구문 엘리먼트들의 통계치들에 따라서 적응적으로 변경될 수 있다. 예를 들어, 임계치들 N, M1 및 M2 는 현재의 청크 또는 블록에서의 sigMapFlag 코딩의 통계치들에 의존할 수도 있다. 좀 더 구체적으로는, 1 의 값을 가진 sigMapFlags 의 개수가 어떤 임계치 Y 보다 크면, gr1Flags 및 gr2Flags 의 명시적인 코딩이 스위칭 오프된다 (즉, N, M1 및 M2 를 0 으로 설정한다). 임계치 Y 는 상이한 프로파일들 및 레벨들에서 상이한 값들을 가질 수 있다.

[0163]

또 다른 예로서, 임계치들 N, M1, M2, K1 및 K2 는 이전에 코딩된 청크들에서의 계수 통계치들에 따라서 결정될

수 있다. 큰 TU에서의 4×4 서브블록의 경우, 임계치들 N, M1, M2, K1 및 K2는 이전에 코딩된 서브블록들의 감쇠 모델을 이용하여 계수들의 통계치들에 따라서 결정될 수 있다.

[0164] 또 다른 예로서, 임계치들 N, M1, M2, K1 및 K2의 값은 현재의 블록을 코딩하는데 사용되는 양자화 파라미터 (QP)의 값에 의존할 수 있다. 일 예로서, 압축 성능과 인코딩 처리량 사이의 최상의 상충관계를 위해, 더 큰 임계치들 N, M1, M2, K1 및 K2는 높은 QP, 예컨대, 임계치보다 큰 QP로 양자화된 계수들에 적용될 수 있으며, 더 작은 임계치들 N, M1, M2, K1 및 K2는 낮은 QP, 예컨대, 임계치보다 작거나 또는 같은 QP로 양자화된 계수들에 적용될 수 있다.

[0165] 또 다른 예로서, 임계치들 N, M1, M2, K1 및 K2의 값은 TU 사이즈에 의존할 수 있다. N, M1, M2, K1 및 K2에 대한 더 큰 임계값들은 큰 사이즈를 가진 TU의 계수들에 적용될 수 있으며, N, M1, M2, K1 및 K2에 대한 더 작은 임계값들은 작은 사이즈를 가진 TU의 계수들에 적용될 수 있다.

[0166] 또 다른 예로서, 임계치들 N, M1, M2, K1 및 K2의 값은 예측 모드에 의존할 수 있다. 예를 들어, N, M1, M2, K1 및 K2에 대한 더 큰 임계값들은 인트라-예측 모드로 예측되는 블록의 계수들에 적용될 수 있으며, N, M1, M2, K1 및 K2에 대한 더 작은 임계값들은 인터-예측 모드로 예측되는 블록의 계수들에 적용될 수 있다. 전형적인 예로서, 예측 모드가 인트라 모드일 때, M2는 1로 설정되고 N은 1로 설정되며, 그리고 예측 모드가 인터 모드일 때, M2는 4로 설정되고 N은 1로 설정된다.

[0167] 또 다른 예로서, 임계치들 N, M1, M2, K1 및 K2의 값은 칼라 성분 유형에 의존할 수 있다. 예를 들어, N, M1, M2, K1 및 K2에 대한 더 큰 임계값들이 루마 성분의 계수들에 적용될 수 있으며, N, M1, M2, K1 및 K2에 대한 더 작은 임계값들이 크로마 성분의 계수들에 적용될 수 있거나, 또는 반대의 경우도 마찬가지이다.

[0168] 또 다른 예로서, 임계치들 N, M1, M2, K1 및 K2의 값은 TU에서의 서브블록의 위치에 의존할 수도 있다. 예를 들어, X 및 Y가 각각 TU 내 서브블록의 수평 및 수직 위치들이라 하자. 16x16 TU의 예에 대해, 6과 동일한 X+Y를 가진 모든 서브블록들에 대해 7의 임계치가 이용될 수도 있으며, 5와 동일한 X+Y를 가진 모든 서브블록들에 대해 6의 임계치가 이용될 수도 있으며, 기타 등등이다. 예컨대, 이 예에서, 임계치는 X + Y + 1과 동일하다. 16x16 TU에서의 4×4 서브블록들에 대한 위치-기반의 임계치들의 일 예가 도 5에 도시된다. 각각의 서브블록의 X, Y 위치들은 서브블록의 하측에 도시되지만, 사용되는 임계치는 서브블록의 중간에 도시된다.

[0169] 또 다른 예로서, 임계값들 N, M1, M2, K1 및 K2는 TU 사이즈가 아닌, 서브블록 위치에만 의존할 수도 있다. 예를 들어, 임계치는 $\min(\text{TH}_{\max}, \text{TH}_{\min}+X+Y)$ 일 수도 있다. 이 예에서, TH_{\max} 및 TH_{\min} 는 허용 가능한 임계치들의 최대 및 최소를 나타내는 2개의 상수들이다.

[0170] 또 다른 예로서, 임계치들 N, M1 및 M2, K1 및 K2의 값은 최종 유의 계수의 위치에 의존할 수도 있다. 예를 들어, 도 4에 나타낸 바와 같은, 최종 유의 계수의 서브블록에 대해 고정된 임계치들에서 시작하는 역 스캔 순서에 대해, 비디오 코더는 임계치를 서브블록 단위로 점차 감소시킬 수도 있다. 예를 들어, 임계치들은 각각의 연속적인 서브블록에 대해 단계 사이즈 (step size) 만큼 감소될 수도 있다. 또 다른 예로서, 임계치는 서브블록들이 상이한 대각선 서브블록 라인들에 있을 때 점차 감소될 수도 있다. 예를 들어, X 및 Y가 서브블록들의 수평 및 수직 위치라 하자. 비디오 코더는 Z와 동일한 X+Y를 갖는 모든 서브블록들에 대해 임계치 8을 사용하고, Z-△와 동일한 X+Y를 갖는 모든 서브블록들에 대해 임계치 7을 사용하고, Z-2△와 동일한 X+Y를 갖는 모든 서브블록들에 대해 임계치 6을 사용하고, 기타 등등이다.

[0171] 도 6 및 도 7은 최종 유의 계수를 가진 서브블록의 위치에서 시작하는 대각선 라인으로 임계치들을 할당하는 기법을 예시한다. 최종 유의 계수를 가진 서브블록은 대각선 음영 (shading)을 이용하여 표시된다. 최종 유의 계수를 갖는 서브세트의 임계치는 또한 TU에서 또는 서브블록에서의 최종 유의 계수의 위치에 의존하도록 변경될 수도 있다.

[0172] 위에서 설명한 바와 같이, 본 개시물의 제안된 기법들은 각각의 청크 (또는, 서브블록) 내에서 계수들 레벨 코딩에 적용되는 컨텍스트-기반의 빈들의 개수를 제한한다. 본 개시물은 또한 전체 TU (예컨대, 4×4 보다 큰 TU) 내 이웃하는 청크들의 상호-의존성을 도입하는 것을 제안한다.

[0173] 예를 들어, 명시적으로 인코딩된 gr2Flag가 현재의 청크 (또는, 서브블록)에서의 적어도 하나의 계수에 대해 스위칭 오프되었으면, 그 명시적으로 코딩된 gr2Flags는 스캔 순서에서 다음에 스캔되는 다음 청크들에 대해 자동적으로 스위칭 오프된다. 이와 유사하게, 명시적으로 인코딩된 gr1Flag가 현재의 청크 (또는, 서브블록)에서의 적어도 하나의 계수에 대해 스위칭 오프되었으면, 그 명시적으로 코딩된 gr1Flags는 스캔 순서에서

이후에 스캔되는 모든 다음 청크들에 대해 자동적으로 스위칭 오프된다.

[0174] 또 다른 예로서, 현재의 청크의 1-보다-큰 계수들 (즉, 1 보다 큰 계수들) 및 비-제로 계수들의 누적된 양이 감쇠 모델을 이용하여 다음 청크로 전파된다. 예를 들어, 누적된 "gr1Flag == 1" 의 초기 개수 및 현재의 청크의 코딩된 gr2Flags 의 개수는 이전에 코딩된 청크들에서의 "1-보다-큰" 계수들의 양에 기초하여 계산될 수 있다. 또한, 현재의 청크의 누적된 코딩된 gr1Flags 의 초기 개수는 이전에 코딩된 청크들에서의 비-제로 계수들의 양에 기초하여 계산될 수 있다.

[0175] 또 다른 예로서, 청크가 큰 TU에서의 4×4 서브블록일 경우, 현재 코딩된 서브블록의 우측 및 하측에 있는 서브블록들의 레벨 코딩에 관련된 정보가 이용될 수 있다. 예를 들어, 명시적으로 인코딩된 gr2Flag 가 현재 코딩된 서브블록의 우측 및/또는 하측에 있는 서브블록에서 이전에 스위칭 오프된 적이 있으면, 명시적으로 인코딩된 gr2Flag 는 그 서브블록에 대해 자동적으로 스위칭 오프될 수도 있다. 이와 유사하게, 명시적으로 인코딩된 gr1Flag 가 현재 코딩된 서브블록의 우측 및/또는 하측에 있는 서브블록에서 이전에 스위칭 오프된 적이 있으면, 명시적으로 인코딩된 gr1Flag 는 그 서브블록에 대해 자동적으로 스위칭 오프될 수도 있다.

[0176] 위에서 설명된 제안된 기법들은 또한 단지 4×4 청크들 대신, 전체 TU의 계수 레벨 코딩에 적용될 수도 있다. 예를 들어, 본 개시물은 또한 역 스캔 순서에서 전체 TU의 하나의 1 보다 큰 값을 가진 처음 N_{TU} 개의 계수들 (즉, 1로서 코딩된 gr1Flag)에 대해 명시적인 gr2Flag 를 인코딩하는 것을 제안한다. N_{TU} 는 0 으로부터 MaxCoeffNum (즉, TU에서의 계수들의 최대 개수; 4×4 TU에서 16, 8×8 TU에서 64, 16×16 TU에서 256, 32×32 TU에서 1024) 까지의 임의의 값이 되도록 선택가능하다. 또 다른 예에서, N_{TU} 는 MaxCoeffNum 미만인 임의의 값이 되도록 설정된다. 일 예에서, 고정된 N_{TU} 값이 모든 TU 사이즈들에 대한 계수들 레벨 코딩에 적용될 수 있다. 코딩 효율과 컨텍스트-기반의 빈들의 개수 사이의 절충으로서, N_{TU} 는 4 의 값으로 설정될 수 있다. 또 다른 예로서, 상이한 N_{TU} 값들이, 이들이 상이한 통계치들을 나타내므로, 변형 TU 사이즈들에 적용될 수 있다. 더 작은 N_{TU} 값들이 더 작은 사이즈들을 가진 TU들의 경우에 이용될 수 있으며, 더 큰 N_{TU} 값들이 더 큰 사이즈들을 가진 TU들의 경우에 이용될 수 있다. 전형적인 예로서, N_{TU} 는 4×4 TU에 대해 1로, 8×8 TU에 대해 2로, 16×16 TU에 대해 4로, 그리고 32×32 TU에 대해 8로 설정될 수 있다.

[0177] 본 개시물은 또한 전체 TU의 역 스캔 순서에서 처음 $M1_{TU}$ 개의 비-제로 계수들에 대해 명시적인 gr1Flag 를 인코딩하는 것을 제안한다. $M1_{TU}$ 는 0 으로부터 TU의 MaxCoeffNum 까지의 임의의 값으로 설정될 수 있다. 상이한 $M1_{TU}$ 값들은, 이들이 상이한 통계치들을 나타내므로, 변형 TU 사이즈들에 대해 설정될 수도 있다. 더 작은 $M1_{TU}$ 값들은 더 작은 사이즈들을 가진 TU들의 경우에 이용될 가능성이 있으며, 더 큰 $M1_{TU}$ 값들은 더 큰 사이즈들을 가진 TU들의 경우에 이용될 가능성이 있다. 전형적인 예로서, $M1_{TU}$ 는 4×4 TU에 대해 4로, 8×8 TU에 대해 8로, 16×16 TU에 대해 16으로, 그리고 32×32 TU에 대해 32로 설정될 수 있다.

[0178] 본 개시물은 또한 전체 TU의 역 스캔 순서에서 1 보다 큰 절대값을 가진 이전에 인코딩된 계수들 (즉, 1로서의 코딩된 gr1Flag)의 개수가 어떤 양 $M2_{TU}$ 로 누적될 때 잔존 계수들에 대해 명시적인 gr1Flag 를 인코딩하는 것을 스위칭 오프하는 것을 제안한다. $M2_{TU}$ 는 0 으로부터 MaxCoeffNum 까지의 임의의 값으로 설정될 수 있다. 일 예로서, 고정된 $M2_{TU}$ 가 모든 TU 사이즈들에 대한 계수들 레벨 코딩에 적용될 수도 있다. 또 다른 예로서, 상이한 $M2_{TU}$ 값들이 변형 TU 사이즈들에 대해 설정될 수 있다. 더 작은 $M2_{TU}$ 값들이 더 작은 사이즈들을 가진 TU들의 경우에 이용될 수 있으며, 더 큰 $M2_{TU}$ 값들이 더 큰 사이즈를 가진 TU들의 경우에 이용될 수 있다. 전형적인 예로서, $M2_{TU}$ 는 4×4 TU에 대해 1로, 8×8 TU에 대해 2로, 16×16 TU에 대해 4로, 그리고 32×32 TU에 대해 8로 설정될 수 있다.

