



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2017년12월19일
(11) 등록번호 10-1810757
(24) 등록일자 2017년12월13일

- (51) 국제특허분류(Int. Cl.)
H04N 19/593 (2014.01) H04N 13/00 (2016.01)
H04N 19/463 (2014.01) H04N 19/597 (2014.01)
H04N 19/70 (2014.01)
- (52) CPC특허분류
H04N 19/593 (2015.01)
H04N 13/0048 (2013.01)
- (21) 출원번호 10-2015-7030268
- (22) 출원일자(국제) 2014년03월27일
심사청구일자 2017년01월05일
- (85) 번역문제출일자 2015년10월20일
- (65) 공개번호 10-2015-0135398
- (43) 공개일자 2015년12월02일
- (86) 국제출원번호 PCT/US2014/032050
- (87) 국제공개번호 WO 2014/160880
국제공개일자 2014년10월02일
- (30) 우선권주장
61/805,901 2013년03월27일 미국(US)
(뒷면에 계속)
- (56) 선행기술조사문헌
“3D-HEVC Test Model 3”, JCT-3V meeting,
<http://phenix.it-sudparis.eu/jct3v/>, no.
JCT3V-C1005 (2013.03.15.)
“3D-HEVC Test Model 2”, JCT-3V meeting,
<http://phenix.it-sudparis.eu/jct3v/>, no.
JCT3V-B1005 (2012.12.23.)

- (73) 특허권자
켈컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 발명자
장 리
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
천 잉
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
자오 신
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
- (74) 대리인
특허법인코리아나

전체 청구항 수 : 총 47 항

심사관 : 장석환

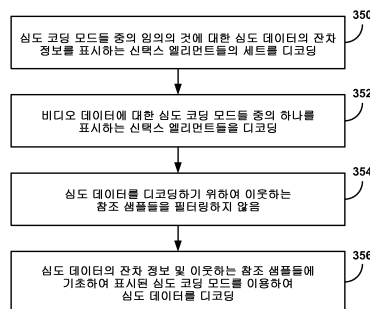
(54) 발명의 명칭 **3D-HEVC 를 위한 심도 데이터의 심도 코딩 모드들 시그널링**

(57) 요약

고효율 비디오 코딩 (HEVC) 코딩 모드들과는 상이한 심도 코딩 모드들을 이용하여 멀티뷰 플러스 심도 포맷으로 표현된 3 차원 (3D) 비디오 데이터에 대한 심도 데이터를 인코딩하고 디코딩하기 위한 기법들이 설명된다. 3D-HEVC 프로세스에서 이용가능한 추가적인 심도 인프라 코딩 모드들의 예들은 심도 모델링 모드 (Depth

(뒷면에 계속)

대표도 - 도13



Modeling Mode; DMM), 간략화된 심도 코딩 (Simplified Depth Coding; SDC) 모드, 및 체인 코딩 모드 (Chain Coding Mode; CCM) 중의 적어도 2 개를 포함한다. 게다가, 추가적인 심도 인터 코딩 모드의 예는 인터 SDC 모드를 포함한다. 하나의 예에서, 기법들은 HEVC 신택스로부터 분리되어 있는 심도 모델링 테이블에서의 3D 비디오 데이터에 대한 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 시그널링하는 것을 포함한다. 또 다른 예에서, 이 개시물의 기법들은 심도 코딩 모드들 중의 2 개 이상에 걸쳐 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보의 시그널링을 통합하는 것을 포함한다.

(52) CPC특허분류

HO4N 19/463 (2015.01)

HO4N 19/597 (2015.01)

HO4N 19/70 (2015.01)

(30) 우선권주장

61/805,922 2013년03월27일 미국(US)

61/811,103 2013년04월11일 미국(US)

61/890,629 2013년10월14일 미국(US)

61/891,816 2013년10월16일 미국(US)

14/226,012 2014년03월26일 미국(US)

명세서

청구범위

청구항 1

비디오 데이터를 디코딩하는 방법으로서,

복수의 심도 코딩 모드들 중 임의의 것을 사용하여 코딩된 상기 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 선택스 엘리먼트들의 통일된 세트를 디코딩하는 단계로서, 상기 선택스 엘리먼트들의 통일된 세트는 상기 복수의 심도 코딩 모드들의 전부에 대하여 사용된 선택스 엘리먼트들의 동일한 세트를 포함하고, 상기 심도 코딩 모드들의 각각은 대응하는 텍스처 데이터에 대한 심도 데이터를 포함하는 하나 이상의 심도 맵들의 코딩을 위해 사용될 수 있고, 상기 복수의 심도 코딩 모드들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 또는 체인 코딩 모드 (CCM) 중 적어도 2 개를 포함하는 복수의 심도 인트라 코딩 모드들을 포함하는, 상기 선택스 엘리먼트들의 통일된 세트를 디코딩하는 단계;

상기 비디오 데이터에 대한 상기 복수의 심도 코딩 모드들 중 하나를 표시하는 하나 이상의 선택스 엘리먼트들을 디코딩하는 단계; 및

상기 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드를 사용하여 상기 비디오 데이터에 대한 심도 데이터를 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 선택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시하는, 비디오 데이터를 디코딩하는 방법.

청구항 3

제 1 항에 있어서,

상기 선택스 엘리먼트들의 통일된 세트는 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 선택스 엘리먼트를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 4

제 3 항에 있어서,

상기 심도 데이터에 대한 코딩 파티션들의 개수에 기초하여 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 상기 선택스 엘리먼트에 대한 컨텍스트 모델을 선택하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 5

제 1 항에 있어서,

상기 선택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 잔차 값들의 부호 및 절대값을 표시하는 선택스 엘리먼트들을 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 6

제 5 항에 있어서,

단일의 이진화 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 심도 데이터의 상기 잔차 값들의 상기 부호 및 절대값을 표시하는 상기 선택스 엘리먼트들을 탈-이진화하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 7

제 1 항에 있어서,

단일 컨텍스트 모델 선택 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 적용된 컨텍스트 모델들의 동일 세트로부터 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 신택스 엘리먼트들의 통일된 세트에 대한 컨텍스트 모델들을 선택하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 8

제 1 항에 있어서,

상기 심도 데이터를 디코딩하는 단계는 상기 이웃하는 참조 샘플들을 필터링하지 않고 상기 잔차 정보 및 상기 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드에 따라 상기 비디오 데이터의 심도 값들을 재구성하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 9

제 8 항에 있어서,

표시된 상기 심도 코딩 모드는 상기 복수의 심도 인트라 코딩 모드들의 심도 인트라 코딩 모드들 중 하나이고, 상기 이웃하는 참조 샘플들의 필터링은 상기 DMM, 상기 CCM, 상기 SDC 모드의 서브-모드 SDCM_DMM_1, 또는 상기 SDC 모드의 서브-모드 SDCM_DC 중 적어도 하나에 대하여 디스에이블되는, 비디오 데이터를 디코딩하는 방법.

청구항 10

제 1 항에 있어서,

상기 복수의 심도 코딩 모드들은 고효율 비디오 코딩 (HEVC) 인트라 코딩 모드들과는 상이한 복수의 심도 인트라 코딩 모드들을 더 포함하고,

표시된 상기 심도 코딩 모드는 상기 HEVC 인트라 코딩 모드들과는 상이한 상기 복수의 심도 인트라 코딩 모드들 중 하나이고,

상기 심도 코딩 모드들 중 하나를 표시하는 상기 하나 이상의 신택스 엘리먼트들을 디코딩하는 단계는,

상기 HEVC 인트라 코딩 모드들과는 상이한 상기 복수의 심도 인트라 코딩 모드들 중의 임의의 것이 상기 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 신택스 엘리먼트를 디코딩하는 단계;

상기 HEVC 인트라 코딩 모드들과는 상이한 상기 심도 인트라 코딩 모드들 중의 임의의 것이 인에이블되는 것을 표시하는 상기 신택스 엘리먼트를 디코딩한 후, 상기 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 디코딩하는 단계로서, 상기 심도 모델링 테이블은 상기 HEVC 인트라 코딩 모드들에 대한 신택스 엘리먼트들로부터 분리되어 있는 상기 심도 인트라 코딩 모드들에 대한 신택스 엘리먼트들을 포함하는, 상기 심도 모델링 테이블을 디코딩하는 단계; 및

상기 심도 모델링 테이블에서의 상기 하나 이상의 신택스 엘리먼트들에 기초하여 상기 HEVC 인트라 코딩 모드들과는 상이한 상기 심도 인트라 코딩 모드들 중의 하나를 선택하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 11

비디오 디코딩 디바이스로서,

비디오 데이터를 저장하도록 구성된 메모리; 및

하나 이상의 프로세서들을 포함하며,

상기 하나 이상의 프로세서들은,

복수의 심도 코딩 모드들 중 임의의 것을 사용하여 코딩된 상기 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 통일된 세트를 디코딩하는 것으로서, 상기 신택스 엘리먼트들의 통일된 세트는 상기 복수의 심도 코딩 모드들의 전부에 대하여 사용된 신택스 엘리먼트들의 동일한 세트를 포함하고, 상기

심도 코딩 모드들의 각각은 대응하는 텍스처 데이터에 대한 심도 데이터를 포함하는 하나 이상의 심도 맵들의 코딩을 위해 사용될 수 있고, 상기 복수의 심도 코딩 모드들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 또는 체인 코딩 모드 (CCM) 중 적어도 2 개를 포함하는 복수의 심도 인트라 코딩 모드들을 포함하는, 상기 선택스 엘리먼트들의 통일된 세트를 디코딩하고;

상기 비디오 데이터에 대한 상기 복수의 심도 코딩 모드들 중 하나를 표시하는 하나 이상의 선택스 엘리먼트들을 디코딩하고; 그리고

상기 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드를 사용하여 상기 비디오 데이터에 대한 심도 데이터를 디코딩하도록

구성되는, 비디오 디코딩 디바이스.

청구항 12

제 11 항에 있어서,

상기 선택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시하는, 비디오 디코딩 디바이스.

청구항 13

제 11 항에 있어서,

상기 선택스 엘리먼트들의 통일된 세트는 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 선택스 엘리먼트를 포함하는, 비디오 디코딩 디바이스.

청구항 14

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 상기 심도 데이터에 대한 코딩 파티션들의 개수에 기초하여 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 상기 선택스 엘리먼트에 대한 컨텍스트 모델을 선택하도록 구성되는, 비디오 디코딩 디바이스.