[0179] 또 다른 예에서, 본 개시물은 또한 전체 TU의 역 스캔 순서에서 처음 $K1_{TU}$ 개의 계수들에 대해 명시적인 sigMapFlag 를 코딩하는 것을 제안한다. $K1_{TU}$ 는 TU의 0 으로부터 MaxCoeffNum (계수들의 최대 개수) 까지의 임의의 값으로 설정될 수 있다. 상이한 $K1_{TU}$ 값들은, 이들이 최대 계수들의 상이한 개수들을 나타내므로, 변형 TU 사이즈들에 대해 설정될 수도 있다. 더 작은 $K1_{TU}$ 값들은 더 작은 사이즈들을 가진 TU들의 경우에 이용될 수도 있으며, 더 큰 $K1_{TU}$ 값들은 더 큰 사이즈들을 가진 TU들의 경우에 이용될 수도 있다. 전형적인

예로서, $K1_{TU}$ 는 4×4 TU 에 대해 8 로, 8×8 TU 에 대해 32 로, 16×16 TU 에 대해 128 로, 그리고 32×32 TU 에 대해 512 로 설정될 수 있다.

[0180] 또 다른 예에서, 본 개시물은 전체 TU 의 스캔 순서에서 1 과 동일한 이전에 인코딩된 sigMapFlag 의 개수가 어떤 양 $K2_{TU}$ 로 누적될 때 잔존 계수들에 대해 명시적인 sigMapFlags 를 인코딩하는 것을 스위칭 오프하는 것을 제안한다. $K2_{TU}$ 는 0 으로부터 MaxCoeffNum 까지의 임의의 값으로 설정될 수 있다. 고정된 $K2_{TU}$ 는 모든 TU 사이즈들에 대한 계수들 레벨 코딩에 적용될 수 있다. 이의 대안으로, 상이한 $K2_{TU}$ 값들이 변형 TU 사이즈들에 대해 설정될 수 있다. 더 작은 $K2_{TU}$ 값은 더 작은 사이즈들을 가진 TU들의 경우에 이용될 수 있으며, 더 큰 $K2_{TU}$ 값은 더 큰 사이즈들을 가진 TU들의 경우에 이용될 수 있다. 전형적인 예로서, $K2_{TU}$ 는 4×4 TU 에 대해 4 로, 8×8 TU 에 대해 16 으로, 16×16 TU 에 대해 32 로, 그리고 32×32 TU 에 대해 64 로 설정될 수 있다.

[0181] 전체 TU 에 대한 gr1Flags, gr2Flags, 및 sigMapFlags 의 명시적인 코딩을 제한하는 위에서 설명된 기법들 각각이 결합될 수도 있다. 또한, 임계치들 N, M1, M2, K1 및 K2 에 대해 위에서 설명된 적응적 임계치 방식들이 $K1_{TU}$, $K2_{TU}$, N_{TU} , $M1_{TU}$ 및 $M2_{TU}$ 를 적응적으로 변화시키기 위해 또한 적용될 수 있다. TU-기반 방식의 시나리오에서, 둘러싸는 블록은 TU 의 좌측과 상측을 의미한다.

[0182] 위에서 설명한 기법들에 따르면, 명시적인 gr1Flag, gr2Flag, 및/또는 sigMapFlag 는 청크에서의 제한된 개수의 계수들에 대해서만 코딩된다. 이와 같이, (작은 절대값을 가진 잔존 계수들을 포함한) 더 많은 계수들이 예컨대, Golomb 코딩, Golomb-Rice 코딩 또는 지수 Golomb 코딩을 이용하여 바이패스 모드에서 바이패스 코딩된다. 상기 기법들에 따르면, 본 개시물은 또한 이전 HEVC 테스트 모델들에서 코딩된 levelRem 값들과 비교하여 상이한 levelRem 값들을 수용하기 위해 Golomb-Rice 코드 적응 파라미터들을 수정하는 것을 제안한다.

[0183] 본 개시물의 일부 예들에 따르면, Golomb-Rice 업데이팅 방법은 코딩중인 계수들의 특성들에 따라서 적용될 수 있다. 일 예로서, GolombRiceParam 업데이트 프로세스는 양자화 파라미터, TU 사이즈, TU 심도, 예측 모드, 현재의 블록의 칼라 성분 유형 (예컨대, Y, U 또는 V), 및 이웃하는 청크들 또는 블록들의 계수들 통계치들 등과 같은, 코딩중인 청크의 비디오 블록의 관련되는 특성들에 의존한다.

[0184] 일반적으로, 본 개시물에서 설명되는 기법들은 Golomb-Rice 코딩이 더 일찍 사용되도록, 청크 내 또는 TU 내 여러 조건들 하에서 gr2Flag, gr1Flag 및/또는 sigMapFlag 중 하나 이상의 코딩을 스위칭 오프한다. 대안적인 예에서, 이들 결정들은 통계치들 또는 비디오 코딩 특성들, 예컨대 유의 계수들의 개수, 이웃하는 TU 들에서 1 보다 더 크거나 또는 2 보다 큰 계수들의 개수에 기초할 수도 있다. 이의 대안으로, 결정들은 슬라이스의 시작 이후 누적 통계치들에 기초할 수도 있다. 또 다른 예에서, 더 높은 레벨 구문 (예컨대, SPS, PPS, APS 또는 슬라이스 헤더)에서의 플래그는 스위칭이 슬라이스의 시작으로부터 바로 일어나야 한다는 것을 나타낼 수도 있다.

[0185] 본 개시물의 또 다른 예에서, 청크, TU, 이웃하는 TU들에서 이 시점까지, 또는 슬라이스의 시작 이후에 코딩된 계수들의 절대값들의 총합이 gr2Flag, gr1Flag 및 sigMapFlag 중 하나 이상의 코딩을 스위칭 오프하는 기준으로서 사용된다.

[0186] 일반적으로, gr2Flag, gr1Flag, 또는 sigMapFlag 와 같은, 특정의 구문 엘리먼트를 스위칭 오프하는 것은, 그 청크에서의 그 구문 엘리먼트에 대한 이미 코딩된 정보에 의존할 수도 있다. 특정의 구문 엘리먼트를 스위칭 오프하는 것은, 또한 유의도에 관련된 정보, 및 이 청크에 대해서뿐만 아니라 슬라이스의 시작 이후 이전 청크들 및 TU들에 대해서 코딩되어진 레벨 정보에 의존할 수도 있다. 예를 들어, gr2Flag 를 코딩하는 것을 스위칭 오프하는 결정은 현재의 청크에서 이전에 코딩된 2 보다 큰 계수들의 개수에 의존할 수도 있다. gr2Flag 를 코딩하는 것을 스위칭 오프하는 결정은 또한 유의 계수들의 개수 및/또는 전체 청크에서 1 보다 큰 절대 크기를 가진 계수들의 개수에 의존할 수도 있다.

[0187] HEVC 테스트 모델의 하나의 버전 (HM5.0) 에서, 전체 청크에 대한 모든 sigMapFlags 가 gr1Flags 등을 전송하기 전에 전송된다는 점에 유의해야 한다. 이와 같이, 본 개시물의 기법들에 따르면, 청크에서의 sigMapFlags 의 코딩된 정보가 gr1Flag 또는 gr2Flag 코딩을 스위칭 오프하는지 여부를 결정하는데 이용될 수도 있다. 유사한 방법으로, 이전 청크들로부터의 모든 유의도 (sigMapFlag) 및 레벨 정보 (gr1Flag, gr2Flag, levelRem) 는, 하나의 청크에 대한 모든 구문 엘리먼트들이 다음 청크로 진행하기 전에 전송되므로, sigMapFlag, gr1Flag, 및/또는 gr2Flag 코딩의 스위칭을 결정하는데 이용될 수도 있다. 이 순서 (예컨대,

인터리빙) 가 변경되면, 스위칭 오프 결정은 그 청크로부터의 이전에 인코딩된/디코딩된 계수들에 대응하는 구문 엘리먼트들에 기초할 수도 있다.

[0188] 청크들을 이용하는 대신, 전체 TU 가 HM5.0 에서 행해지는 것과 같이, 계수 단위로 또는 (sigMapFlag, gr1Flag 등과 같은) 구문 엘리먼트 플레인들 (planes) 을 이용하여 코딩되면, 특정의 컨텍스트 엘리먼트에 대해 스위칭 오프 결정을 행하기 위해 서포트의 공간 영역이 정의될 수도 있다. 서포트의 영역은 코딩중인 변환 계수에 가까운 변환 계수들의 그룹일 수도 있다. 스위칭 오프되는 구문 엘리먼트 뿐만 아니라, 이전에 더 일찍 코딩된 구문 엘리먼트들에 대한 서포트 영역으로부터의 인코딩된/디코딩된 정보가, 그 결정을 행할 때에 이용될 수 있다. 예를 들어, gr2Flag 를 스위칭 오프하는 결정은 공간 서포트 영역으로부터의 gr2Flag, gr1Flag 및 /또는 sigMapFlag 를 이용할 수 있다. 서포트의 공간 영역은 현재의 TU 에서의 변환 계수들의 (즉, 스캐닝 순서를 따라서 이미 코딩된) 인과적인 (causal) 이웃을 포함할 수도 있다. 서포트의 영역은 또한 다른 TU들로부터 동일한 빈도의 로케이션에서의 계수들을 포함할 수도 있다. 청크들이 사용되고 있으면, 서포트의 영역은 더 일찍 코딩된 구문 엘리먼트들, 예컨대, 상기 예들에서의 gr1Flag 및 sigMapFlag 에 대해 비인과적일 수도 있다.

[0189] sigMapFlag 의 코딩이 스위칭 오프되는 예들을 포함하여, 본원에서 설명되는 모든 예들에서, 최종 비-제로 계수 위치의 위치는 CABAC (또는, 임의의 다른 엔트로피 코더) 를 이용하여 코딩된다. 명확하게 하기 위해, sigMapFlags, gr1Flags 및 gr2Flags 모두가 스위칭 오프될 때, TU 에서의 최종 비-제로 계수의 위치가 여전히 코딩된다. 이것에 이어서, 최종 계수로 시작하여 역 스캔 순서로 DC 계수로 진행하는 (Golomb-Rice 코딩을 이용한) 레벨 정보가 뒤따른다.

[0190] HEVC 에 대한 일부 제안들에서, 추가적인 방법들이 변환 계수 코딩에 도입된다. 이를 방법들에서, 위에서 설명된 동일한 5개의 구문 엘리먼트들이 코딩된다. sigMapFlag, gr1Flag 및 gr2Flag 에 대해 CABAC 에 대한 컨텍스트를 유도하기 위해, 현재 코딩 계수와 가까운 계수들의 이웃과 관련된 통계치들이 사용된다. 계수들의 이웃과 관련된 통계치들이 또한 구문 엘리먼트 levelRem 에 대한 Golomb-rice 파라미터 (Golomb 파라미터) 를 결정하는데 사용될 수도 있다.

[0191] 대각선 하방-좌측 스캔의 경우, 현재 코딩된 변환 계수 X 에 대한 컨텍스트 유도에 사용되는 이웃의 일 예가, 도 8 에 점선 사변형으로 도시된다. 이 이웃은 sigMapFlags 의 코딩을 위한 일 예이지만, 다른 이웃들도 역시 사용될 수 있다. 이웃에서의 계수들은 X_0, X_1, X_2, X_3 , 및 X_4 로서 표기된다. 상이한 구문 엘리먼트들에 대한 컨텍스트 유도는 이웃에서의 계수들의 절대 레벨들의 합수에 기초한다. 각각의 서브세트 또는 청크를 5개의 경로들 (passes) (위에서 설명된 각각의 구문 엘리먼트에 대해 하나의 경로) 에서 코딩하는 대신, 하나의 변환 계수에 대한 구문 엘리먼트들이 다음 계수로 진행하기 전에 코딩될 수 있다.

[0192] 일 예로서, 다음의 양들이 컨텍스트 이웃을 이용하여 정의될 수 있다. 이 예에서, 합계는 컨텍스트 이웃에서의 모든 계수들에 결친다.

$$num_significant_coeff = \sum \delta_i(x_i) \quad (1)$$

$$\text{여기서 } \delta_i(x_i) = \begin{cases} 1 & x_i \neq 0 \\ 0 & x_i = 0 \end{cases}$$

$$sum_absolute_level = \sum |x_i| \quad (2)$$

$$sum_absolute_levelMinus1 = \sum \delta_j(x_i) \quad (3)$$

$$\text{여기서 } \delta_j(x_i) = \begin{cases} |x_i| - 1 & |x_i| > 0 \\ 0 & x_i = 0 \end{cases}$$

[0194] 그후, sum_absolute_level 이 sigMapFlag 에 대한 컨텍스트 인덱스를 유도하는데 사용될 수 있으며, sum_absolute_levelMinus1 이 gr1Flag 및 gr2Flag 에 대한 컨텍스트 인덱스들을 유도하는데 사용될 수 있다.