청구항 15

제 11 항에 있어서,

상기 선택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 심도 데이터의 잔차 값들의 부호 및 절대값을 표시하는 선택스 엘리먼트들을 포함하는, 비디오 디코딩 디바이스.

청구항 16

제 15 항에 있어서,

상기 하나 이상의 프로세서들은 단일의 이진화 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 심도 데이터의 상기 잔차 값들의 상기 부호 및 절대값을 표시하는 상기 선택스 엘리먼트들을 탈-이진화하도록 구성되는, 비디오 디코딩 디바이스.

청구항 17

제 11 항에 있어서,

상기 하나 이상의 프로세서들은 단일 컨텍스트 모델 선택 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 적용된 컨텍스트 모델들의 동일 세트로부터 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 선택스 엘리먼트들의 통일된 세트에 대한 컨텍스트 모델들을 선택하도록 구성되는, 비디오 디코딩 디바이스.

청구항 18

제 11 항에 있어서,

상기 하나 이상의 프로세서들은 상기 이웃하는 참조 샘플들을 필터링하지 않고 상기 잔차 정보 및 상기 이웃하

는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드에 따라 상기 비디오 데이터의 심도 값들을 재구성하도록 구성되는, 비디오 디코딩 디바이스.

청구항 19

제 18 항에 있어서,

표시된 상기 심도 코딩 모드는 상기 복수의 심도 인트라 코딩 모드들의 심도 인트라 코딩 모드들 중 하나이고, 상기 이웃하는 참조 샘플들의 필터링은 상기 DMM, 상기 CCM, 상기 SDC 모드의 서브-모드 SDCM_DMM_1, 또는 상기 SDC 모드의 서브-모드 SDCM_DC 중 적어도 하나에 대하여 디스에이블되는, 비디오 디코딩 디바이스.

청구항 20

제 11 항에 있어서,

상기 복수의 심도 코딩 모드들은 고효율 비디오 코딩 (HEVC) 인트라 코딩 모드들과는 상이한 복수의 심도 인트라 코딩 모드들을 더 포함하고,

표시된 상기 심도 코딩 모드는 상기 HEVC 인트라 코딩 모드들과는 상이한 복수의 심도 인트라 코딩 모드들 중 하나이고,

심도 모드들 중 하나를 표시하는 상기 하나 이상의 신택스 엘리먼트들을 디코딩하기 위해, 상기 하나 이상의 프로세서들은,

상기 HEVC 인트라 코딩 모드들과는 상이한 상기 복수의 심도 인트라 코딩 모드들 중의 임의의 것이 상기 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 신택스 엘리먼트를 디코딩하고;

상기 HEVC 인트라 코딩 모드들과는 상이한 상기 심도 인트라 코딩 모드들 중의 임의의 것이 인에이블되는 것을 표시하는 상기 신택스 엘리먼트를 디코딩한 후, 상기 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 디코딩하는 것으로서, 상기 심도 모델링 테이블은 상기 HEVC 인트라 코딩 모드들에 대한 신택스 엘리먼트들로부터 분리되어 있는 상기 심도 인트라 코딩 모드들에 대한 신택스 엘리먼트들을 포함하는, 상기 심도 모델링 테이블을 디코딩하고; 그리고

상기 심도 모델링 테이블에서의 상기 하나 이상의 신택스 엘리먼트들에 기초하여 상기 심도 인트라 코딩 모드들 중의 하나를 선택하도록

구성되는, 비디오 디코딩 디바이스.

청구항 21

비디오 디코딩 디바이스로서,

복수의 심도 코딩 모드들 중 임의의 것을 사용하여 코딩된 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 통일된 세트를 디코딩하기 위한 수단으로서, 상기 신택스 엘리먼트들의 통일된 세트는 상기 복수의 심도 코딩 모드들의 전부에 대하여 사용된 신택스 엘리먼트들의 동일한 세트를 포함하고, 상기 심도 코딩 모드들의 각각은 대응하는 텍스처 데이터에 대한 심도 데이터를 포함하는 하나 이상의 심도 맵들의 코딩을 위해 사용될 수 있고, 상기 복수의 심도 코딩 모드들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 또는 체인 코딩 모드 (CCM) 중 적어도 2 개를 포함하는 복수의 심도 인트라 코딩 모드들을 포함하는, 상기 신택스 엘리먼트들의 통일된 세트를 디코딩하기 위한 수단;

상기 비디오 데이터에 대한 상기 복수의 심도 코딩 모드들 중 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 디코딩하기 위한 수단; 및

상기 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드를 사용하여 상기 비디오 데이터에 대한 심도 데이터를 디코딩하기 위한 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 22

명령들을 포함하는 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 비디오 데이터를 디코딩하기 위한 디바이스의 하나 이상의 프로세서들로 하여금,

복수의 심도 코딩 모드들 중 임의의 것을 사용하여 코딩된 상기 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 통일된 세트를 디코딩하게 하는 것으로서, 상기 신택스 엘리먼트들의 통일된 세트는 상기 복수의 심도 코딩 모드들의 전부에 대하여 사용된 신택스 엘리먼트들의 동일한 세트를 포함하고, 상기 심도 코딩 모드들의 각각은 대응하는 텍스처 데이터에 대한 심도 데이터를 포함하는 하나 이상의 심도 맵들의 코딩을 위해 사용될 수 있고, 상기 복수의 심도 코딩 모드들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 또는 체인 코딩 모드 (CCM) 중 적어도 2 개를 포함하는 복수의 심도 인트라 코딩 모드들을 포함하는, 상기 신택스 엘리먼트들의 통일된 세트를 디코딩하게 하고;

상기 비디오 데이터에 대한 상기 복수의 심도 코딩 모드들 중 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 디코딩하게 하고; 그리고

상기 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드를 사용하여 상기 비디오 데이터에 대한 심도 데이터를 디코딩하게 하는, 컴퓨터 판독가능 저장 매체.

청구항 23

비디오 데이터를 인코딩하는 방법으로서,

복수의 심도 코딩 모드들 중 선택된 심도 코딩 모드를 사용하여 상기 비디오 데이터에 대한 심도 데이터를 인코딩하는 단계로서, 상기 심도 데이터를 인코딩하는 것은 상기 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 상기 심도 데이터의 잔차 정보를 생성하는 것을 포함하는, 상기 심도 데이터를 인코딩하는 단계;

상기 비디오 데이터에 대하여 상기 선택된 심도 코딩 모드를 표시하는 하나 이상의 신택스 엘리먼트들을 인코딩하는 단계; 및

상기 복수의 심도 코딩 모드들 중 임의의 것을 사용하여 인코딩된 상기 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 통일된 세트를 인코딩하는 단계로서, 상기 신택스 엘리먼트들의 통일된 세트는 상기 복수의 심도 코딩 모드들의 전부에 대하여 사용된 신택스 엘리먼트들의 동일한 세트를 포함하고, 상기 심도 코딩 모드들의 각각은 대응하는 텍스처 데이터에 대한 심도 데이터를 포함하는 하나 이상의 심도 맵들의 코딩을 위해 사용될 수 있고, 상기 복수의 심도 코딩 모드들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 또는 체인 코딩 모드 (CCM) 중 적어도 2 개를 포함하는 복수의 심도 인트라 코딩 모드들을 포함하는, 상기 신택스 엘리먼트들의 통일된 세트를 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 24

제 23 항에 있어서,

상기 신택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시하는, 비디오 데이터를 인코딩하는 방법.

청구항 25

제 23 항에 있어서,

상기 신택스 엘리먼트들의 통일된 세트는 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 신택스 엘리먼트를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 26

제 25 항에 있어서,

상기 심도 데이터에 대한 코딩 파티션들의 개수에 기초하여 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 상기 신택스 엘리먼트에 대한 컨텍스트 모델을 선택하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 27

제 23 항에 있어서,

상기 신택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 잔차 값들의 부호 및 절대값을 표시하는 신택스 엘리먼트들을 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 28

제 27 항에 있어서,

단일의 이진화 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 심도 데이터의 상기 잔차 값들의 상기 부호 및 절대값을 표시하는 상기 신택스 엘리먼트들을 이진화하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 29

제 23 항에 있어서,

단일 컨텍스트 모델 선택 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 적용된 컨텍스트 모델들의 동일 세트로부터 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 신택스 엘리먼트들의 통일된 세트에 대한 컨텍스트 모델들을 선택하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 30

제 23 항에 있어서,

상기 심도 데이터를 인코딩하는 단계는 상기 이웃하는 참조 샘플들을 필터링하지 않고 상기 비디오 데이터의 원래의 심도 값들 및 상기 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드에 따라 상기 심도 데이터의 잔차 값들을 생성하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 31

제 30 항에 있어서,

표시된 상기 심도 코딩 모드는 상기 복수의 심도 인트라 코딩 모드들의 심도 인트라 코딩 모드들 중 하나이고, 상기 이웃하는 참조 샘플들의 필터링은 상기 DMM, 상기 CCM, 상기 SDC 모드의 서브-모드 SDCM_DMM_1, 또는 상기 SDC 모드의 서브-모드 SDCM_DC 중 적어도 하나에 대하여 디스에이블되는, 비디오 데이터를 인코딩하는 방법.

청구항 32

제 23 항에 있어서,

상기 복수의 심도 코딩 모드들은 고효율 비디오 코딩 (HEVC) 인트라 코딩 모드들과는 상이한 복수의 심도 인트라 코딩 모드들을 더 포함하고,

상기 선택된 심도 코딩 모드는 상기 HEVC 인트라 코딩 모드들과는 상이한 상기 복수의 심도 인트라 코딩 모드들 중 하나이고,

선택된 심도 인트라 모드를 표시하는 상기 하나 이상의 신택스 엘리먼트들을 인코딩하는 단계는,

상기 심도 인트라 코딩 모드들이 상기 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 인코딩하는 단계; 및

상기 심도 인트라 코딩 모드들 중 선택된 심도 인트라 코딩 모드를 표시하는 상기 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 인코딩하는 단계를 포함하며,

상기 심도 모델링 테이블은 상기 HEVC 인트라 코딩 모드들에 대한 신택스 엘리먼트들로부터 분리되어 있는 상기 심도 인트라 코딩 모드들에 대한 신택스 엘리먼트들을 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 33

비디오 인코딩 디바이스로서,

비디오 데이터를 저장하도록 구성된 메모리; 및

하나 이상의 프로세서들을 포함하며,

상기 하나 이상의 프로세서들은,

상기 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 심도 데이터의 잔차 정보를 생성하기 위해, 복수의 심도 코딩 모드들 중 선택된 심도 코딩 모드를 사용하여 상기 비디오 데이터에 대한 심도 데이터를 인코딩하고;

상기 비디오 데이터에 대하여 상기 선택된 심도 코딩 모드를 표시하는 하나 이상의 신택스 엘리먼트들을 인코딩하고; 그리고

상기 복수의 심도 코딩 모드들 중 임의의 것을 사용하여 인코딩된 상기 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 통일된 세트를 인코딩하는 것으로서, 상기 신택스 엘리먼트들의 통일된 세트는 상기 복수의 심도 코딩 모드들의 전부에 대하여 사용된 신택스 엘리먼트들의 동일한 세트를 포함하고, 상기 심도 코딩 모드들의 각각은 대응하는 텍스처 데이터에 대한 심도 데이터를 포함하는 하나 이상의 심도 맵들의 코딩을 위해 사용될 수 있고, 상기 복수의 심도 코딩 모드들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 또는 체인 코딩 모드 (CCM) 중 적어도 2 개를 포함하는 복수의 심도 인트라 코딩 모드들을 포함하는, 상기 신택스 엘리먼트들의 통일된 세트를 인코딩하도록

구성되는, 비디오 인코딩 디바이스.

청구항 34

제 33 항에 있어서,

상기 신택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시하는, 비디오 인코딩 디바이스.

청구항 35

제 33 항에 있어서,

상기 신택스 엘리먼트들의 통일된 세트는 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 신택스 엘리먼트를 포함하는, 비디오 인코딩 디바이스.

청구항 36

제 35 항에 있어서,

상기 하나 이상의 프로세서들은 상기 심도 데이터에 대한 코딩 파티션들의 개수에 기초하여 상기 심도 데이터의 잔차 값들이 상기 심도 코딩 모드들 중 임의의 것에 대하여 존재하는지 여부를 표시하는 상기 신택스 엘리먼트에 대한 컨텍스트 모델을 선택하도록 구성되는, 비디오 인코딩 디바이스.

청구항 37

제 33 항에 있어서,

상기 신택스 엘리먼트들의 통일된 세트는 상기 심도 코딩 모드들 중 임의의 것에 대한 심도 데이터의 잔차 값들의 부호 및 절대값을 표시하는 신택스 엘리먼트들을 포함하는, 비디오 인코딩 디바이스.

청구항 38

제 37 항에 있어서,

상기 하나 이상의 프로세서들은 단일의 이진화 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 심도 데이터의 상기 잔차 값들의 상기 부호 및 절대값을 표시하는 상기 신택스 엘리먼트들을 이진화하도록 구성되는, 비디오 인코딩 디바이스.

청구항 39

제 33 항에 있어서,

상기 하나 이상의 프로세서들은 단일 컨텍스트 모델 선택 방법을 사용하여 상기 심도 코딩 모드들 중 임의의 것에 적용된 컨텍스트 모델들의 동일 세트로부터 상기 심도 코딩 모드들 중 임의의 것에 대한 상기 선택스 엘리먼트들의 통일된 세트에 대한 컨텍스트 모델들을 선택하도록 구성되는, 비디오 인코딩 디바이스.

청구항 40

제 33 항에 있어서,

상기 하나 이상의 프로세서들은 상기 이웃하는 참조 샘플들을 필터링하지 않고 상기 비디오 데이터의 원래의 심도 값들 및 상기 이웃하는 참조 샘플들에 기초하여 표시된 상기 심도 코딩 모드에 따라 상기 심도 데이터의 잔차 값들을 생성하도록 구성되는, 비디오 인코딩 디바이스.

청구항 41

제 40 항에 있어서,

표시된 상기 심도 코딩 모드는 상기 복수의 심도 인트라 코딩 모드들의 심도 인트라 코딩 모드들 중 하나이고, 상기 이웃하는 참조 샘플들의 필터링은 상기 DMM, 상기 CCM, 상기 SDC 모드의 서브-모드 SDCM_DMM_1, 또는 상기 SDC 모드의 서브-모드 SDCM_DC 중 적어도 하나에 대하여 디스에이블되는, 비디오 인코딩 디바이스.

청구항 42

제 33 항에 있어서,

상기 복수의 심도 코딩 모드들은 고효율 비디오 코딩 (HEVC) 인트라 코딩 모드들과는 상이한 복수의 심도 인트라 코딩 모드들을 더 포함하고,

상기 선택된 심도 코딩 모드는 상기 HEVC 인트라 코딩 모드들과는 상이한 상기 복수의 심도 인트라 코딩 모드들 중 하나이고,

상기 하나 이상의 프로세서들은,

상기 심도 인트라 코딩 모드들이 상기 비디오 데이터에 대해 인에이블되는 것을 표시하는 선택스 엘리먼트를 인코딩하고; 그리고

상기 심도 인트라 코딩 모드들 중 선택된 심도 인트라 코딩 모드를 표시하는 상기 하나 이상의 선택스 엘리먼트들을 포함하는 심도 모델링 테이블을 인코딩하도록 구성되며,

상기 심도 모델링 테이블은 상기 HEVC 인트라 코딩 모드들에 대한 선택스 엘리먼트들로부터 분리되어 있는 상기 심도 인트라 코딩 모드들에 대한 선택스 엘리먼트들을 포함하는, 비디오 인코딩 디바이스.

청구항 43

제 11 항에 있어서,

상기 비디오 데이터의 적어도 일부를 디스플레이하도록 구성된 디스플레이 디바이스를 더 포함하는, 비디오 디코딩 디바이스.

청구항 44

제 11 항에 있어서,

하나 이상의 집적 회로들;

하나 이상의 마이크로프로세서들;

하나 이상의 디지털 신호 프로세서들 (DSP들);

하나 이상의 필드 프로그래밍가능 게이트 어레이들 (FPGA들);

데스크톱 컴퓨터;

랩톱 컴퓨터;

태블릿 컴퓨터;
전화기;
텔레비전;
카메라;
디스플레이 디바이스;
디지털 미디어 플레이어;
비디오 게임 콘솔;
비디오 게임 디바이스;
비디오 스트리밍 디바이스; 또는
무선 통신 디바이스
중 적어도 하나를 더 포함하는, 비디오 디코딩 디바이스.

청구항 45

제 21 항에 있어서,
상기 비디오 데이터의 적어도 일부를 디스플레이하는 수단을 더 포함하는, 비디오 디코딩 디바이스.

청구항 46

제 33 항에 있어서,
하나 이상의 집적 회로들;
하나 이상의 마이크로프로세서들;
하나 이상의 디지털 신호 프로세서들 (DSP들);
하나 이상의 필드 프로그래밍가능 게이트 어레이들 (FPGA들);
데스크톱 컴퓨터;
랩톱 컴퓨터;
태블릿 컴퓨터;
전화기;
텔레비전;
카메라;
디스플레이 디바이스;
디지털 미디어 플레이어;
비디오 게임 콘솔;
비디오 게임 디바이스;
비디오 스트리밍 디바이스; 또는
무선 통신 디바이스
중 적어도 하나를 더 포함하는, 비디오 인코딩 디바이스.

청구항 47

제 33 항에 있어서,

상기 비디오 데이터의 적어도 일부를 캡처하도록 구성된 카메라를 더 포함하는, 비디오 인코딩 디바이스.

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

발명의 설명

기술 분야

[0001] 이 출원은 모두 "Simplified Intra Modes Signaling of Depth Maps for 3D-HEVC" 라는 명칭의, 2013 년 3 월 27 일자로 출원된 미국 가출원 제 61/805,901 호, 2013 년 3 월 27 일자로 출원된 미국 가출원 제 61/805,922 호, 2013 년 4 월 11 일자로 출원된 미국 가출원 제 61/811,103 호, 2013 년 10 월 14 일자로 출원된 미국 가출원 제 61/890,629 호, 및 2013 년 10 월 16 일자로 출원된 미국 가출원 제 61/891,816 호의 이익을 주장하고, 이들 각각의 전체 내용은 참조를 위해 본원에 편입된다.

[0002] 이 게시물은 비디오 코딩, 특히, 3 차원 (3D) 비디오 코딩에 관한 것이다.

배경 기술

[0003] 디지털 비디오 기능들은, 디지털 텔레비전들, 디지털 다이렉트 브로드캐스트 시스템 (digital direct broadcast system) 들, 무선 브로드캐스트 시스템들, 개인 정보 단말 (personal digital assistant; PDA) 들, 랩톱 또는 데스크톱 컴퓨터들, 태블릿 컴퓨터들, 전자책 (e-book) 리더들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게임용 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 라디오 전화들, 소위 "스마트폰들", 화상 원격회의 디바이스들, 비디오 스트리밍 디바이스들 등을 포함하는 광범위한 디바이스들 내로 편입될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Part 10, 진보된 비디오 코딩 (Advanced Video Coding; AVC), 고효율 비디오 코딩 (High Efficiency Video Coding; HEVC) 에 의해 정의된 표준들, 및 이러한 표준들의 확장들에서 설명된 것들과 같은 비디오 압축 기법들을 구현한다. 비디오 디바이스들은 이러한 비디오 압축 기법들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신하고, 수신하고, 인코딩하고, 디코딩하고, 및/또는 저장할 수도 있다.

[0004] 비디오 압축 기법들은 비디오 시퀀스들에 내재된 중복성을 감소시키거나 제거하기 위해 공간적 (인트라-픽처 (intra-picture)) 예측 및/또는 시간적 (인터-픽처 (inter-picture)) 예측을 수행한다. 블록-기반 비디오 코딩을 위하여, 비디오 슬라이스 (즉, 픽처 또는 픽처의 부분) 는 비디오 블록들로 파티셔닝 (partitioning) 될 수도 있으며, 이 비디오 블록들은 또한, 트리블록 (treeblock) 들, 코딩 유닛 (coding unit; CU) 들 및/또는 코딩 노드들로서 지칭될 수도 있다. 픽처의 인트라-코딩된 (intra-coded) (I) 슬라이스 내의 비디오 블록들은 동일한 픽처에서의 이웃하는 블록들 내의 참조 샘플들에 대한 공간적 예측을 이용하여 인코딩된다. 픽처의 인터-코딩된 (inter-coded) (P 또는 B) 슬라이스 내의 비디오 블록들은 동일한 픽처에서의 이웃하는 블록들 내의 참조 샘플들에 대한 공간적 예측, 또는 다른 참조 픽처들에서의 참조 샘플들에 대한 시간적 예측을 이용할 수도 있다.