[0195] 위에서 설명한 바와 같이, 본 개시물의 기법들은 컨텍스트-코딩된 sigMapFlag, gr1Flag 및 gr2Flag 코딩과 Golomb-rice 코딩으로의 스위칭을 적응적으로 스위칭 오프시켜 컨텍스트-코딩된 빈들을 감소시킴으로써 CABAC

처리량을 향상시킬 수도 있다. 본 개시물은 또한 이웃 계수들로부터 컨텍스트를 결정하는 기법들 및 그들의 잠재적인 변형들로 이들 컨셉들을 확장하는 기법들을 설명한다.

[0196] 일 예로서, 현재의 계수에 대한 이웃 영역 (이웃 영역의 하나의 전형적인 예가 도 8에 도시됨) 이 정의될 수 있으며, 이웃 영역 내 계수들 레벨 정보가 현재의 계수의 특정의 컨텍스트 엘리먼트에 대해 스위칭 오프 결정을 행하는데 사용될 수 있다. 예를 들어, 계수에 대해 gr2Flag 를 스위칭 오프하는 결정은 이웃 영역 내 gr2Flag, gr1Flag 및/또는 sigMapFlag 를 이용할 수 있다. 계수에 대해 gr1Flag 를 스위칭 오프하는 결정은 이웃 영역 내 gr1Flag 및/또는 sigMapFlag 를 이용할 수 있다. 계수들이 계수 단위 (coefficient-by-coefficient) 의 방법으로 코딩될 때, 이웃 내 모든 레벨 정보가 이용가능할 수 있으며, 현재의 계수의 gr2Flag, gr1Flag 및/또는 sigMapFlag 를 스위칭 오프할지 여부를 결정하는데 이용될 수 있다. 일부의 경우, 다음에 리스트된 정보 1-6의 값들이 일부 어떤 임계치들보다 더 클 때, 현재의 계수의 gr2Flag, gr1Flag 및/또는 sigMapFlag 는 스위칭 오프될 수 있다.

[0197] 1. 수식 (1)에 정의된 바와 같은, 이웃 영역 내 유의 계수들의 개수

[0198] 2. 수식 (2)에 정의된 바와 같은, 이웃 영역 내에서의 절대 레벨의 합계

[0199] 3. 수식 (3)에 정의된 바와 같은, 이웃 영역 내에서의 절대 레벨 마이너스 1의 합계

[0200] 4. 이웃 영역 내에서의 최대 절대 레벨

[0201] 5. 이웃 영역 내에서의 임의의 다른 선형 또는 비선형 레벨 정보의 계산

[0202] 6. 항목들 1, 2, 3, 및/또는 4에서의 값들에 기초한 임의의 다른 선형 또는 비선형 계산

[0203] 계수들이 계수 단위로 코딩될 때, 위에서 언급된 이웃 영역은 인과적 계수들 (causal coefficients) 을 오직 포함한다. 계수들이 별개의 경로들로 코딩될 때, 위에서 언급된 이웃 영역은 또한 비-인과적 계수들을 포함할 수 있다. 일부 예들에서, 위에서 언급된 이웃 영역은 현재의 청크 전체 또는 심지어 현재의 TU 전체일 수 있다.

[0204] 위에서 언급한 바와 같이, 이웃 영역 내 계수들의 전체 레벨 정보가 구문 엘리먼트의 컨텍스트-기반의 코딩을 스위칭 오프하기로 결정을 행하는데 사용될 수 있다. 그러나, 전체 레벨을 코딩하는 것은 일부 경우, 스토리지 리소스들에 부담을 지울 수도 있다. 따라서, 일부 경우, 계수들 레벨은 (2, 3, 7 등 일 수 있는) 어떤 임계치로 한도가 정해지고, 그후 변환 계수에 대한 특정의 구문 엘리먼트에 대해 컨텍스트-기반의 코딩을 스위칭 오프할지 여부를 결정하는데 사용될 수 있다.

[0205] 위에서 언급한 바와 같이, 계수에 대한 특정의 컨텍스트 코딩된 구문 엘리먼트를 스위칭 오프할지 여부의 결정이 각각의 계수에 대해 이루어질 수 있다. 이의 대안으로, 특정의 컨텍스트 코딩된 구문 엘리먼트에 대한 스위칭 오프 결정이 트리거되어 전파될 수 있다. 예를 들어, 일단 특정의 컨텍스트 코딩된 구문 엘리먼트가 현재의 계수에 대해 스위칭 오프되면, 이 특정의 컨텍스트 코딩된 구문 엘리먼트가 현재의 청크에서의 모든 잔존 계수들에 대해, 또는 심지어 현재의 TU에서의 모든 잔존 계수들에 대해 스위칭 오프될 수도 있다.

[0206] 도 9는 본 개시물에서 설명되는 바와 같이, 변환 계수들을 코딩하는 기법들을 이용할 수도 있는 비디오 인코더 (20)의 일 예를 예시하는 블록도이다. 비디오 인코더 (20)는 예시의 목적으로 HEVC 코딩의 맥락에서 설명될 것이지만, 변환 계수들의 스캐닝을 필요로 할 수도 있는 다른 코딩 표준들 또는 방법들에 관해서도 본 개시물은 제한이 없다. 비디오 인코더 (20)는 비디오 프레임들 내 CU들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 내 비디오 데이터에서 공간 리던던시를 감소시키거나 또는 제거하기 위해 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 현재의 프레임과 이전에 코딩된 프레임들 사이에 시간 리던던시를 감소시키거나 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I-모드)는 여러 공간-기반의 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다. 단방향 예측 (P-모드) 또는 양방향 예측 (B-모드)과 같은 인터-모드들은 여러 시간-기반의 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다.

[0207] 도 9에 나타낸 바와 같이, 비디오 인코더 (20)는 인코딩되는 비디오 프레임 내에서 현재의 비디오 블록을 수신한다. 도 9의 예에서, 비디오 인코더 (20)는 모션 보상 유닛 (44), 모션 추정 유닛 (42), 인트라-예측 프로세싱 유닛 (46), 참조 프레임 버퍼 (64), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56)을 포함한다. 도 9에 예시된 변환 프로세싱 유닛 (52)은 잔여 데이터의 블록에 실제 변환 또는 변환의 조합들을 적용하는 유닛이며, CU의 변환 유닛 (TU)으로서 또한 지칭될 수도 있는 변환

계수들의 블록과 혼동되지 않아야 한다. 비디오 블록 재구성을 위해, 비디오 인코더 (20)는 또한 역양자화 유닛 (58), 역변환 프로세싱 유닛 (60), 및 합산기 (62)를 포함한다. 디블록킹 필터 (도 9에 미도시)가 또한 블록 경계들을 필터링하여 재구성된 비디오로부터 블록킹 현상 아티팩트들을 제거하기 위해 포함될 수도 있다. 원활 경우, 디블록킹 필터는 일반적으로 합산기 (62)의 출력을 필터링할 것이다.

[0208] 인코딩 프로세스 동안, 비디오 인코더 (20)는 코딩될 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 최대 코딩 유닛들 (LCU들)과 같은 다수의 비디오 블록들로 분할될 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44)은 시간 압축을 제공하기 위해 하나 이상의 참조 프레임들에서 하나 이상의 블록들에 대해 그 수신된 비디오 블록의 인터-예측 코딩을 수행한다. 인트라-예측 프로세싱 유닛 (46)은 공간 압축을 제공하기 위해 코딩될 블록과 동일한 프레임 또는 슬라이스에서 하나 이상의 이웃하는 블록들에 대해서 그 수신된 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다.

[0209] 모드 선택 유닛 (40)은 코딩 모드들, 즉, 인트라 또는 인터 중 하나를, 예컨대, 각각의 모드에 대한 에러 (즉, 왜곡) 결과들에 기초하여 선택하고, 최종 인트라- 또는 인터-예측된 블록 (예컨대, 예측 유닛 (PU))을 합산기 (50)에 제공하여 잔여 블록 데이터를 발생하고, 그리고 합산기 (62)에 제공하여 참조 프레임에서의 사용을 위한 인코딩된 블록을 재구성할 수도 있다. 합산기 (62)는 아래에서 더 자세히 설명하는 바와 같이, 그 예측된 블록을 그 블록에 대한 역변환 프로세싱 유닛 (60)으로부터의 역 양자화된, 역변환된 데이터와 합산하여, 인코딩된 블록을 재구성한다. 일부 비디오 프레임들은 I-프레임들로서 지정될 수도 있으며, 여기서, I-프레임에서의 모든 블록들은 인트라-예측 모드에서 인코딩된다. 일부의 경우, 인트라-예측 프로세싱 유닛 (46)은 예컨대, 모션 추정 유닛 (42)에 의해 수행되는 모션 탐색이 그 블록의 충분한 예측을 초래하지 않을 때, P- 또는 B-프레임에서 블록의 인트라-예측 인코딩을 수행할 수도 있다.

[0210] 모션 추정 유닛 (42) 및 모션 보상 유닛 (44)은 고도로 통합될 수도 있지만, 개념적인 목적들을 위해 별개로 예시된다. 모션 추정 (또는, 모션 탐색)은 모션 벡터들을 발생하는 프로세스이며, 이 프로세스는 비디오 블록들에 대한 모션을 추정한다. 모션 벡터는, 예를 들어, 참조 프레임의 참조 샘플에 대한, 현재의 프레임에서의 예측 유닛의 변위를 나타낼 수도 있다. 모션 추정 유닛 (42)은 예측 유닛을 참조 프레임 버퍼 (64)에 저장된 참조 프레임의 참조 샘플들과 비교함으로써, 인터-코딩된 프레임의 예측 유닛에 대한 모션 벡터를 계산한다. 참조 샘플은 코딩중인 PU를 포함하는 CU의 부분에 픽셀 차이의 관점에서 가깝게 매칭하는 것으로 발견되는 블록일 수도 있으며, 이것은 SAD (sum of absolute difference), SSD (sum of squared difference), 또는 다른 차이 메트릭들에 의해 결정될 수도 있다. 참조 샘플은 참조 프레임 또는 참조 슬라이스 내 어디서든지 발생할 수도 있으며, 반드시 참조 프레임 또는 슬라이스의 블록 (예컨대, 코딩 유닛) 경계에서 발생하지 않을 수도 있다. 일부 예들에서, 참조 샘플은 분수 픽셀 위치에서 발생할 수도 있다.

[0211] 모션 추정 유닛 (42)은 그 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44)으로 전송한다. 모션 벡터에 의해 식별된 참조 프레임의 부분은 참조 샘플로서 지정될 수도 있다. 모션 보상 유닛 (44)은 현재의 CU의 예측 유닛에 대한 예측 값을, 예컨대, PU에 대한 모션 벡터에 의해 식별되는 참조 샘플을 취출함으로써, 계산할 수도 있다.

[0212] 인트라-예측 프로세싱 유닛 (46)은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44)에 의해 수행되는 인터-예측에 대한 대안으로서, 수신된 블록을 인트라-예측할 수도 있다. 인트라-예측 프로세싱 유닛 (46)은 블록들에 대한 좌에서 우로, 위에서 아래로의 인코딩 순서를 가정할 때, 이웃하는, 이전에 코딩된 블록들, 예컨대, 현재의 블록의 상측, 우상측, 좌상측, 또는 좌측 블록들에 대해 그 수신된 블록을 예측할 수도 있다. 인트라-예측 프로세싱 유닛 (46)은 다양한 상이한 인트라-예측 모드들로 구성될 수도 있다. 예를 들어, 인트라-예측 프로세싱 유닛 (46)은 인코딩되는 CU의 사이즈에 기초하여, 어떤 수의 방향 예측 모드들, 예컨대, 34개의 방향 예측 모드들로 구성될 수도 있다.

[0213] 인트라-예측 프로세싱 유닛 (46)은 예를 들어, 여러 인트라-예측 모드들에 대한 에러 값을 계산하여 최저 에러 값을 산출하는 모드를 선택함으로써, 인트라-예측 모드를 선택할 수도 있다. 방향 예측 모드들은 공간적으로 이웃하는 픽셀들의 값을 결합하여 그 결합된 값을 PU 내 하나 이상의 픽셀 위치들에 적용하는 기능들을 포함할 수도 있다. 일단 PU 내 모든 픽셀 위치들에 대한 값을 계산되었으면, 인트라-예측 프로세싱 유닛 (46)은 PU와 인코딩되는 수신된 블록 사이의 픽셀 차이들에 기초하여, 예측 모드에 대한 에러 값을 계산할 수도 있다. 인트라-예측 프로세싱 유닛 (46)은 허용가능한 에러 값을 산출하는 인트라-예측 모드가 발견될 때까지 인트라-예측 모드들을 계속해서 테스트할 수도 있다. 인트라-예측 프로세싱 유닛 (46)은 그 후 PU를 합산기 (50)로 전송할 수도 있다.