[0005] 공간적 또는 시간적 예측은 코딩되어야 할 블록에 대한 예측 블록으로 귀착된다. 잔차 데이터는 코딩되어야 할 원래의 블록과 예측 블록과의 사이의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 지시하는 모션 벡터와, 코딩된 블록 및 예측 블록 사이의 차이를 표시하는 잔차 데이터에 따라 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라 인코딩된다. 추가의 압축을 위하여, 잔차 데이터는 공간 도메인으로부터 변환 도메인으로 변환되어 잔차 변환 계수들로 귀착될 수도 있고, 그 후 이들은 양자화될 수도 있다. 초기에 2 차원 어레이로 배치된 양자화된 변환 계수들은 변환 계수들의 1 차원 벡터를 생성하기 위해 스캐닝될 수도 있고, 엔트로피 코딩은 훨씬 더 많은 압축을 달성하기 위해 적용될 수도 있다.

[0006] 멀티뷰 코딩 비트스트림 (multiview coding bitstream) 은 예컨대, 다수의 관점들로부터의 뷰들을 인코딩함으로써 생성될 수도 있다. 멀티뷰 코딩 양태들을 이용하는 일부의 3 차원 (3D) 비디오 표준들이 개발되었다. 예를 들어, 상이한 뷰들은 3D 비디오를 지원하기 위하여 좌안 및 우안 뷰 (view) 들을 송신할 수도 있다. 대안적으로, 일부의 3D 비디오 코딩 프로세스들은 소위 멀티뷰 플러스 심도 코딩 (multiview plus depth coding) 을 적용할 수도 있다. 멀티뷰 플러스 심도 코딩에서, 3D 비디오 비트스트림은 텍스처 뷰 (texture view) 컴포넌트들뿐만 아니라, 심도 뷰 (depth view) 컴포넌트들을 포함할 수도 있다. 예를 들어, 각각의 뷰는 하나의 텍스처 뷰 컴포넌트 및 하나의 심도 뷰 컴포넌트를 포함할 수도 있다.

발명의 내용

과제의 해결 수단

[0007] 일반적으로, 이 개시물은 고효율 비디오 코딩 (HEVC) 코딩 모드들과는 상이한 심도 코딩 모드들을 이용하여 멀티뷰 플러스 심도 포맷으로 표현된 3 차원 (3D) 비디오 데이터에 대한 심도 데이터를 인코딩하고 디코딩하기 위한 기법들에 관한 것이다. 3D-HEVC 프로세스와 같은 3D 비디오 코딩 프로세스에서 이용가능한 추가적인 심도 인트라 코딩 모드들의 예들은 심도 모델링 모드 (Depth Modeling Mode; DMM), 간략화된 심도 코딩 (Simplified Depth Coding; SDC) 모드, 및 체인 코딩 모드 (Chain Coding Mode; CCM) 중의 적어도 2 개를 포함한다. 게다가, 추가적인 심도 인트라 코딩 모드의 예는 인트라 SDC 모드를 포함한다. 하나의 예에서, 이 개시물의 기법들은 HEVC 신택스로부터 분리되어 있는 심도 모델링 테이블 (depth modeling table) 에서의 3D 비디오 데이터에 대한 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 시그널링하는 것을 포함한다. 또 다른 예에서, 이 개시물의 기법들은 심도 인트라 코딩 모드들 및 심도 인트라 코딩 모드를 포함하는 심도 코딩 모드들 중의 2 개 이상에 걸쳐 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보의 시그널링을 통합하는 것을 포함한다. 이 개시물의 기법들은 HEVC 인트라 코딩 모드들로부터 분리되어 있는 심도 모델링 테이블 신택스에서 심도 인트라 코딩 모드들을 시그널링함으로써, 그리고 심도 코딩 모드들에 대한 잔차 정보 코딩 메커니즘들을 통합함으로써, 디코더 과성 부담 (decoder parsing burden) 을 감소시킬 수도 있고, 요구된 컨텍스트 모드들 및 이진화 방법들의 수를 감소시킬 수도 있다.

[0008] 하나의 예에서, 이 개시물은 비디오 데이터를 디코딩하는 방법에 관한 것으로, 상기 방법은, 복수의 심도 인트라 코딩 모드들 중의 임의의 것이 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 신택스 엘리먼트를 디코딩하는 단계로서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과는 상이한, 상기 신택스 엘리먼트를 디코딩하는 단계, 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩한 후, 심도 모델링 테이블을 디코딩하는 단계, 심도 모델링 테이블에서의 하나 이상의 신택스 엘리먼트들에 기초하여 심도 인트라 코딩 모드들 중의 하나를 선택하는 단계, 및 선택된 심도 인트라 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하는 단계를 포함한다.

[0009] 또 다른 예에서, 이 개시물은 비디오 데이터를 저장하도록 구성된 메모리 및 하나 이상의 프로세서들을 포함하는 비디오 디코딩 디바이스에 관한 것이다. 비디오 디코딩 디바이스의 프로세서들은, 복수의 심도 인트라 코딩 모드들 중의 임의의 것이 인에이블되는지 여부를 표시하는 신택스 엘리먼트를 디코딩하는 것으로서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과는 상이한, 상기 신택스 엘리먼트를 디코딩하고, 심도 인트라 코딩 모드들이 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩한 후, 심도 모델링 테이블을 디코딩하고, 심도 모델링 테이블에서의 하나 이상의 신택스 엘리먼트들에 기초하여 심도 인트라 코딩 모드들 중의 하나를 선택하고, 그리고 선택된 심도 인트라 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하도록 구성된다.

[0010] 추가의 예에서, 이 개시물은 비디오 디코딩 디바이스에 관한 것으로, 상기 비디오 디코딩 디바이스는, 복수의 심도 인트라 코딩 모드들 중의 임의의 것이 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 신택스 엘리먼트를 디코딩하기 위한 수단으로서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과는 상이한, 상기 디코딩하기 위한 수단, 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩한 후, 심도 모델링 테이블을 디코딩하기 위한 수단, 심도 모델링 테이블에서의 하나 이상의 신택스 엘리먼트들에 기초하여 심도 인트라 코딩 모드들 중의 하나를 선택하기 위한 수단, 및 선택된 심도 인트라 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하기 위한 수단을 포함한다.

[0011] 추가적인 예에서, 이 개시물은 비디오 데이터를 디코딩하기 위한 명령들을 포함하는 컴퓨터-판독가능한 매체에 관한 것으로, 상기 명령들은, 실행될 경우, 하나 이상의 프로세서들로 하여금, 복수의 심도 인트라 코딩 모드들

중의 임의의 것이 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 신택스 엘리먼트를 디코딩하게 하는 것으로서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과는 상이한, 상기 신택스 엘리먼트를 디코딩하게 하고, 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩한 후, 심도 모델링 테이블을 디코딩하게 하고, 심도 모델링 테이블에서의 하나 이상의 신택스 엘리먼트들에 기초하여 심도 인트라 코딩 모드들 중의 하나를 선택하게 하고, 그리고 선택된 심도 인트라 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하게 한다.

[0012] 또 다른 예에서, 이 개시물은 비디오 데이터를 인코딩하는 방법에 관한 것으로, 상기 방법은, 복수의 심도 인트라 코딩 모드들 중의 선택된 하나를 이용하여 비디오 데이터에 대한 심도 데이터를 인코딩하는 단계로서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과는 상이한, 상기 심도 데이터를 인코딩하는 단계, 심도 인트라 코딩 모드들이 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 인코딩하는 단계, 및 선택된 심도 인트라 코딩 모드를 표시하는 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 인코딩하는 단계를 포함한다.

[0013] 추가의 예에서, 이 개시물은 비디오 데이터를 저장하도록 구성된 메모리 및 하나 이상의 프로세서들을 포함하는 비디오 인코딩 디바이스에 관한 것이다. 비디오 인코딩 디바이스의 프로세서들은, 복수의 심도 인트라 코딩 모드들 중의 선택된 하나를 이용하여 비디오 데이터에 대한 심도 데이터를 인코딩하는 것으로서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과는 상이한, 상기 심도 데이터를 인코딩하고, 심도 인트라 코딩 모드들이 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 인코딩하고, 그리고 선택된 심도 인트라 코딩 모드를 표시하는 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 인코딩하도록 구성된다.

[0014] 하나의 예에서, 이 개시물은 비디오 데이터를 디코딩하는 방법에 관한 것으로, 상기 방법은, 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 디코딩하는 단계로서, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는, 상기 신택스 엘리먼트들의 세트를 디코딩하는 단계, 비디오 데이터에 대한 복수의 심도 코딩 모드들 중의 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 디코딩하는 단계, 및 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 심도 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하는 단계를 포함한다.

[0015] 또 다른 예에서, 이 개시물은 비디오 데이터를 저장하도록 구성된 메모리 및 하나 이상의 프로세서들을 포함하는 비디오 디코딩 디바이스에 관한 것이다. 비디오 디코딩 디바이스의 프로세서들은, 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 디코딩하는 것으로서, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는, 상기 신택스 엘리먼트들의 세트를 디코딩하고, 비디오 데이터에 대한 복수의 심도 코딩 모드들 중의 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 디코딩하고, 그리고 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 심도 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하도록 구성된다.

[0016] 추가의 예에서, 이 개시물은 비디오 디코딩 디바이스에 관한 것으로, 상기 비디오 디코딩 디바이스는, 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 디코딩하기 위한 수단으로서, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는, 상기 신택스 엘리먼트들의 세트를 디코딩하기 위한 수단, 비디오 데이터에 대한 복수의 심도 코딩 모드들 중의 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 디코딩하기 위한 수단, 및 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 심도 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하기 위한 수단을 포함한다.

[0017] 추가적인 예에서, 이 개시물은 비디오 데이터를 디코딩하기 위한 명령들을 포함하는 컴퓨터-관독가능한 매체에 관한 것으로, 상기 명령들은, 하나 이상의 프로세서로 하여금, 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 디코딩하게 하는 것으로서, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는, 상기 신택스 엘리먼트들의 세트를 디코딩하게 하고, 비디오 데이터에 대한 복수의 심도 코딩 모드들 중의 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 디코딩하게 하고, 그리고 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 표시된 심도 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩하게 한다.