- [0214] 비디오 인코더 (20)는 코딩중인 원래 비디오 블록으로부터, 모션 보상 유닛 (44) 또는 인트라-예측 프로세싱 유닛 (46)에 의해 계산된 예측 데이터를 감산함으로써, 잔여 블록을 형성한다. 합산기 (50)는 이 감산 동작을 수행하는 구성요소 또는 구성요소들을 나타낸다. 잔여 블록은 픽셀 차이 값들의 2차원 매트릭스에 대응할 수도 있으며, 여기서, 잔여 블록에서의 값들의 수는 잔여 블록에 대응하는 PU 내 픽셀들의 개수와 동일하다. 잔여 블록에서의 값들은 PU에, 그리고 코딩되는 원래 블록에서 동일 장소에 배치된 픽셀들의 값들 사이의 차이들, 즉, 에러에 대응할 수도 있다. 그 차이들은 코딩되는 블록의 유형에 따라서 크로마 또는 루마 차이들일 수도 있다.
- [0215] 변환 프로세싱 유닛 (52)은 잔여 블록으로부터 하나 이상의 변환 유닛들 (TU들)을 포함할 수도 있다. 변환 프로세싱 유닛 (52)은 복수의 변환들 중에서 변환을 선택한다. 이 변환은 블록 사이즈, 코딩 모드 등과 같은 하나 이상의 코딩 특성들에 기초하여 선택될 수도 있다. 변환 프로세싱 유닛 (52)은 그후 그 선택된 변환을 TU에 적용하여, 변환 계수들의 2차원 어레이를 포함하는 비디오 블록을 발생한다.
- [0216] 변환 프로세싱 유닛 (52)은 최종 변환 계수들을 양자화 유닛 (54)으로 전송할 수도 있다. 양자화 유닛 (54)은 그후 변환 계수들을 양자화할 수도 있다. 엔트로피 인코딩 유닛 (56)은 그후 매트릭스의 양자화된 변환 계수들의 스캔을 스캐닝 모드에 따라서 수행할 수도 있다. 본 개시물은 스캔을 수행하는 것으로서 엔트로피 인코딩 유닛 (56)을 기술한다. 그러나, 다른 예들에서, 양자화 유닛 (54)과 같은, 다른 프로세싱 유닛들이 스캔을 수행할 수 있는 것으로 이해되어야 한다.
- [0217] 일단 변환 계수들이 1차원 어레이로 스캔되면, 엔트로피 인코딩 유닛 (56)은 CABAC, 구문-기반 컨텍스트-적용 2진 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE), 또는 또 다른 엔트로피 코딩 방법론과 같은, 엔트로피 코딩을 계수들에 적용할 수도 있다.
- [0218] CABAC을 수행하기 위해, 엔트로피 인코딩 유닛 (56)은 어떤 컨텍스트에 적용할 컨텍스트 모델을 선택하여, 송신되는 심볼들을 인코딩할 수도 있다. 컨텍스트는 예를 들어, 이웃하는 값들이 비-제로인지 여부에 관련될 수도 있다. 엔트로피 인코딩 유닛 (56)은 또한 그 선택된 변환을 나타내는 신호와 같은 구문 엘리먼트들을 엔트로피 인코딩할 수도 있다. 본 개시물의 기법들에 따르면, 엔트로피 인코딩 유닛 (56)은 예를 들어, 컨텍스트 모델 선택에 사용되는 다른 인자들 중에서, 인트라-예측 모드들에 대한 인트라-예측 방향, 구문 엘리먼트들에 대응하는 계수의 스캔 위치, 블록 유형, 및/또는 변환 유형에 기초하여, 이를 구문 엘리먼트들을 인코딩하는데 사용되는 컨텍스트 모델을 선택할 수도 있다.
- [0219] 엔트로피 인코딩 유닛 (56)에 의한 엔트로피 코딩 이후, 최종 인코딩된 비디오는 비디오 디코더 (30)와 같은 또 다른 디바이스로 송신되거나, 또는 추후 송신 또는 취출을 위해 아카이브될 수도 있다.
- [0220] 일부의 경우, 엔트로피 인코딩 유닛 (56) 또는 비디오 인코더 (20)의 또 다른 유닛이 엔트로피 코딩에 더해서, 다른 코딩 기능들을 수행하도록 구성될 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (56)은 CU들 및 PU들에 대한 코딩된 블록 패턴 (CBP) 값들을 결정하도록 구성될 수도 있다. 또한, 일부의 경우, 엔트로피 인코딩 유닛 (56)은 계수들의 런 길이 코딩 (run length coding)을 수행할 수도 있다.
- [0221] 역양자화 유닛 (58) 및 역변환 프로세싱 유닛 (60)은 역양자화 및 역변환을 각각 적용하여, 예컨대, 추후 참조 블록으로 사용을 위해, 픽셀 도메인에서 잔여 블록을 재구성한다. 모션 보상 유닛 (44)은 잔여 블록을 참조 프레임 버퍼 (64)의 프레임들 중 하나의 예측 블록에 가산함으로써 참조 블록을 계산할 수도 있다. 모션 보상 유닛 (44)은 또한 하나 이상의 내삽 필터들을 그 재구성된 잔여 블록에 적용하여, 모션 추정에 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62)는 재구성된 잔여 블록을 모션 보상 유닛 (44)에 의해 발생된 모션 보상된 예측 블록에 가산하여, 참조 프레임 버퍼 (64)에의 저장을 위한 재구성된 비디오 블록을 발생한다. 재구성된 비디오 블록은 후속 비디오 프레임에서 블록을 인터-코딩하기 위해 모션 추정 유닛 (42) 및 모션 보상 유닛 (44)에 의해 참조 블록으로서 사용될 수도 있다.
- [0222] 도 10은 인코딩된 비디오 시퀀스를 디코딩하는 비디오 디코더 (30)의 일 예를 예시하는 블록도이다. 도 10의 예에서, 비디오 디코더 (30)는 엔트로피 디코딩 유닛 (70), 모션 보상 유닛 (72), 인트라-예측 프로세싱 유닛 (74), 역양자화 유닛 (76), 역변환 유닛 (78), 참조 프레임 버퍼 (82) 및 합산기 (80)를 포함한다. 비디오 디코더 (30)는 일부 예들에서, 비디오 인코더 (20) (도 9 참조)에 대해 설명되는 인코딩 과정과는 일 반적으로 반대인 디코딩 과정을 수행할 수도 있다.
- [0223] 엔트로피 디코딩 유닛 (70)은 인코딩된 비트스트림에 대해 엔트로피 디코딩 프로세스를 수행하여, 변환 계수들의 1차원 어레이를 취출한다. 사용되는 엔트로피 디코딩 프로세스는 비디오 인코더 (20)에 의해 사용되는

엔트로피 코딩 (예컨대, CABAC) 에 의존한다. 인코더에 의해 사용되는 엔트로피 코딩 프로세스는 인코딩된 비트스트림에서 시그널링될 수도 있거나 또는 미리 결정된 프로세스일 수도 있다.

[0224] 일부 예들에서, 엔트로피 디코딩 유닛 (70) (또는, 역양자화 유닛 (76)) 은 비디오 인코더 (20) 의 엔트로피 인코딩 유닛 (56) (또는, 양자화 유닛 (54)) 에 의해 사용되는 스캐닝 모드를 미러링하는 스캔을 이용하여 수신된 값들을 스캔할 수도 있다. 계수들의 스캐닝이 역양자화 유닛 (76) 에서 수행될 수도 있지만, 스캐닝은 예시의 목적을 위해 엔트로피 디코딩 유닛 (70) 에 의해 수행되는 것으로 설명될 것이다. 게다가, 예시의 용이를 위해 별개의 기능적 유닛들로서 나타내지만, 엔트로피 디코딩 유닛 (70), 역양자화 유닛 (76), 및 비디오 디코더 (30) 의 다른 유닛들의 구조 및 기능은 서로 고도로 통합될 수도 있다.

[0225] 역양자화 유닛 (76) 은 비트스트림에서 제공되어 엔트로피 디코딩 유닛 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화한다, 즉, 양자화 해제한다. 역양자화 프로세스는 종래의 프로세스, 예컨대, HEVC 에 대해 제안되거나 또는 H.264 디코딩 표준에 의해 정의된 프로세스들과 유사한, 프로세스를 포함할 수도 있다.

역양자화 프로세스는 양자화의 정도를 결정하기 위해, 그리고 이와 유사하게, 적용되어야 하는 역양자화의 정도를 결정하기 위해, CU 에 대해 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터 QP 의 사용을 포함할 수도 있다. 역양자화 유닛 (76) 은 계수들이 1차원 어레이로부터 2차원 어레이로 변환되기 이전에 또는 이후에 변환 계수들을 역양자화할 수도 있다.

[0226] 역변환 프로세싱 유닛 (78) 은 역양자화된 변환 계수들에 역변환을 적용한다. 일부 예들에서, 역변환 프로세싱 유닛 (78) 은 비디오 인코더 (20) 로부터의 시그널링에 기초하여, 또는 블록 사이즈, 코딩 모드 등과 같은 하나 이상의 코딩 특성들로부터 변환을 추론함으로써, 역변환을 결정할 수도 있다. 일부 예들에서, 역변환 프로세싱 유닛 (78) 은 현재의 블록을 포함하는 LCU 에 대한 퀘드트리의 루트 노드에서의 시그널링된 변환에 기초하여 현재의 블록에 적용할 변환을 결정할 수도 있다. 이의 대안으로, 변환은 LCU 퀘드트리에서 일-노드 CU 에 대한 TU 퀘드트리의 루트에서 시그널링될 수도 있다. 일부 예들에서, 역변환 프로세싱 유닛 (78) 은 캐스케이트된 역변환을 적용할 수도 있으며, 여기서, 역변환 프로세싱 유닛 (78) 은 디코딩되는 현재의 블록의 변환 계수들에 2개의 이상의 역변환들을 적용한다.

[0227] 인트라-예측 프로세싱 유닛 (74) 은 현재의 프레임의 이전에 디코딩된 블록들로부터의 시그널링된 인트라-예측 모드 및 데이터에 기초하여, 현재의 프레임의 현재의 블록에 대한 예측 데이터를 발생할 수도 있다.

[0228] 모션 보상 유닛 (72) 은 인코딩된 비트스트림으로부터 모션 벡터, 모션 예측 방향 및 참조 인덱스를 취출할 수도 있다. 참조 예측 방향은 인터-예측 모드가 단방향 (예컨대, P 프레임) 또는 양방향 (B 프레임) 인지 여부를 표시한다. 참조 인덱스는 어느 참조 프레임에 후보 모션 벡터가 기초하는지를 나타낸다.

[0229] 취출된 모션 예측 방향, 참조 프레임 인덱스, 및 모션 벡터에 기초하여, 모션 보상 유닛은 현재의 부분에 대한 모션 보상되는 블록을 발생한다. 이들 모션 보상되는 블록들은 본질적으로 잔여 데이터를 발생하는데 사용되는 예측 블록을 재생성한다.

[0230] 모션 보상 유닛 (72) 은, 어쩌면 내삽 필터들에 기초하여 내삽을 수행함으로써 모션 보상되는 블록들을 발생할 수도 있다. 서브-픽셀 정밀도를 갖는 모션 추정에 이용되는 내삽 필터들에 대한 식별자들이 구문 엘리먼트들에 포함될 수도 있다. 모션 보상 유닛 (72) 은 비디오 블록의 인코딩 동안 비디오 인코더 (20) 에 의해 사용되는 것과 같은 내삽 필터들을 이용하여, 참조 블록의 서브-정수 픽셀들에 대한 내삽된 값들을 계산할 수도 있다. 모션 보상 유닛 (72) 은 수신된 구문 정보에 따라서 비디오 인코더 (20) 에 의해 사용되는 내삽 필터들을 결정하고 그 내삽 필터들을 이용하여 예측 블록들을 발생할 수도 있다.

[0231] 게다가, HEVC 예에서, 모션 보상 유닛 (72) 및 인트라-예측 프로세싱 유닛 (74) 은 (예컨대, 퀘드트리에 의해 제공되는) 구문 정보의 일부를 이용하여, 인코딩된 비디오 시퀀스의 프레임(들) 을 인코딩하는데 사용되는 LCU 들의 사이즈들을 결정할 수도 있다. 모션 보상 유닛 (72) 및 인트라-예측 프로세싱 유닛 (74) 은 또한 구문 정보를 이용하여, 인코딩된 비디오 시퀀스의 프레임의 각각의 CU 가 분할되는 방식 (그리고, 유사하게, 서브-CU 들이 분할되는 방식) 을 기술하는 분할 정보를 결정할 수도 있다. 구문 정보는 또한, 각각의 분할이 인코딩되는 방식을 나타내는 모드들 (예컨대, 인트라- 또는 인터-예측, 그리고 인트라-예측에 있어, 인트라-예측 인코딩 모드), 각각의 인터-인코딩된 PU 에 대한 하나 이상의 참조 프레임들 (및/또는 참조 프레임들에 대한 식별자들을 포함하는 참조 리스트들), 및 인코딩된 비디오 시퀀스를 디코딩하기 위한 다른 정보를 포함할 수도 있다.

[0232] 합산기 (80) 는 잔여 블록들을 모션 보상 유닛 (72) 또는 인트라-예측 프로세싱 유닛 (74) 에 의해 발생된 대응하는 예측 블록들과 합산하여, 디코딩된 블록들을 형성한다. 원할 경우, 블록킹 현상 아티팩트들

(blockiness artifacts) 을 제거하기 위해 디블록킹 필터가 또한 그 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 디코딩된 비디오 블록들은 그후 참조 프레임 버퍼 (82) 에 저장될 수도 있으며, 그 참조 프레임 버퍼는 후속 모션 보상을 위해 참조 블록들을 제공하며, 또한 (도 1 의 디스플레이 디바이스 (32) 와 같은) 디스플레이 디바이스 상에의 프리젠테이션을 위해 디코딩된 비디오를 발생한다.