[0018] 또 다른 예에서, 이 개시물은 비디오 데이터를 인코딩하는 방법에 관한 것으로, 상기 방법은, 복수의 심도 코딩 모드들 중의 선택된 하나를 이용하여 비디오 데이터에 대한 심도 데이터를 인코딩하는 단계를 포함하고, 상기

심도 데이터를 인코딩하는 단계는, 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 심도 데이터의 잔차 정보를 생성하는 단계, 비디오 데이터에 대한 선택된 심도 코딩 모드를 표시하는 하나 이상의 선택스 엘리먼트들을 인코딩하는 단계, 및 심도 데이터의 잔차 정보를 표시하는 선택스 엘리먼트들의 세트를 인코딩하는 단계를 포함하고, 선택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시한다.

[0019] 추가의 예에서, 이 개시물은 비디오 데이터를 저장하도록 구성된 메모리 및 하나 이상의 프로세서들을 포함하는 비디오 인코딩 디바이스에 관한 것이다. 비디오 인코딩 디바이스의 프로세서들은, 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 심도 데이터의 잔차 정보를 생성하고, 비디오 데이터에 대한 선택된 심도 코딩 모드를 표시하는 하나 이상의 선택스 엘리먼트들을 인코딩하고, 그리고 심도 데이터의 잔차 정보를 표시하는 선택스 엘리먼트들의 세트를 인코딩하기 위하여, 복수의 심도 코딩 모드들 중의 선택된 하나를 이용하여 비디오 데이터에 대한 심도 데이터를 인코딩하도록 구성되고, 선택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시한다.

[0020] 개시물의 하나 이상의 예들의 세부사항들은 첨부한 도면들 및 이하의 설명에서 기재된다. 다른 특징들, 목적들, 및 장점들은 설명, 도면들 및 청구항들로부터 명백할 것이다.

도면의 간단한 설명

[0021] 도 1 은 심도 데이터 코딩을 위하여 이 개시물의 기법들을 사용할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2 는 심도 데이터 코딩을 위한 기법들을 구현할 수도 있는 비디오 인코더의 예를 예시하는 블록도이다.

도 3 은 심도 데이터 코딩을 위한 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록도이다.

도 4 는 일반적으로, HEVC 에서의 방향성 인트라 예측 모드들과 연관된 예측 방향들을 예시한다.

도 5 는 각도 인트라 예측 모드들에 대한 V 및 H 의 값들을 도시하는 표이다.

도 6a 및 도 6b 는 영역 경계 체인 코딩 모드 (chain coding mode; CCM) 에서 일 예의 심도 파티션 패턴, 및 일 예의 코딩된 체인들을 각각 예시하는 개념도들이다.

도 7a 및 도 7b 는 심도 모델링 모드 (DMM) 에서, 8x8 블록에 대한 일 예의 웨지렛 패턴 (Wedgelet pattern), 및 8x8 블록에 대한 일 예의 윤곽 패턴 (Contour pattern) 을 각각 예시하는 개념도들이다.

도 8a 및 도 8b 는 DMM 에서, 웨지렛 분리 라인 시작/종료 위치들의 일 예의 대략적 서브세트 및 하나의 웨지렛 패턴 서브세트 내의 일 예의 세분화들을 각각 예시하는 개념도들이다.

도 9 는 상이한 PU 사이즈들에 대한 이용가능한 DMM 서브-모드들 및 웨지렛 패턴들의 수를 도시하는 표이다.

도 10 은 심도 모델링 테이블에서의 3D 비디오 데이터에 대한 심도 데이터를 인코딩하기 위해 이용된 심도 인트라 코딩 모드들을 표시하는 비디오 인코더의 일 예의 동작을 예시하는 플로우차트이다.

도 11 은 심도 모델링 테이블로부터의 3D 비디오 데이터에 대한 심도 데이터를 디코딩하기 위해 이용된 심도 인트라 코딩 모드들을 결정하는 비디오 디코더의 일 예의 동작을 예시하는 플로우차트이다.

도 12 는 2 개 이상의 심도 코딩 모드들에 대해 통합된 메커니즘들을 이용하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 비디오 인코더의 일 예의 동작을 예시하는 플로우차트이다.

도 13 은 2 개 이상의 심도 코딩 모드들에 대해 통합된 메커니즘들을 이용하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 결정하는 비디오 디코더의 일 예의 동작을 예시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0022] 일반적으로, 이 개시물은 고효율 비디오 코딩 (HEVC) 코딩 모드들과는 상이한 심도 코딩 모드들을 이용하여 멀티뷰 플러스 심도 포맷으로 표현된 3 차원 (3D) 비디오 데이터에 대한 심도 데이터를 인코딩하고 디코딩하기 위한 기법들에 관한 것이다. 멀티뷰 플러스 심도 포맷에서, 3D 비디오 데이터의 캡처된 뷰들 (예컨대, 텍스처들) 은 텍스처 데이터에 대한 심도 데이터를 설명하는 대응하는 심도 맵들과 연관된다. 3D-HEVC 프로세스와 같은, 3D 비디오 코딩 프로세스에 따르면, 텍스처 뷰들 및 심도 맵들이 코딩될 수도 있고 3D 비디오 비트스트림

으로 멀티플렉싱될 수도 있다. 심도 맵들은 루마 샘플들이 심도 값들을 나타내는 그레이스케일 (grayscale) 비디오로서 코딩될 수도 있다. 다시 말해서, 텍스처 뷰들의 잔차 값들은 그레이스케일 값들로 맵핑되고, 심도 맵들의 잔차 값들은 제로 디스패리티 평면 (zero disparity plane) 과 관련된 그레이스케일 값들에 대응하는 심도 값들로 맵핑된다.

[0023] 심도 맵들은 일정한 에어리어 (area) 들과, 대응하는 텍스처와의 강한 상관들을 나타내는 예리한 에지 (edge) 들에 의해 특징된다. HEVC 표준에서의 인터-코딩 및 인트라-코딩 모드들과 같은 기존의 인터-코딩 및 인트라-코딩 모드들은 심도 맵들을 코딩하기 위하여 이용될 수 있지만, 텍스처 및 대응하는 심도 사이의 상이한 통계들 및 상관들로 인해, 추가적인 예측 모드들은 심도 맵들에 대해 설계되었다. 추가적인 심도 인트라 코딩 모드들의 예들은 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 및 체인 코딩 모드 (CCM) 를 포함한다. 게다가, 추가적인 심도 인터 코딩 모드의 예는 인터 SDC 모드를 포함한다.

[0024] DDM 은 복수의 서브-모드들을 포함할 수도 있다. 하나의 예시적인 예에서, DMM 은 이 개시물에서 DMM_1, DMM_2, DMM_3 및 DMM_4 로서 지칭된 4 개의 서브-모드들을 포함한다. 심도 서브-모드들의 4 개 전부에서는, 심도 블록이 DMM 패턴에 의해 특징된 2 개의 영역들로 파티셔닝되고, 여기서, 각각의 영역은 일정한 값에 의해 표현된다. 심도 블록이 직선 라인에 의해 2 개의 영역들로 파티셔닝되는 웨지렛 파티셔닝과, 심도 블록이 2 개의 불규칙적인 영역들로 파티셔닝되는 윤곽 파티셔닝을 포함하는, DMM 에서 정의된 2 개의 파티셔닝 모델들이 있다. DMM 웨지렛 파티셔닝 패턴은 명시적으로 시그널링될 수 있거나 (DMM_1), 공간적으로 이웃하는 블록들에 의해 예측될 수 있거나 (DMM_2), 공동-위치된 텍스처 블록들에 의해 예측될 수 있다 (DMM_3). DMM 윤곽 파티셔닝 패턴은 공동-위치된 텍스처 블록들에 의해 예측될 수 있다 (DMM_4). 다른 예들에서, DMM 은 상이한 수, 즉, 4 개보다 더 많거나 더 적은 심도 서브-모드들을 정의할 수도 있다.

[0025] SDC 모드는 또한, 복수의 서브-모드들을 포함할 수도 있다. 예를 들어, SDC 모드는 이 개시물에서 SDCM_DC, SDCM_DMM_1, 및 SDCM_Planar 로서 지칭된 3 개의 서브-모드들을 포함할 수도 있다. SDC 모드에서는, 양자화된 변환 계수들을 코딩하는 대신에, 심도 블록은 일 타입의 파티션, 예컨대, 단일 DC 파티션, 2 개의 DMM_1 파티션들, 또는 단일 평면 파티션과, 각각의 파티션에 대한 잔차 값으로서 시그널링된다. 인터 SDC 모드에서는, 단일의 고유 파티션을 포함하는 각각의 심도 블록에 대해 하나의 DC 잔차 값만을 코딩함으로써, SDC 모드의 개념이 심도 인터 코딩을 위해 추가로 확장된다. 이 개시물은 주로 SDC 모드를 간략화된 심도 코딩 (SDC) 모드로서 지칭하지만, 다른 예들에서는, SDC 모드가 세그먼트이션별 DC 코딩 (Segmentation-wise DC Coding; SDC) 모드로서 지칭될 수도 있다. 다른 예들에서, SCD 모드는 상이한 수, 즉, 3 개보다 더 많거나 더 적은 심도 서브-모드들을 정의할 수도 있다.

[0026] CCM 에서, 심도 블록은 "체인들" 을 이용하여 시그널링되는 임의적인 파티션 패턴에 의해 특징된 2 개의 영역들로 파티셔닝된다. 체인은 샘플과, 8-접속성 (eight-connectivity) 샘플들 중의 하나와의 사이의 접속이다. 각각의 체인에는 방향 인덱스가 배정되고, 일련의 접속된 체인 인덱스들은 파티션 패턴을 정의하기 위하여 시그널링된다. 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드에 대한 심도 데이터의 시그널링은 이하에서 더욱 상세하게 설명된다.

[0027] 하나의 예에서, 이 개시물의 기법들은 심도 모델링 테이블에서 3D 비디오 데이터에 대한 심도 맵들을 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 시그널링하는 것을 포함한다. 심도 인트라 코딩 모드들은 DMM, SDC 모드 및 CCM 중의 2 개 이상을 포함할 수도 있다. 심도 모델링 테이블은 HEVC 인트라 코딩 모드들에 대한 선택으로부터 분리되어 있는 추가적인 심도 인트라 코딩 모드들에 대한 선택을 시그널링한다. 게다가, 기법들은 HEVC 인트라 코딩 모드들과는 상이한 심도 인트라 코딩 모드들 중의 임의의 것이 3D 비디오 데이터에 대해 인에이블되는지 여부를 표시하기 위하여, 플래그 (flag) 와 같은 선택 엘리먼트를 시그널링하는 것을 포함한다. 심도 인트라 코딩 모드들이 인에이블될 때 (즉, 시그널링된 선택 엘리먼트가, 심도 인트라 코딩 모드들 중의 적어도 하나가 3D 비디오 데이터에 대해 인에이블되는 것을 표시할 때), 비디오 디코더는 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들 중의 하나를 결정하기 위하여 심도 모델링 테이블을 디코딩한다. 이 개시물의 기법들은, HEVC 인트라 코딩 모드들로부터 분리되어 있는 심도 모델링 테이블에서의 심도 인트라 코딩 모드들을 시그널링함으로써, 그리고 심도 인트라 코딩 모드들이 언제 인에이블되는지와, 심도 모델링 테이블이 언제 디코딩되어야 하는지를 표시함으로써, 디코더 파싱 부담을 감소시킬 수도 있다.

[0028] 또 다른 예에서, 이 개시물의 기법들은 심도 코딩 모드들 중의 2 개 이상에 걸쳐 3D 비디오 데이터에 대한 심도 맵들의 잔차 정보의 시그널링을 통합하는 것을 포함한다. 심도 코딩 모드들은 DMM, SDC 모드, 인터 SDC 모드 및 CCM 중의 2 개 이상을 포함할 수도 있다. 선택 엘리먼트들의 통합된 세트는 잔차 값들이 심도 코딩

모드들 중의 임의의 것에 대해 코딩되는지 여부를 표시하는 선택스 엘리먼트를 포함할 수도 있다. 선택스 엘리먼트들의 통합된 세트는 또한, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 값들의 절대값 및 부호를 표시하는 선택스 엘리먼트들을 포함할 수도 있다. 심도 코딩 모드들 중의 2 개 이상에 대한 잔차 정보를 표시하기 위하여 선택스 엘리먼트들의 동일한 세트를 공유함으로써, 컨텍스트 모델 선택 및 이진화 (binarization) 방법들이 또한 통합될 수도 있다. 게다가, 심도 인트라 모드들에 대하여, 이웃하는 참조 샘플들은 잔차 값들로서 지칭된 인트라 예측 샘플들의 생성 동안, 또는 잔차 값들에 기초한 심도 값들의 재구성 동안에 필터링되지 않을 수도 있다. 이 개시물의 기법들은 심도 코딩 모드들에 대한 잔차 정보 코딩 메커니즘들을 통합함으로써, 디코더 과잉 부담을 감소시킬 수도 있고, 요구된 컨텍스트 모델들 및 이진화 방법들의 수를 감소시킬 수도 있다.

[0029] 도 1 은 심도 데이터 코딩을 위하여 이 개시물의 기법들을 사용할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템 (10) 을 예시하는 블록도이다. 도 1 에서 도시된 바와 같이, 시스템 (10) 은 목적지 디바이스 (14) 에 의해 더 이후의 시간에 디코딩되어야 할 인코딩된 비디오 데이터를 제공하는 소스 디바이스 (12) 를 포함한다. 특히, 소스 디바이스 (12) 는 컴퓨터-관독가능한 매체 (16) 를 통해 비디오 데이터를 목적지 디바이스 (14) 에 제공한다. 소스 디바이스 (12) 및 목적지 디바이스 (14) 는, 데스크톱 컴퓨터들, 노트북 (즉, 랩톱) 컴퓨터들, 태블릿 컴퓨터들, 셋톱 (set-top) 박스들, 소위 "스마트" 폰들과 같은 전화 핸드셋들, 소위 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게임용 콘솔들, 비디오 스트리밍 디바이스들 등을 포함하는 광범위한 디바이스들 중의 임의의 것을 포함할 수도 있다. 일부의 경우들에는, 소스 디바이스 (12) 및 목적지 디바이스 (14) 가 무선 통신을 위해 구비될 수도 있다.

[0030] 목적지 디바이스 (14) 는 컴퓨터-관독가능한 매체 (16) 를 통해, 디코딩되어야 할 인코딩된 비디오 데이터를 수신할 수도 있다. 컴퓨터-관독가능한 매체 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동시킬 수 있는 임의의 타입의 매체 또는 디바이스를 포함할 수도 있다. 하나의 예에서, 컴퓨터-관독가능한 매체 (16) 는 소스 디바이스 (12) 가 인코딩된 비디오 데이터를 실시간으로 목적지 디바이스 (14) 로 직접 송신하는 것을 인에이블하기 위한 통신 매체를 포함할 수도 있다.

[0031] 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조될 수도 있고, 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 라디오 주파수 (radio frequency; RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 인에이블하기 위해 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0032] 일부의 예들에서, 인코딩된 데이터는 출력 인터페이스 (22) 로부터 저장 디바이스로 출력될 수도 있다. 유사하게, 인코딩된 데이터는 입력 인터페이스에 의해 저장 디바이스로부터 액세스될 수도 있다. 저장 디바이스는 하드 드라이브, 블루-레이 (Blu-ray) 디스크들, DVD 들, CD-ROM 들, 플래시 메모리, 휘발성 또는 비-휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적당한 디지털 저장 매체들과 같은, 분산되거나 국부적으로 액세스된 다양한 데이터 저장 매체들 중의 임의의 것을 포함할 수도 있다. 추가의 예에서, 저장 디바이스는 소스 디바이스 (12) 에 의해 생성된 인코딩된 비디오를 저장할 수도 있는 파일 서버 또는 또 다른 중간 저장 디바이스에 대응할 수도 있다.

[0033] 목적지 디바이스 (14) 는 스트리밍 또는 다운로드를 통해 저장 디바이스로부터 저장된 비디오 데이터를 액세스할 수도 있다. 파일 서버는 인코딩된 비디오 데이터를 저장할 수 있으며 그 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 송신할 수 있는 임의의 타입의 서버일 수도 있다. 일 예의 파일 서버들은 (예컨대, 웹사이트를 위한) 웹 서버, FTP 서버, 네트워크 연결 저장 (network attached storage; NAS) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바이스 (14) 는 인터넷 접속을 포함하는 임의의 표준 데이터 접속을 통해 인코딩된 비디오 데이터를 액세스할 수도 있다. 이것은, 파일 서버 상에 저장된 인코딩된 비디오 데이터를 액세스하기 위해 적당한 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양자의 조합을 포함할 수도 있다. 저장 디바이스로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 그 조합일 수도 있다.

[0034] 이 개시물의 기법들은 무선 애플리케이션들 또는 세팅들로 반드시 제한되는 것은 아니다. 기법들은 오버-더-에어 (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, HTTP 를 통한 동적 적응 스트리밍 (dynamic adaptive streaming over HTTP; DASH) 과 같은 인터넷 스트리밍 비디오 송신들,

데이터 저장 매체 상으로 인코딩되는 디지털 비디오, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은, 다양한 멀티미디어 애플리케이션들 중의 임의의 것의 지원 하에서 비디오 코딩에 적용될 수도 있다. 일부의 예들에서, 시스템 (10)은 비디오 스트리밍, 비디오 재생, 비디오 브로드캐스팅, 및/또는 화상 통화 (video telephony)와 같은 애플리케이션들을 지원하기 위하여 일방향 (one-way) 또는 양방향 (two-way) 비디오 송신을 지원하도록 구성될 수도 있다.

[0035] 도 1의 예에서, 소스 디바이스 (12)는 비디오 소스 (18), 비디오 인코더 (20), 및 출력 인터페이스 (22)를 포함한다. 목적지 디바이스 (14)는 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32)를 포함한다. 이 개시물에 따르면, 소스 디바이스 (12)의 비디오 인코더 (20)는 멀티-뷰 코딩에서 모션 벡터 예측을 위한 기법들을 적용하도록 구성될 수도 있다. 다른 예들에서, 소스 디바이스 및 목적지 디바이스는 다른 컴포넌트들 또는 배치들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (12)는 외부 카메라와 같은 외부 비디오 소스 (18)로부터 비디오 데이터를 수신할 수도 있다. 마찬가지로, 목적지 디바이스 (14)는 통합된 디스플레이 디바이스를 포함하는 것이 아니라, 외부 디스플레이 디바이스와 인터페이스할 수도 있다.

[0036] 도 1의 예시된 시스템 (10)은 하나의 예에 불과하다. 심도 데이터 코딩을 위한 기법들은 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스에 의해 수행될 수도 있다. 일반적으로, 이 개시물의 기법들은 비디오 인코딩 디바이스에 의해 수행되지만, 기법들은 또한, "CODEC"으로서 전형적으로 지칭된 비디오 인코더/디코더에 의해 수행될 수도 있다. 또한, 이 개시물의 기법들은 또한, 비디오 프리프로세서 (video preprocessor)에 의해 수행될 수도 있다. 소스 디바이스 (12) 및 목적지 디바이스 (14)는, 소스 디바이스 (12)가 목적지 디바이스 (14)로의 송신을 위한 코딩된 비디오 데이터를 생성하는 이러한 코딩 디바이스들의 예들에 불과하다. 일부의 예들에서, 디바이스들 (12, 14)은, 디바이스들 (12, 14)의 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 이 때문에, 시스템 (10)은 예컨대, 비디오 스트리밍, 비디오 재생, 비디오 브로드캐스팅, 또는 화상 통화를 위하여, 비디오 디바이스들 (12, 14) 사이에서 일방향 또는 양방향 비디오 송신을 지원할 수도 있다.

[0037] 소스 디바이스 (12)의 비디오 소스 (18)는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 비디오를 포함하는 비디오 아카이브 (video archive), 및/또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 공급 인터페이스를 포함할 수도 있다. 추가의 대안으로서, 비디오 소스 (18)는 소스 비디오로서, 또는 라이브 비디오 (live video), 아카이빙된 비디오 (archived video), 및 컴퓨터-생성된 비디오의 조합으로서, 컴퓨터 그래픽-기반 (computer graphics-based) 데이터를 생성할 수도 있다. 일부의 경우들에는, 비디오 소스 (18)가 비디오 카메라일 경우, 소스 디바이스 (12) 및 목적지 디바이스 (14)가 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 그러나, 위에서 언급된 바와 같이, 이 개시물에서 설명된 기법들은 일반적으로 비디오 코딩에 적용가능할 수도 있고, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다. 각각의 경우에 있어서, 캡처된, 프리-캡처된 (pre-captured), 또는 컴퓨터-생성된 비디오는 비디오 인코더 (20)에 의해 인코딩될 수도 있다. 다음으로, 인코딩된 비디오 정보는 출력 인터페이스 (22)에 의해 컴퓨터-판독가능한 매체 (16) 상으로 출력될 수도 있다.