[0233] 도 11 은 본 개시물의 기법들에 따른 예시적인 방법을 나타내는 플로우차트이다. 도 11 의 방법은 비디오 코더에 의해 (예컨대, 도 9 의 비디오 인코더 (20) 에 의해, 또는 도 10 의 비디오 디코더 (30) 에 의해) 실행될 수도 있다.

[0234] 비디오 인코더 (20) 및 비디오 디코더 (30) (즉, 비디오 코더) 는 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법을 수행하도록 구성될 수도 있다. 비디오 코더는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시한다 (1110). 비디오 코더는 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 첫 N개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하도록 추가로 구성될 수도 있으며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시한다 (1120). 비디오 코더는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 추가로 구성될 수도 있으며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 N 은 청크에서의 변환 계수들의 최대 개수 미만이다 (1130). 일 예에서, N 의 값은 1 이다. 또 다른 예에서, N 의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.

[0235] 비디오 코더는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성될 수도 있다 (1140). 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타낸다. 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타낸다. 비디오 코더는 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하도록 추가로 구성될 수도 있으며, 여기서, 부호 플래그는 특정의 변환 계수의 부호를 표시한다 (1150).

[0236] 본 개시물의 일 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그는 적응 컨텍스트 모델과 함께 CABAC 을 이용하여 코딩된다. 레벨 잔존 값은 Golomb-Rice 코딩을 이용하여 CABAC 바이패스 모드에서 코딩된다. 부호 플래그는 CABAC 바이패스 모드에서 코딩된다. 본 개시물의 또 다른 예에서, 레벨 잔존 값은 Golomb-Rice 코딩을 이용하여 코딩되며, 여기서, Golomb-Rice 코딩은 Golomb 파라미터에 따라서 적응된다. Golomb 파라미터는 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 예측 모드, 칼라 성분 유형, 및 이웃하는 청크들의 계수들 통계치들 중 적어도 하나에 기초하여 결정된다.

[0237] 본 개시물의 또 다른 예에서, 유의도 맵 플래그는 첫번째로 코딩되며, 1-보다-큰 플래그는 두번째로 코딩되며, 2-보다-큰 플래그는 세번째로 코딩되며, 그리고 레벨 잔존 값은 2-보다-큰 플래그 이후에 코딩된다. 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그, 및 레벨 잔존 값의 각각은 스캔 순서에 따라서 청크에서의 변환 계수들에 대해 코딩된다. 일 예에서, 스캔 순서는 역 대각선 스캔 순서이다.

[0238] 본 개시물의 다른 예들에 따르면, 청크는 변환 유닛의 서브블록이다. 청크는 변환 유닛의 16 개의 변환 계수들의 서브블록일 수도 있다. 또 다른 예에서, 청크는 전체 변환 유닛이다. 또 다른 예에서, 청크는 스캔 순서에 따른 다수의 연속되는 변환 계수들이다.

[0239] 비디오 인코더가 도 11 의 방법을 수행하고 있는 경우, 본 방법은, 잔여 비디오 데이터를 발생하기 위해 픽셀 데이터를 인코딩하는 단계; 변환 계수들의 청크를 발생하기 위해 잔여 비디오 데이터를 변환하는 단계; 및 코딩된 유의도 맵 플래그, 코딩된 1-보다-큰 플래그, 및 코딩된 2-보다-큰 플래그를, 인코딩된 비디오 비트스트림에서 시그널링하는 단계를 더 포함할 수도 있다. 게다가, 또 다른 예에서, 비디오 인코더는 N 의 값을 인코딩된 비디오 비트스트림에서 시그널링할 수도 있다. N 의 값은 시퀀스 파라미터 세트, 화상 파라미터 세트, 적응 파라미터 세트, 및 슬라이스 헤더 중 적어도 하나에서 시그널링될 수도 있다.

[0240] 비디오 디코더가 도 11 의 방법을 수행하고 있는 경우, 본 방법은, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 수신하는 단계; 잔여 비디오 데이터를 발생하기 위해 변환 계수들의 청크를 역변환하는 단계; 및 픽셀 데이터를 발생하기 위해 잔여 비디오 데이터를 디코딩하는 단계를

더 포함할 수도 있다.

[0241] 도 12 는 본 개시물의 기법들에 따른 또 다른 예시적인 방법을 나타내는 플로우차트이다. 도 12 의 방법은 비디오 코더에 의해 (예컨대, 도 9 의 비디오 인코더 (20) 에 의해, 또는 도 10 의 비디오 디코더 (30) 에 의해) 실행될 수도 있다.

[0242] 비디오 인코더 (20) 및 비디오 디코더 (30) (즉, 비디오 코더) 는 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법을 수행하도록 구성될 수도 있다. 비디오 코더는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시한다 (1210). 비디오 코더는 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하도록 추가로 구성될 수도 있으며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시한다 (1220). M1 은 청크에서의 변환 계수들의 최대 개수 미만이다. 비디오 코더는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 추가로 구성될 수도 있으며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다 (1230). 일 예에서, M1 의 값은 8 이다. 또 다른 예에서, M1 의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.

[0243] 비디오 코더는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성될 수도 있다 (1240). 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타낸다. 레벨 잔존 값은 코딩된 유의도 맵 플래그를 갖지만 1-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타낸다. 비디오 코더는 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하도록 추가로 구성될 수도 있으며, 여기서, 부호 플래그는 특정의 변환 계수의 부호를 표시한다 (1250).

[0244] 본 개시물의 일 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그는 적응 컨텍스트 모델과 함께 CABAC 을 이용하여 코딩된다. 레벨 잔존 값은 Golomb-Rice 코딩을 이용하여 CABAC 바이패스 모드에서 코딩된다. 부호 플래그는 CABAC 바이패스 모드에서 코딩된다. 본 개시물의 또 다른 예에서, 레벨 잔존 값은 Golomb-Rice 코딩을 이용하여 코딩되며, 여기서, Golomb-Rice 코딩은 Golomb 파라미터에 따라서 적응된다. Golomb 파라미터는 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 예측 모드, 칼라 성분 유형, 및 이웃하는 청크들의 계수들 통계치들 중 적어도 하나에 기초하여 결정된다.

[0245] 본 개시물의 또 다른 예에서, 유의도 맵 플래그는 첫번째로 코딩되며, 1-보다-큰 플래그는 두번째로 코딩되며, 2-보다-큰 플래그는 세번째로 코딩되며, 그리고 레벨 잔존 값은 2-보다-큰 플래그 이후에 코딩된다. 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그, 및 레벨 잔존 값의 각각은 스캔 순서에 따라서 청크에서의 변환 계수들에 대해 코딩된다. 일 예에서, 스캔 순서는 역 대각선 스캔 순서이다.

[0246] 본 개시물의 다른 예들에 따르면, 청크는 변환 유닛의 서브블록이다. 청크는 변환 유닛의 16 개의 변환 계수들의 서브블록일 수도 있다. 또 다른 예에서, 청크는 전체 변환 유닛이다. 또 다른 예에서, 청크는 스캔 순서에 따른 다수의 연속되는 변환 계수들이다.

[0247] 비디오 인코더가 도 12 의 방법을 수행하고 있는 경우, 본 방법은, 잔여 비디오 데이터를 발생하기 위해 픽셀 데이터를 인코딩하는 단계; 변환 계수들의 청크를 발생하기 위해 잔여 비디오 데이터를 변환하는 단계; 및 코딩된 유의도 맵 플래그, 코딩된 1-보다-큰 플래그, 및 코딩된 2-보다-큰 플래그를, 인코딩된 비디오 비트스트림에서 시그널링하는 단계를 더 포함할 수도 있다. 게다가, 또 다른 예에서, 비디오 인코더는 M1 의 값을 인코딩된 비디오 비트스트림에서 시그널링할 수도 있다. M1 의 값은 시퀀스 파라미터 세트, 화상 파라미터 세트, 적응 파라미터 세트, 및 슬라이스 헤더 중 적어도 하나에서 시그널링될 수도 있다.

[0248] 비디오 디코더가 도 12 의 방법을 수행하고 있는 경우, 본 방법은, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 수신하는 단계; 잔여 비디오 데이터를 발생하기 위해 변환 계수들의 청크를 역변환하는 단계; 및 픽셀 데이터를 발생하기 위해 잔여 비디오 데이터를 디코딩하는 단계를 더 포함할 수도 있다.

[0249] 도 13 은 본 개시물의 기법들에 따른 또 다른 예시적인 방법을 나타내는 플로우차트이다. 도 13 의 방법은 비디오 코더에 의해 (예컨대, 도 9 의 비디오 인코더 (20) 에 의해, 또는 도 10 의 비디오 디코더 (30) 에

의해) 실행될 수도 있다.

[0250] 비디오 인코더 (20) 및 비디오 디코더 (30) (즉, 비디오 코더) 는 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법을 수행하도록 구성될 수도 있다. 비디오 코더는 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하도록 구성될 수도 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시한다 (1310). 비디오 코더는 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하도록 추가로 구성될 수도 있으며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시한다 (1320). M1 은 청크에서의 변환 계수들의 최대 개수 미만이다. 비디오 코더는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 추가로 구성될 수도 있으며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 N 은 청크에서의 변환 계수들의 최대 개수 미만이다 (1330). 일 예에서, N 의 값은 1 이다. 또 다른 예에서, N 의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다. 일 예에서, M1 의 값은 8 이다. 또 다른 예에서, M1 의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.

[0251] 비디오 코더는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성될 수도 있다 (1340). 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타낸다. 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1 로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타낸다. 레벨 잔존 값은 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타낸다. 비디오 코더는 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하도록 추가로 구성될 수도 있으며, 여기서, 부호 플래그는 특정의 변환 계수의 부호를 표시한다 (1350).

[0252] 본 개시물의 일 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법은, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 단계; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 단계; 및 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하는 단계를 포함하며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 M1 은 청크에서의 변환 계수들의 최대 개수 미만이며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.

[0253] 본 개시물의 또 다른 예에서, 본 방법은 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하는 단계를 더 포함할 수도 있으며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내며, 상기 레벨 잔존 값은 코딩된 유의도 맵 플래그를 갖지만 1-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타낸다.

[0254] 본 개시물의 또 다른 예에서, 본 방법은 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하는 단계를 더 포함할 수도 있으며, 상기 부호 플래그는 특정의 변환 계수의 부호를 표시한다.

[0255] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그는 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩된다.

[0256] 본 개시물의 또 다른 예에서, 본 방법은 레벨 잔존 값을 Golomb 파라미터를 이용하여 코딩하는 단계를 더 포함할 수도 있으며, 상기 Golomb 파라미터는 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 예측 모드, 칼라 성분 유형, 및 이웃하는 청크들의 계수들 통계치들 중 적어도 하나에 기초하여 결정된다.

[0257] 본 개시물의 또 다른 예에서, 유의도 맵 플래그는 첫번째로 코딩되며, 1-보다-큰 플래그는 두번째로 코딩되며, 2-보다-큰 플래그는 세번째로 코딩되며, 그리고 레벨 잔존 값은 2-보다-큰 플래그 이후에 코딩된다.

[0258] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그, 및 레벨 잔존 값의 각각

은 스캔 순서에 따라서 청크에서의 변환 계수들에 대해 코딩된다.

[0259] 본 개시물의 또 다른 예에서, 스캔 순서는 역 대각선 스캔 순서이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 16 개의 변환 계수의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛이다. 본 개시물의 또 다른 예에서, 청크는 스캔 순서에 따른 다수의 연속되는 변환 계수들이다. 본 개시물의 또 다른 예에서, 스캔 순서는 역 스캔 순서이다.

[0260] 본 개시물의 또 다른 예에서, M1 의 값은 8 이다. 본 개시물의 또 다른 예에서, M1 의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.

[0261] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스는 비디오 인코딩 프로세스이며, 본 방법은 잔여 비디오 데이터를 발생하기 위해 픽셀 데이터를 인코딩하는 단계; 변환 계수들의 청크를 발생하기 위해 잔여 비디오 데이터를 변환하는 단계; 및 코딩된 유의도 맵 플래그, 코딩된 1-보다-큰 플래그, 및 코딩된 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 시그널링하는 단계를 더 포함한다.

[0262] 본 개시물의 또 다른 예에서, 본 방법은 M1 의 값을 인코딩된 비디오 비트스트림에서 시그널링하는 단계를 더 포함한다. 본 개시물의 또 다른 예에서, M1 의 값은 시퀀스 파라미터 세트, 화상 파라미터 세트, 적응 파라미터 세트, 및 슬라이스 헤더 중 적어도 하나에서 시그널링된다.

[0263] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스는 비디오 디코딩 프로세스이며, 본 방법은 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 수신하는 단계; 잔여 비디오 데이터를 발생하기 위해 변환 계수들의 청크를 역변환하는 단계; 및 픽셀 데이터를 발생하기 위해 잔여 비디오 데이터를 디코딩하는 단계를 더 포함한다.

[0264] 본 개시물의 또 다른 예에서, 2-보다-큰 플래그를 코딩하는 단계는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 단계를 포함하며, 상기 N 은 청크에서의 변환 계수들의 최대 개수 미만이며, 그리고 본 방법은 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하는 단계를 더 포함하며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내며, 상기 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1 로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내며, 상기 레벨 잔존 값은 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타낸다.

[0265] 본 개시물의 또 다른 예에서, M1 의 값은 8 이다.

[0266] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하도록 구성된 장치는, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성된 비디오 코더를 포함하며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 M1 은 청크에서의 변환 계수들의 최대 개수 미만이며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.

[0267] 본 개시물의 또 다른 예에서, 비디오 코더는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성되며; 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내며, 상기 레벨 잔존 값은 코딩된 유의도 맵 플래그를 갖지만 1-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타낸다.

[0268] 본 개시물의 또 다른 예에서, 비디오 코더는 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하도록 추가로 구성되며, 상기 부호 플래그는 특정의 변환 계수의 부호를 표시한다.

[0269] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그는 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩된다.

- [0270] 본 개시물의 또 다른 예에서, 비디오 코더는 레벨 잔존 값을 Golomb 파라미터를 이용하여 코딩하도록 추가로 구성되며, 상기 Golomb 파라미터는 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 예측 모드, 칼라 성분 유형, 및 이웃하는 청크들의 계수들 통계치들 중 적어도 하나에 기초하여 결정된다.
- [0271] 본 개시물의 또 다른 예에서, 유의도 맵 플래그는 첫번째로 코딩되며, 1-보다-큰 플래그는 두번째로 코딩되며, 2-보다-큰 플래그는 세번째로 코딩되며, 그리고 레벨 잔존 값은 2-보다-큰 플래그 이후에 코딩된다. 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그, 및 레벨 잔존 값은 스캔 순서에 따라서 청크에서의 변환 계수들에 대해 코딩된다.
- [0272] 본 개시물의 또 다른 예에서, 스캔 순서는 역 대각선 스캔 순서이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 16 개의 변환 계수의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛이다. 본 개시물의 또 다른 예에서, 청크는 스캔 순서에 따른 다수의 연속되는 변환 계수들이다. 본 개시물의 또 다른 예에서, 스캔 순서는 역 스캔 순서이다.
- [0273] 본 개시물의 또 다른 예에서, M1 의 값은 8 이다. 본 개시물의 또 다른 예에서, M1 의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.
- [0274] 본 개시물의 또 다른 예에서, 비디오 코더는 비디오 인코더이며, 비디오 인코더는 잔여 비디오 데이터를 발생하기 위해 픽셀 데이터를 인코딩하고; 변환 계수들의 청크를 발생하기 위해 잔여 비디오 데이터를 변환하고; 그리고 코딩된 유의도 맵 플래그, 코딩된 1-보다-큰 플래그, 및 코딩된 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 시그널링하도록 추가로 구성된다. 본 개시물의 또 다른 예에서, 비디오 코더는 M1 의 값을 인코딩된 비디오 비트스트림에서 시그널링하도록 추가로 구성된다. 본 개시물의 또 다른 예에서, M1 의 값은 시퀀스 파라미터 세트, 화상 파라미터 세트, 적응 파라미터 세트, 및 슬라이스 헤더 중 적어도 하나에서 시그널링된다.
- [0275] 본 개시물의 또 다른 예에서, 비디오 코더는 비디오 디코더이며, 비디오 디코더는 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 수신하고; 잔여 비디오 데이터를 발생하기 위해 변환 계수들의 청크를 역변환하고; 그리고 픽셀 데이터를 발생하기 위해 잔여 비디오 데이터를 디코딩하도록 추가로 구성된다.
- [0276] 본 개시물의 또 다른 예에서, 비디오 코더는 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 추가로 구성되며, 상기 N 은 청크에서의 변환 계수들의 최대 개수 미만이며, 본 방법은 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하는 단계를 더 포함하며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1 로서 갖는 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3 을 나타내며, 상기 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1 로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2 를 나타내며, 상기 레벨 잔존 값은 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1 을 나타낸다. 본 개시물의 또 다른 예에서, N 의 값은 1 이다.
- [0277] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하도록 구성된 장치는, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 수단; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 수단; 및 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하는 수단을 포함하며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 M1 은 청크에서의 변환 계수들의 최대 개수 미만이며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.
- [0278] 본 개시물의 또 다른 예에서, 컴퓨터-판독가능 저장 매체는, 실행될 때, 비디오 데이터를 코딩하도록 구성된 디바이스의 하나 이상의 프로세서들로 하여금, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로

표시된 청크에서의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 하는 명령들을 저장하고 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 M1은 청크에서의 변환 계수들의 최대 개수 미만이며, 상기 2-보다-큰 플래그는 특정의 변환 계수들이 2 보다 큰 절대값을 갖는지 여부를 표시한다.

[0279] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 방법은, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 단계; 유의도 맵 플래그에 의해 0 보다 큰 절대값을 갖는 것으로 표시된 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 단계; 및 1-보다-큰 플래그에 의해 1 보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하는 단계를 포함하며; 상기 유의도 맵 플래그는 특정의 변환 계수가 0 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2 보다 큰 절대값을 갖는지 여부를 표시하며, 상기 N은 청크에서의 변환 계수들의 최대 개수 미만이다.

[0280] 본 개시물의 또 다른 예에서, 본 방법은 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하는 단계를 더 포함하며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3을 나타내며, 상기 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2를 나타낸다.

[0281] 본 개시물의 또 다른 예에서, 본 방법은 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하는 단계를 더 포함하며, 상기 부호 플래그는 특정의 변환 계수의 부호를 표시한다.

[0282] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그는 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩된다.

[0283] 본 개시물의 또 다른 예에서, 본 방법은 레벨 잔존 값을 Golomb 파라미터를 이용하여 코딩하는 단계를 더 포함하며, 상기 Golomb 파라미터는 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 예측 모드, 칼라 성분 유형, 및 이웃하는 청크들의 계수들 통계치들 중 적어도 하나에 기초하여 결정된다.

[0284] 본 개시물의 또 다른 예에서, 유의도 맵 플래그는 첫번째로 코딩되며, 1-보다-큰 플래그는 두번째로 코딩되며, 2-보다-큰 플래그는 세번째로 코딩되며, 그리고 레벨 잔존 값은 2-보다-큰 플래그 이후에 코딩된다.

[0285] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그, 및 레벨 잔존 값의 각각은 스캔 순서에 따라서 청크에서의 변환 계수들에 대해 코딩된다.

[0286] 본 개시물의 또 다른 예에서, 스캔 순서는 역 대각선 스캔 순서이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 16 개의 변환 계수들의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛이다. 본 개시물의 또 다른 예에서, 청크는 스캔 순서에 따른 다수의 연속되는 변환 계수들이다. 본 개시물의 또 다른 예에서, 스캔 순서는 역 스캔 순서이다.

[0287] 본 개시물의 또 다른 예에서, N의 값은 1이다. 본 개시물의 또 다른 예에서, N의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.

[0288] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스는 비디오 인코딩 프로세스이며, 본 방법은 잔여 비디오 데이터를 발생하기 위해 픽셀 데이터를 인코딩하는 단계; 변환 계수들의 청크를 발생하기 위해 잔여 비디오 데이터를 변환하는 단계; 및 코딩된 유의도 맵 플래그, 코딩된 1-보다-큰 플래그, 및 코딩된 2-보다-큰 플래그를, 인코딩된 비디오 비트스트림에서 시그널링하는 단계를 더 포함한다. 본 개시물의 또 다른 예에서, 본 방법은 N의 값을 인코딩된 비디오 비트스트림에서 시그널링하는 단계를 더 포함한다. 본 개시물의 또 다른 예에서, N의 값을 시퀀스 파라미터 세트, 화상 파라미터 세트, 적응 파라미터 세트, 및 슬라이스 헤더 중 적어도 하나에서 시그널링된다.

[0289] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스는 비디오 디코딩 프로세스이며, 본 방법은 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 수신하는 단계; 잔여 비디오 데이터를 발생하기 위해 변환 계수들의 청크를 역변환하는 단계; 및 픽셀 데이터를 발생하기 위해 잔여 비디오 데

이터를 디코딩하는 단계를 더 포함한다.

[0290] 본 개시물의 또 다른 예에서, 1-보다-큰 플래그를 코딩하는 단계는, 비-제로인 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 단계를 포함하며, 상기 M1은 청크에서의 변환 계수들의 최대 개수 미만이며, 본 방법은 청크에서의 변환 계수들에 대해 레벨 잔존 값은 코딩하는 단계를 더 포함하며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1로서 갖는 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3을 나타내며, 상기 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2를 나타내며, 상기 레벨 잔존 값은 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1을 나타낸다. 본 개시물의 또 다른 예에서, M1의 값은 8이다.

[0291] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하도록 구성된 장치는, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0보다 큰 절대값을 갖는 것으로 표시된 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 구성된 비디오 코더를 포함하며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2보다 큰 절대값을 갖는지 여부를 표시하며, 상기 N은 청크에서의 변환 계수들의 최대 개수 미만이다.

[0292] 본 개시물의 또 다른 예에서, 비디오 코더는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성되며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1로서 갖는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 3을 나타내며, 상기 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2를 나타낸다.

[0293] 본 개시물의 또 다른 예에서, 비디오 코더는 청크에서의 변환 계수들에 대해 부호 플래그를 코딩하도록 추가로 구성되며, 상기 부호 플래그는 특정의 변환 계수의 부호를 표시한다.

[0294] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그는 적응 컨텍스트 모델과 함께 컨텍스트 적응 2진 산술 코딩 (CABAC) 을 이용하여 코딩되며, 상기 레벨 잔존 값은 CABAC 바이패스 모드에서 코딩되며, 상기 부호 플래그는 CABAC 바이패스 모드에서 코딩된다.

[0295] 본 개시물의 또 다른 예에서, 비디오 코더는 레벨 잔존 값을 Golomb 파라미터를 이용하여 코딩하도록 추가로 구성되며, 상기 Golomb 파라미터는 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 예측 모드, 칼라 성분 유형, 및 이웃하는 청크들의 계수들 통계치들 중 적어도 하나에 기초하여 결정된다.

[0296] 본 개시물의 또 다른 예에서, 유의도 맵 플래그는 첫번째로 코딩되며, 1-보다-큰 플래그는 두번째로 코딩되며, 2-보다-큰 플래그는 세번째로 코딩되며, 그리고 레벨 잔존 값은 2-보다-큰 플래그 이후에 코딩된다.

[0297] 본 개시물의 또 다른 예에서, 유의도 맵 플래그, 1-보다-큰 플래그, 2-보다-큰 플래그, 및 레벨 잔존 값의 각각은 스캔 순서에 따라서 청크에서의 변환 계수들에 대해 코딩된다. 본 개시물의 또 다른 예에서, 스캔 순서는 역 대각선 스캔 순서이다.

[0298] 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛의 16개의 변환 계수들의 서브블록이다. 본 개시물의 또 다른 예에서, 청크는 변환 유닛이다. 본 개시물의 또 다른 예에서, 청크는 스캔 순서에 따른 다수의 연속되는 변환 계수들이다. 본 개시물의 또 다른 예에서, 스캔 순서는 역 스캔 순서이다.

[0299] 본 개시물의 또 다른 예에서, N의 값은 1이다. 본 개시물의 또 다른 예에서, N의 값은 양자화 파라미터, 변환 유닛 사이즈, 변환 유닛 심도, 칼라 성분 유형, 변환 유닛에서의 청크의 위치, 청크에서의 최종 유의 계수의 존재, 및 이웃하는 청크의 계수 통계치들 중 적어도 하나에 기초한다.

[0300] 본 개시물의 또 다른 예에서, 비디오 코더는 비디오 인코더이며, 비디오 인코더는 잔여 비디오 데이터를 발생하기 위해 픽셀 데이터를 인코딩하고; 변환 계수들의 청크를 발생하기 위해 잔여 비디오 데이터를 변환하고; 그리고 코딩된 유의도 맵 플래그, 코딩된 1-보다-큰 플래그, 및 코딩된 2-보다-큰 플래그를, 인코딩된 비디오 비트 스트림에서 시그널링하도록 추가로 구성된다. 본 개시물의 또 다른 예에서, 비디오 코더는 N의 값을 인코

딩된 비디오 비트스트림에서 시그널링하도록 추가로 구성된다. 본 개시물의 또 다른 예에서, N의 값은 시퀀스 파라미터 세트, 화상 파라미터 세트, 적응 파라미터 세트, 및 슬라이스 헤더 중 적어도 하나에서 시그널링 된다.

[0301] 본 개시물의 또 다른 예에서, 비디오 코더는 비디오 디코더이며, 비디오 디코더는 유의도 맵 플래그, 1-보다-큰 플래그, 및 2-보다-큰 플래그를 인코딩된 비디오 비트스트림에서 수신하고; 잔여 비디오 데이터를 발생하기 위해 변환 계수들의 청크를 역변환하고; 그리고 픽셀 데이터를 발생하기 위해 잔여 비디오 데이터를 디코딩하도록 추가로 구성된다.