[0038] 컴퓨터-판독가능한 매체 (16)는 무선 브로드캐스트 또는 유선 네트워크 송신과 같은 일시적 매체 (transient medium)들, 또는 하드 디스크, 플래시 드라이브, 콤팩트 디스크, 디지털 비디오 디스크, 블루-레이 디스크, 또는 다른 컴퓨터-판독가능한 매체들과 같은 저장 매체들 (즉, 비-일시적인 저장 매체들)을 포함할 수도 있다. 일부의 예들에서, 네트워크 서버 (도시되지 않음)는 예컨대, 네트워크 송신을 통해, 소스 디바이스 (12)로부터 인코딩된 비디오 데이터를 수신할 수도 있으며 인코딩된 비디오 데이터를 목적지 디바이스 (14)에 제공할 수도 있다. 유사하게, 디스크 스탬핑 (disc stamping) 설비와 같은 매체 생산 설비의 컴퓨팅 디바이스는 소스 디바이스 (12)로부터 인코딩된 비디오 데이터를 수신할 수도 있고, 인코딩된 비디오 데이터를 포함하는 디스크를 생산할 수도 있다. 그러므로, 컴퓨터-판독가능한 매체 (16)는 다양한 예들에서, 다양한 형태들의 하나 이상의 컴퓨터-판독가능한 매체들을 포함하는 것으로 이해될 수도 있다.

[0039] 이 개시물은 일반적으로, 비디오 인코더 (20)가 어떤 정보를 비디오 디코더 (30)와 같은 또 다른 디바이스에 "시그널링하는 것"을 지칭할 수도 있다. 그러나, 비디오 인코더 (20)는 어떤 신택스 엘리먼트들을 비디오 데이터의 다양한 인코딩된 부분들과 연관시킴으로써 정보를 시그널링할 수도 있다는 것을 이해해야 한다. 즉, 비디오 인코더 (20)는 어떤 신택스 엘리먼트들을 비디오 데이터의 다양한 인코딩된 부분들의 헤더들에 저장함으로써 데이터를 "시그널링"할 수도 있다. 일부의 경우들에는, 이러한 신택스 엘리먼트들이 비디오 디코더 (30)에 의해 수신되고 디코딩되기 전에, 인코딩되고 저장 (예컨대, 컴퓨터-판독가능한 매체 (16)에 저장

됨) 될 수도 있다. 이에 따라, 용어 "시그널링" 은 일반적으로, 매체에 저장된 후의 임의의 시간에 디코딩 디바이스에 의해 취출 (retrieve) 될 수도 있는 선택스 엘리먼트들을, 인코딩 시에 매체에 저장할 때에 발생할 수도 있는 것과 같은 이러한 통신이 실시간으로 또는 실시간에 근접하여 또는 시간의 기간 동안에 발생하든지 간에, 압축된 비디오 데이터를 디코딩하기 위한 선택스 또는 다른 데이터의 통신을 지칭할 수도 있다.

[0040] 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 컴퓨터-관독가능한 매체 (16) 로부터 정보를 수신한다. 컴퓨터-관독가능한 매체 (16) 의 정보는, 블록들 및 다른 코딩된 유닛들, 예컨대, GOP 들의 특성들 및/또는 프로세싱을 설명하는 선택스 엘리먼트들을 포함하는 선택스 정보로서, 비디오 인코더 (20) 에 의해 정의되며 또한, 비디오 디코더 (30) 에 의해 이용되는 상기 선택스 정보를 포함할 수도 있다. 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 음극선관 (cathode ray tube; CRT), 액정 디스플레이 (liquid crystal display; LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (organic light emitting diode; OLED) 디스플레이, 또는 또 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중의 임의의 것을 포함할 수도 있다.

[0041] 도 1 에서 도시되지 않았지만, 일부의 양태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각 오디오 인코더 및 디코더와 통합될 수도 있고, 공통의 데이터 스트림 또는 별도의 데이터 스트림들에서 오디오 및 비디오 양자의 인코딩을 핸들링하기 위한 적절한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 사용자 데이터그램 프로토콜 (user datagram protocol; UDP) 과 같은 다른 프로토콜들을 준수할 수도 있다.

[0042] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각, 적용가능한 바와 같이, 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서 (digital signal processor; DSP) 들, 애플리케이션 특정 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍가능한 게이트 어레이 (field programmable gate array; FPGA) 들, 개별 로직 회로부, 소프트웨어, 하드웨어, 펌웨어 또는 그 임의의 조합들과 같은 다양한 적당한 인코더 또는 디코더 회로부 중의 임의의 것으로서 구현될 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 의 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있고, 인코더들 또는 디코더들의 어느 하나는 조합된 비디오 인코더/디코더 (CODEC) 의 일부로서 통합될 수도 있다. 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 를 포함하는 디바이스는 집적 회로, 마이크로프로세서, 및/또는 셀룰러 전화와 같은 무선 통신 디바이스를 포함할 수도 있다.

[0043] 일부의 예들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 3D-HEVC 와 같은 비디오 코딩 표준에 따라 동작할 수도 있고, HEVC 테스트 모델 (HEVC Test Model; HM) 을 준수할 수도 있다. HEVC 는 ITU-T 비디오 코딩 전문가 그룹 (Video Coding Experts Group; VCEG) 및 ISO/IEC 동화상 전문가 그룹 (Motion Picture Experts Group; MPEG) 의 비디오 코딩에 관한 합동 협력 팀 (Joint Collaboration Team on Video Coding; JCT-VC) 에 의해 개발되었다. HEVC 의 초안 ("FDIS & 최후 호출에 대한) 고효율 비디오 코딩 (HEVC) 텍스트 사양 초안 10" Bross 등, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 JCT-VC, 12 차 회의: Geneva, CH, 2013 년 1 월 14 - 23 일, JCTVC-L1003) 은 http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip 으로부터 입수가 가능하다.

[0044] HEVC 표준화 노력들은 HEVC 테스트 모델 (HM) 로서 지칭된 비디오 코딩 디바이스의 진화형 모델에 기초하였다. HM 은 예컨대, ITU-T H.264/AVC 에 따라 현존하는 디바이스들과 관련된 비디오 코딩 디바이스들의 몇몇 추가적인 기능들을 추정한다. 예를 들어, H.264 가 9 개의 인트라-예측 인코딩 모드들을 제공하는 반면, HM 은 35 개만큼 많은 인트라-예측 인코딩 모드들을 제공할 수도 있다.

[0045] 일반적으로, HM의 작업 모델은, 비디오 픽처 (또는 "프레임") 가 루마 및 크로마 샘플들의 양자를 포함하는 트리블록들 또는 최대 코딩 유닛 (largest coding unit; LCU) 들의 시퀀스 (sequence) 로 분할될 수도 있다는 것을 설명한다. 비트스트림 내의 선택스 데이터는 픽셀들의 수의 측면에서 최대 코딩 유닛인 LCU 에 대한 사이즈를 정의할 수도 있다. 슬라이스는 코딩 순서에서 다수의 연속적인 트리블록들을 포함한다. 픽처는 하나 이상의 슬라이스들로 파티셔닝될 수도 있다. 각각의 트리블록은 쿼드트리에 따라 코딩 유닛 (CU) 들로 분할될 수도 있다. 일반적으로, 쿼드트리 데이터 구조는 CU 당 하나의 노드를 포함하고, 루트 노드 (root node) 는 트리블록에 대응한다. CU 가 4 개의 서브-CU 들로 분할될 경우, CU 에 대응하는 노드는 4 개의 리프 노드 (leaf node) 들을 포함하고, 4 개의 리프 노드들의 각각은 서브-CU 들 중의 하나에 대응한다.

[0046] 쿼드트리 데이터 구조의 각각의 노드는 대응하는 CU 에 대한 선택스 데이터를 제공할 수도 있다. 예를 들어, 쿼드트리에서의 노드는, 노드에 대응하는 CU 가 서브-CU 들로 분할되는지 여부를 표시하는 분할 플래그

(split flag) 를 포함할 수도 있다. CU 에 대한 선택스 엘리먼트들은 재귀적으로 정의될 수도 있고, CU 가 서브-CU 들로 분할되는지 여부에 종속될 수도 있다. CU 가 추가로 분할되지 않을 경우, 그것은 리프-CU 로서 지칭된다. 이 개시물에서는, 원래의 리프-CU 의 명시적 분할이 없더라도, 리프-CU 의 4 개의 서브-CU 들은 또한 리프-CU 들로서 지칭될 것이다. 예를 들어, 16x16 사이즈에서의 CU 가 추가로 분할되지 않을 경우, 16x16 CU 가 결코 분할되지 않았지만, 4 개의 8x8 서브-CU 들이 또한 리프-CU 들로서 지칭될 것이다.

[0047] CU 가 사이즈 구분 (size distinction) 을 가지지 않는다는 것을 제외하고는, CU 는 H.264 표준의 매크로블록 (macroblock) 과 유사한 목적을 가진다. 예를 들어, 트리블록은 4 개의 자식 노드 (child node) 들 (또한, 서브-CU 들로서 지칭됨) 로 분할될 수도 있고, 각각의 자식 노드는 궁극적으로 부모 노드 (parent node) 일 수도 있고, 또 다른 4 개의 자식 노드들로 분할될 수도 있다. 퀴드트리의 리프 노드로서 지칭된, 최종적인 분할되지 않은 자식 노드는 리프-CU 로서 또한 지칭된 코딩 노드를 포함한다. 코딩된 비트스트림과 연관된 선택스 데이터는 최대 CU 심도로서 지칭된, 트리블록이 분할될 수도 있는 최대 횡수를 정의할 수도 있고, 또한, 코딩 노드들의 최소 사이즈를 정의할 수도 있다. 따라서, 비트스트림은 또한, 최소 코딩 유닛 (smallest coding unit; SCU) 을 정의할 수도 있다. 이 개시물은 HEVC 의 문맥에서의 CU, PU, 또는 TU 중의 임의의 것, 또는 다른 표준들의 문맥에서의 유사한 데이터 구조들 (예컨대, H.264/AVC 에서의 매크로블록들 및 그 서브-블록들) 을 지칭하기 위하여 용어 "블록" 을 이용한다.