[0302] 본 개시물의 또 다른 예에서, 비디오 코더는, 비-제로인 청크에서의 처음 M1 개의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하도록 추가로 구성되며, 상기 M1은 청크에서의 변환 계수들의 최대 개수 미만이며, 본원에서 비디오 코더는 청크에서의 변환 계수들에 대해 레벨 잔존 값을 코딩하도록 추가로 구성되며, 상기 레벨 잔존 값은 코딩된 2-보다-큰 플래그를 1로서 갖는 청크에서의 변환 계수에 대해 대응하는 계수의 절대값 마이너스 3을 나타내며, 상기 레벨 잔존 값은 코딩된 1-보다-큰 플래그를 1로서 갖지만 2-보다-큰 플래그가 코딩되지 않은 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 2를 나타내며, 상기 레벨 잔존 값은 비-제로이지만 코딩된 1-보다-큰 플래그를 갖지 않는 청크에서의 변환 계수들에 대해 대응하는 계수의 절대값 마이너스 1을 나타낸다. 본 개시물의 또 다른 예에서, M1의 값은 8이다.

[0303] 본 개시물의 또 다른 예에서, 비디오 코딩 프로세스에서 변환 계수들을 코딩하도록 구성된 장치는, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하는 수단; 유의도 맵 플래그에 의해 0보다 큰 절대값을 갖는 것으로 표시된 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하는 수단; 및 1-보다-큰 플래그에 의해 1보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하는 수단을 포함하며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1보다 큰 절대값을 갖는지 여부를 표시하며, 상기 레벨 잔존 값은 청크에서의 변환 계수들의 최대 개수 미만이다.

[0304] 본 개시물의 또 다른 예에서, 컴퓨터-관독가능 저장 매체는, 실행될 때, 비디오 코딩 프로세스에서 변환 계수들을 코딩하도록 구성된 하나 이상의 프로세서들로 하여금, 변환 계수들의 청크에서의 변환 계수들에 대해 유의도 맵 플래그를 코딩하고; 유의도 맵 플래그에 의해 0보다 큰 절대값을 갖는 것으로 표시된 변환 계수들의 청크에서의 변환 계수들에 대해 1-보다-큰 플래그를 코딩하고; 그리고 1-보다-큰 플래그에 의해 1보다 큰 절대값을 갖는 것으로 표시된 청크에서의 처음 N개의 변환 계수들에 대해 2-보다-큰 플래그를 코딩하도록 하는 명령들을 저장하고 있으며, 상기 유의도 맵 플래그는 특정의 변환 계수가 0보다 큰 절대값을 갖는지 여부를 표시하며, 상기 1-보다-큰 플래그는 특정의 변환 계수가 1보다 큰 절대값을 갖는지 여부를 표시하며, 상기 2-보다-큰 플래그는 특정의 변환 계수가 2보다 큰 절대값을 갖는지 여부를 표시하며, 상기 N은 청크에서의 변환 계수들의 최대 개수 미만이다.

[0305] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 그 기능들은 하나 이상의 명령들 또는 코드로서, 컴퓨터-관독가능 매체 상에 저장되거나 또는 컴퓨터-관독가능 매체를 통해서 송신될 수도 있으며, 하드웨어-기반의 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 관독가능 매체들은 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터 관독가능 저장 매체들, 또는 예를 들어, 통신 프로토콜에 따라 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이와 같이, 컴퓨터-관독가능 매체들은 일반적으로 (1) 비일시성 유형의 컴퓨터-관독가능 저장 매체 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시물에서 설명하는 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 제품은 컴퓨터-관독가능 매체를 포함할 수도 있다.

[0306] 비제한적인 예로서, 이런 컴퓨터-관독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광디스크 스토리지, 자기디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터-관독가능 매체로 적절히 지정된다. 예를 들어, 동축 케이블, 광섬유 케이블, 연선, 디지털 가입자 회선 (DSL), 또는 무선 기술들 예컨대 적외선, 무선, 및 마

이크로파를 이용하여 명령들이 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되는 경우, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 무선 기술들, 예컨대 적외선, 무선, 및 마이크로파가 그 매체의 정의에 포함된다.

그러나, 컴퓨터-판독가능 저장 매체 및 데이터 저장 매체는 접속부들, 캐리어 파들, 신호들, 또는 다른 일시성 매체를 포함하지 않고, 그 대신, 비-일시성 유형의 저장 매체로 송신되는 것으로 해석되어야 한다. 디스크 (disk) 및 디스크 (disc) 는, 본원에서 사용된 바와 같이, 컴팩트 디스크 (CD), 레이저 디스크, 광 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루-레이 디스크를 포함하며, 디스크들 (disks) 은 데이터를 자기적으로 보통 재생하지만, 디스크들 (discs) 은 레이저로 데이터를 광학적으로 재생한다. 앞에서 언급한 것들의 조합들이 또한 컴퓨터-판독가능 매체들의 범위 내에 포함되어야 한다.

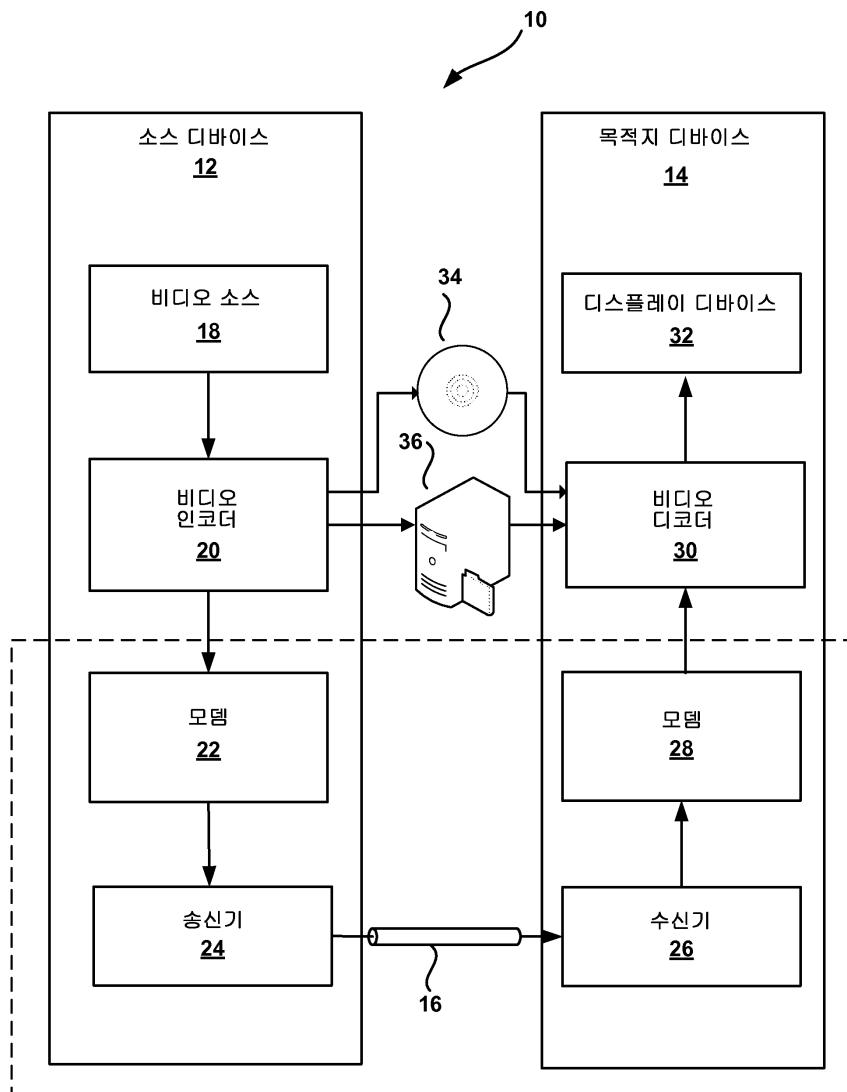
[0307] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 등가의 통합 또는 이산 로직 회로와 같은, 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 용어 "프로세서" 는, 본원에서 사용된 바와 같이, 전술한 구조 중 임의의 구조 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능은 전용 하드웨어 및/또는 인코딩 및 디코딩을 위해 구성되는 소프트웨어 모듈들 내에 제공되거나, 또는 결합된 코덱에 포함될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 앤리먼트들로 전적으로 구현될 수 있다.

[0308] 본 개시물의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예컨대, 칩 세트) 를 포함한, 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 개시한 기법들을 수행하도록 구성되는 디바이스들의 기능적 양태들을 강조하기 위해서 여러 구성요소들, 모듈들, 또는 유닛들이 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다. 오히려, 위에서 설명한 바와 같이, 여러 유닛들이 코덱 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명한 바와 같은 하나 이상의 프로세서들을 포함한, 상호작용하는 하드웨어 유닛들의 컬렉션으로 제공될 수도 있다.

[0309] 여러 예들이 설명되었다. 이를 및 다른 예들은 다음 청구항들의 범위 이내이다.

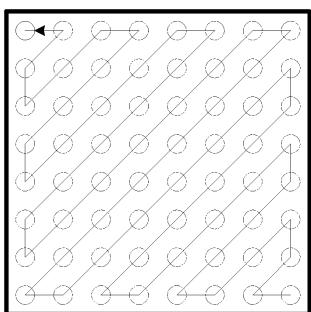
도면

도면1

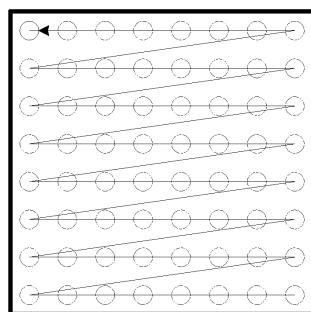


도면2

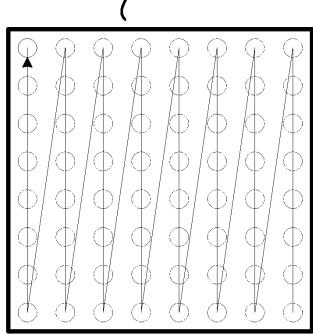
29



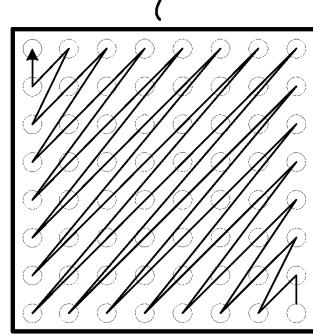
33



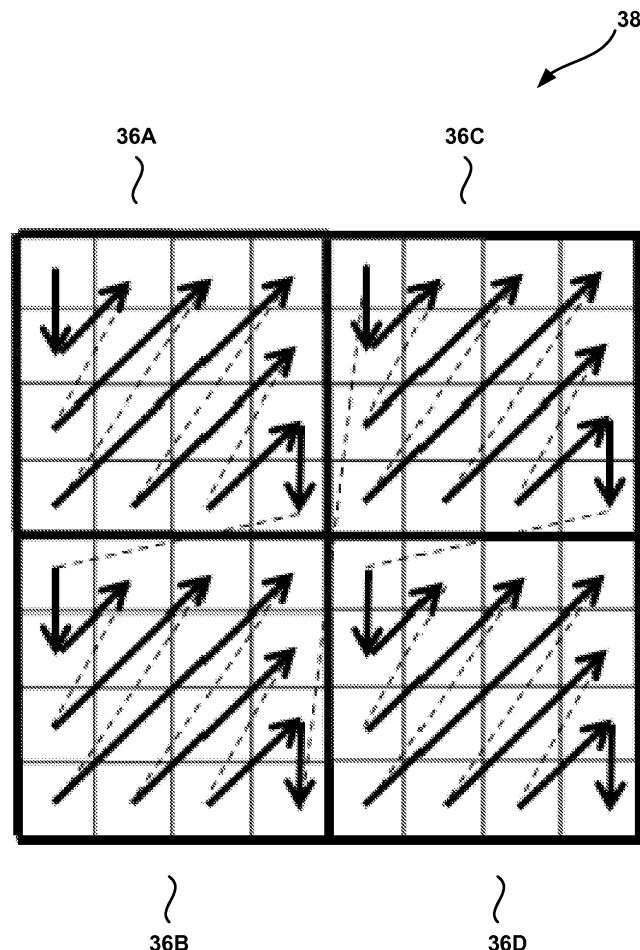
31



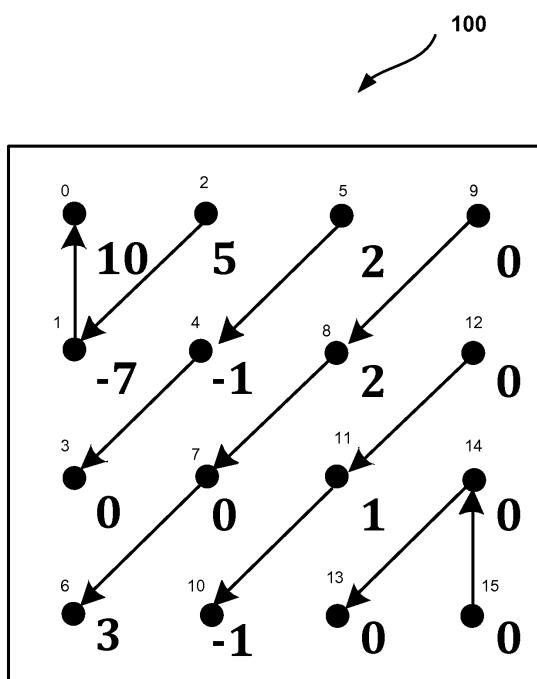
35



도면3



도면4



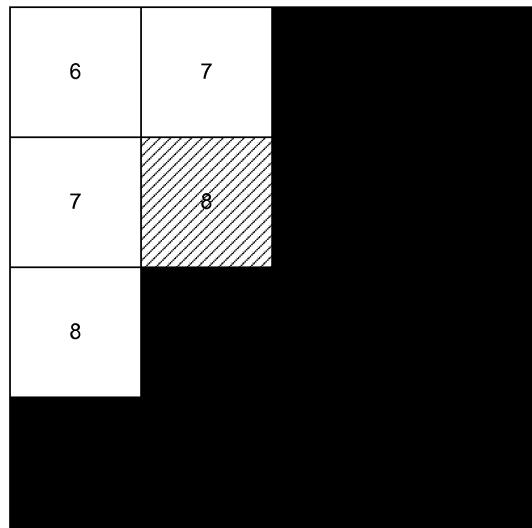
도면5

1 (X=0, Y=0)	2 (X=1, Y=0)	3 (X=2, Y=0)	4 (X=3, Y=0)
2 (X=0, Y=1)	3 (X=1, Y=1)	4 (X=2, Y=1)	5 (X=3, Y=1)
3 (X=0, Y=2)	4 (X=1, Y=2)	5 (X=2, Y=2)	6 (X=3, Y=2)
4 (X=0, Y=3)	5 (X=1, Y=3)	6 (X=2, Y=3)	7 (X=3, Y=3)

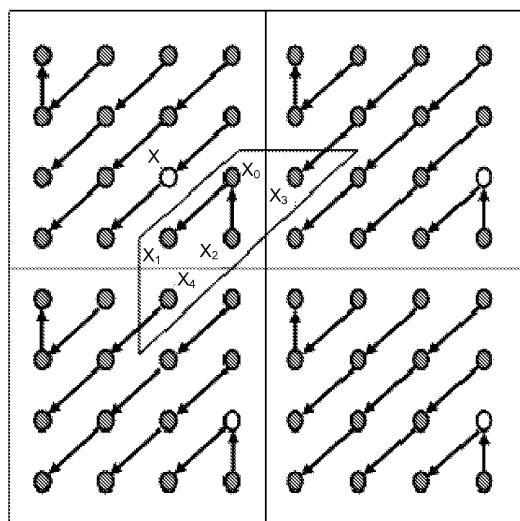
도면6

4	5	6	7
5	6	7	
6	7	8	
7	8		

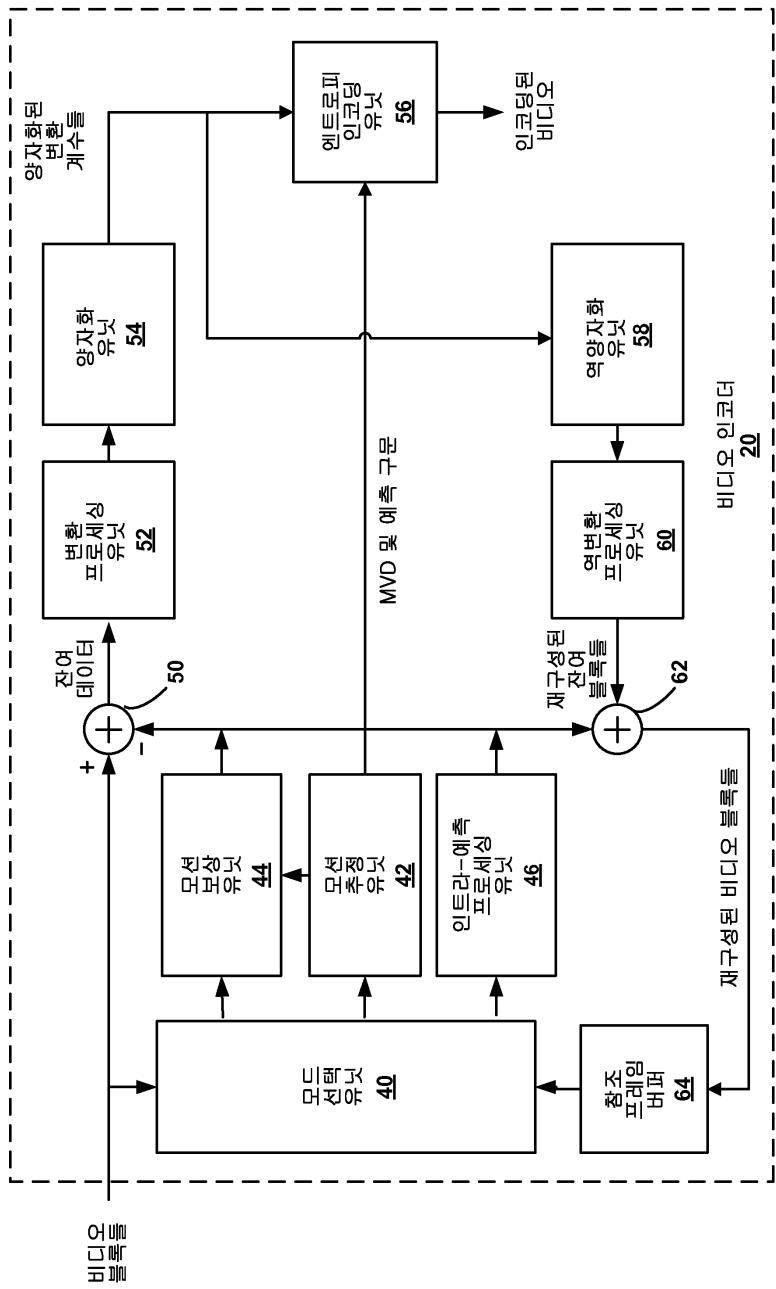
도면7



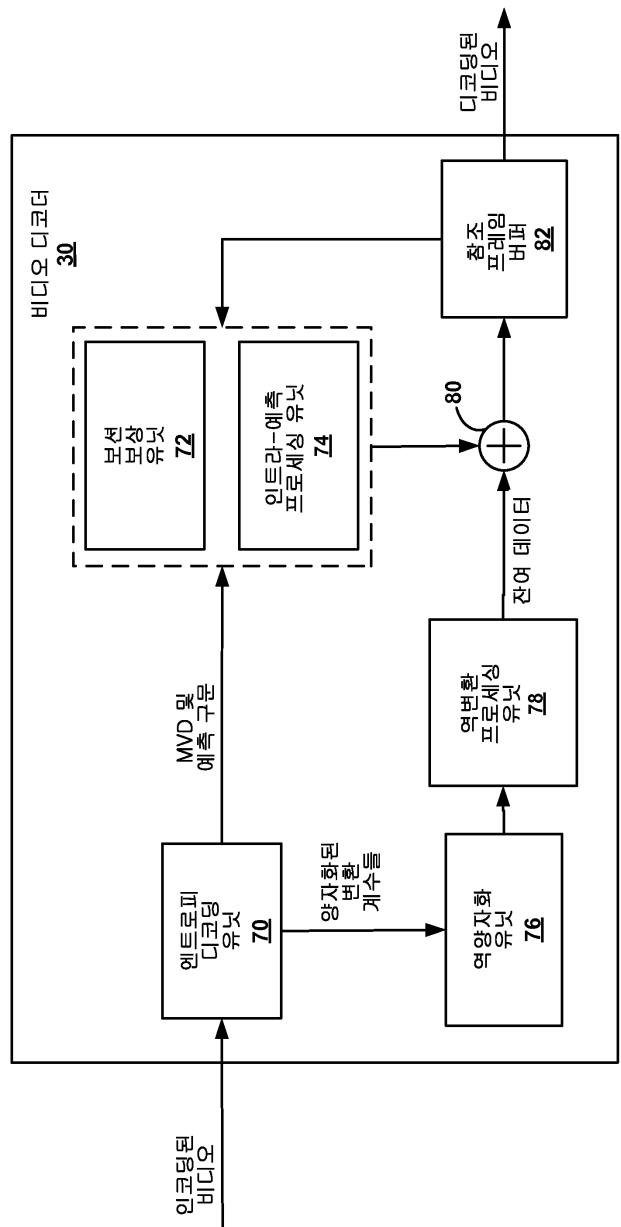
도면8



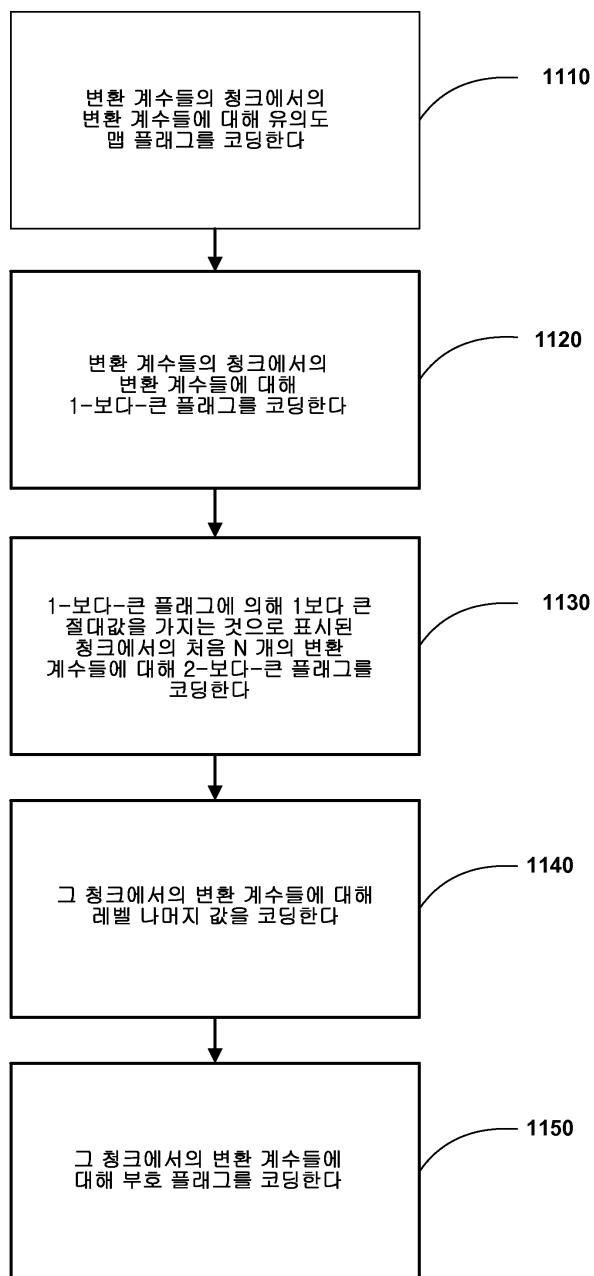
도면9



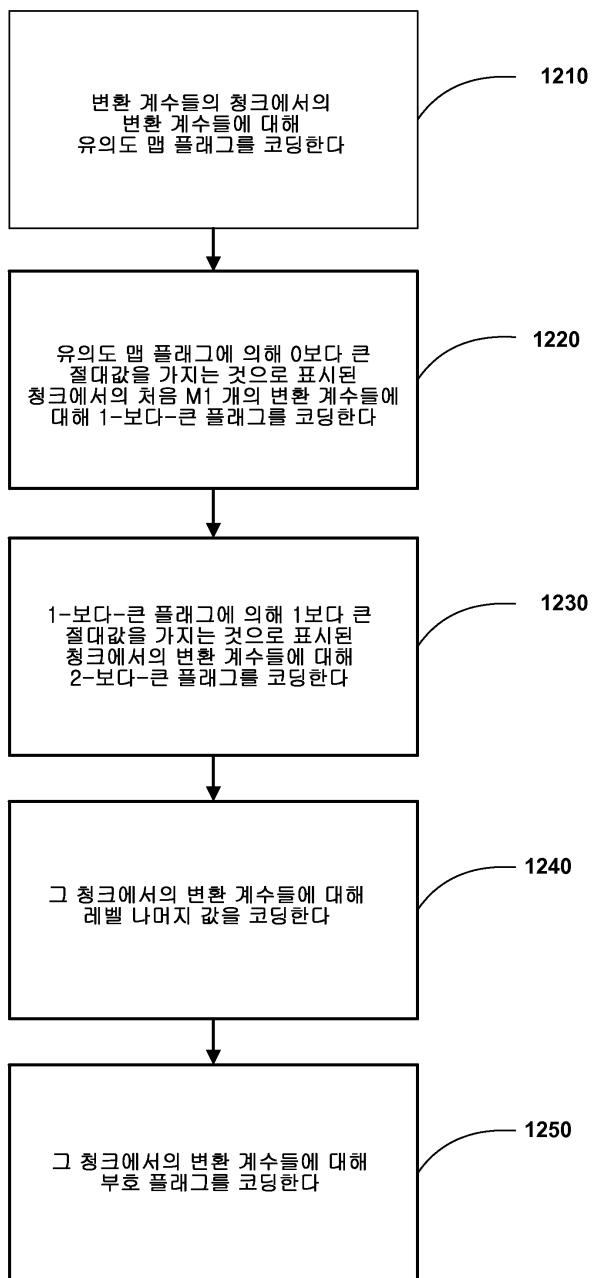
도면10



도면11



도면12



도면13