[0048] CU 는 코딩 노드와, 코딩 노드와 연관된 예측 유닛 (prediction unit; PU) 들 및 변환 유닛 (transform unit; TU) 들을 포함한다. CU 의 사이즈는 코딩 노드의 사이즈에 대응하고, 형상에 있어서 정사각형이어야 한다. CU 의 사이즈는 8x8 픽셀들로부터 64x64 픽셀들 이상의 최대치를 갖는 트리블록의 사이즈까지의 범위일 수도 있다. 각각의 CU 는 하나 이상의 PU 들 및 하나 이상의 TU 들을 포함할 수도 있다. CU 와 연관된 선택스 데이터는 예를 들어, 하나 이상의 PU 들로의 CU 의 파티셔닝을 설명할 수도 있다. 파티셔닝 모드들은 CU 가 스킵 (skip) 또는 직접 모드 (direct mode) 인코딩되는지, 인트라-예측 모드 인코딩되는지, 또는 인터-예측 모드 인코딩되는지 여부의 사이에서 상이할 수도 있다. PU 들은 형상에 있어서 비-정사각형 (non-square) 이 되도록 파티셔닝될 수도 있다. CU 와 연관된 선택스 데이터는 또한, 예를 들어, 퀴드트리에 따른 하나 이상의 TU 들로의 CU 의 파티셔닝을 설명할 수도 있다. TU 는 형상에 있어서 정사각형 또는 비-정사각형 (예컨대, 직사각형) 일 수 있다.

[0049] HEVC 표준은 상이한 CU 들에 대해 상이할 수도 있는, TU 들에 따른 변환들을 허용한다. 인트라 모드들에 대하여, TU 들은 전형적으로, 파티셔닝된 LCU 에 대해 정의된 주어진 CU 내에서의 PU 들의 사이즈에 기초하여 사이즈가 정해지지만, 이것은 항상 그러하지는 않을 수도 있다. TU 들은 전형적으로 동일한 사이즈이거나 PU 들보다 더 작다. 인터 모드들에 대하여, TU 들은 PU 들보다 더 클 수 있다. 일부의 예들에서, CU 에 대응하는 잔차 샘플들은 "잔차 퀴드 트리" (residual quad tree; RQT) 로서 알려진 퀴드트리 구조를 이용하여 더 작은 유닛들로 재분할될 수도 있다. RQT 의 리프 노드들은 변환 유닛 (TU) 들로서 지칭될 수도 있다. TU 들과 연관된 픽셀 차이 값들은 양자화될 수도 있는 변환 계수들을 생성하기 위하여 변환될 수도 있다.

[0050] 리프-CU 는 하나 이상의 예측 유닛 (PU) 들을 포함할 수도 있다. 일반적으로, PU 는 대응하는 CU 의 전부 또는 부분에 대응하는 공간적인 에어리어를 나타내고, PU 에 대한 참조 샘플을 추출하기 위한 데이터를 포함할 수도 있다. 또한, PU 는 예측과 관련된 데이터를 포함한다. 예를 들어, PU 가 인트라-모드 인코딩될 때, PU 에 대한 데이터는 PU 에 대응하는 TU 에 대한 인트라-예측 모드를 설명하는 데이터를 포함할 수도 있는 잔차 퀴드트리 (RQT) 내에 포함될 수도 있다. 또 다른 예로서, PU 가 인터-모드 인코딩될 때, PU 는 PU 에 대한 하나 이상의 모션 벡터들을 정의하는 데이터를 포함할 수도 있다. PU 에 대한 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 컴포넌트, 모션 벡터의 수직 컴포넌트, 모션 벡터에 대한 해상도 (예컨대, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 지시하는 참조 픽처, 및/또는 모션 벡터에 대한 참조 픽처 리스트 (예컨대, List 0, List 1, 또는 List C) 를 설명할 수도 있다.

[0051] 하나 이상의 PU 들을 가지는 리프-CU 는 또한, 하나 이상의 변환 유닛 (TU) 들을 포함할 수도 있다. 위에서 논의된 바와 같이, 변환 유닛들은 RQT (또한, TU 퀴드트리 구조로서 지칭됨) 를 이용하여 특정될 수도 있다. 예를 들어, 분할 플래그는 리프-CU 가 4 개의 변환 유닛들로 분할되는지 여부를 표시할 수도 있다. 다음으로, 각각의 변환 유닛은 추가의 서브-TU 들로 추가로 분할될 수도 있다. TU 가 추가로 분할되지 않을 때, 그것은 리프-TU 로서 지칭될 수도 있다. 일반적으로, 인트라 코딩을 위하여, 리프-CU 에 속하는 모든 리프-TU 들은 동일한 인트라 예측 모드를 공유한다. 즉, 동일한 인트라-예측 모드는 리프-CU 의 모든 TU 들에 대한 예측된 값들을 계산하기 위하여 일반적으로 적용된다. 인트라 코딩을 위하여, 비디오 인코더 (20) 는 TU 에 대응하는 CU 의 부분과 원래의 블록과의 사이의 차이로서, 인트라 예측 모드를 이용하여 각각의 리프-TU 에

대한 잔차 값을 계산할 수도 있다. TU 는 반드시 PU 의 사이즈로 제한되는 것은 아니다. 이에 따라, TU 들은 PU 보다 더 크거나 더 작을 수도 있다. 인트라 코딩을 위하여, PU 는 동일한 CU 에 대한 대응하는 리프-TU 와 연결될 수도 있다. 일부의 예들에서, 리프-TU 의 최대 사이즈는 대응하는 리프-CU 의 사이즈에 대응할 수도 있다.

[0052] 또한, 리프-CU 들의 TU 들은 또한, 잔차 쿼드트리 (RQT) 들로서 지칭된 각각의 쿼드트리 데이터 구조들과 연결될 수도 있다. 즉, 리프-CU 는 어떻게 리프-CU 가 TU 들로 파티셔닝되는지를 표시하는 쿼드트리를 포함할 수도 있다. TU 쿼드트리의 루트 노드는 일반적으로 리프-CU 에 대응하는 반면, CU 쿼드트리의 루트 노드는 일반적으로 트리블록 (또는 LCU) 에 대응한다. 분할되지 않은 RQT 의 TU 들은 리프-TU 들로서 지칭된다. 일반적으로, 이와 다르게 언급되지 않으면, 이 개시물은 리프-CU 및 리프-TU 를 각각 지칭하기 위하여 용어들 CU 및 TU 를 이용한다.

[0053] 비디오 시퀀스는 전형적으로 일련의 픽처들을 포함한다. 본원에서 설명된 바와 같이, "픽처" 및 "프레임" 은 상호 교환가능하게 이용될 수도 있다. 즉, 비디오 데이터를 포함하는 픽처는 비디오 프레임, 또는 간단하게 "프레임" 으로서 지칭될 수도 있다. 픽처들의 그룹 (group of pictures; GOP) 은 일반적으로 비디오 픽처들 중의 일련의 하나 이상을 포함한다. GOP 는 GOP 의 헤더, 픽처들 중의 하나 이상의 픽처의 헤더, 또는 다른 곳에서의 선택스 데이터로서, GOP 에 포함된 다수의 픽처들을 설명하는 상기 선택스 데이터를 포함할 수도 있다. 픽처의 각각의 슬라이스는 각각의 슬라이스에 대한 인코딩 모드를 설명하는 슬라이스 선택스 데이터를 포함할 수도 있다. 비디오 인코더 (20) 는 전형적으로 비디오 데이터를 인코딩하기 위하여 개별적인 비디오 슬라이스들 내의 비디오 블록들에 대해 동작한다. 비디오 블록은 CU 내의 코딩 노드에 대응할 수도 있다. 비디오 블록들은 고정된 또는 변동되는 사이즈들을 가질 수도 있고, 특정된 코딩 표준에 따라 사이즈에 있어서 상이할 수도 있다.

[0054] 일 예로서, HM 은 다양한 PU 사이즈들에 있어서의 예측을 지원한다. 특별한 CU 의 사이즈가 $2N \times 2N$ 인 것으로 가정하면, HM 은 $2N \times 2N$ 또는 $N \times N$ 의 PU 사이즈들에 있어서의 인트라-예측과, $2N \times 2N$, $2N \times N$, $N \times 2N$, 또는 $N \times N$ 의 대칭적인 PU 사이즈들에 있어서의 인터-예측을 지원한다. HM 은 또한, $2N \times nU$, $2N \times nD$, $nL \times 2N$, 및 $nR \times 2N$ 의 PU 사이즈들에 있어서의 인터-예측을 위한 비대칭적 파티셔닝을 지원한다. 비대칭적 파티셔닝에서는, CU 의 하나의 방향이 파티셔닝되지 않는 반면, 다른 방향은 25 % 및 75 % 로 파티셔닝된다. 25 % 파티션에 대응하는 CU 의 부분은 "n" 과, 그 다음으로, "Up (상부)", "Down (하부)", "Left (좌측)", 또는 "Right (우측)" 의 표시에 의해 표시된다. 이에 따라, 예를 들어, " $2N \times nU$ " 는 상부의 $2N \times 0.5N$ PU 및 하부의 $2N \times 1.5N$ PU 로 수평으로 파티셔닝되는 $2N \times 2N$ CU 를 지칭한다.

[0055] 이 개시물에서, " $N \times N$ " 및 "N 대 (by) N" 은 수직 및 수평 차원들의 측면에서의 비디오 블록의 픽셀 차원들, 예컨대, 16×16 픽셀들 또는 16 대 16 픽셀들을 지칭하기 위하여 상호 교환가능하게 이용될 수도 있다. 일반적으로, 16×16 블록은 수직 방향에서의 16 개의 픽셀들 ($y = 16$) 및 수평 방향에서의 16 개의 픽셀들 ($x = 16$) 을 가질 것이다. 마찬가지로, $N \times N$ 블록은 일반적으로 수직 방향에서의 N 개의 픽셀들 및 수평 방향에서의 N 개의 픽셀들을 가지며, 여기서, N 은 음이 아닌 정수 값을 나타낸다. 블록 내의 픽셀들은 행 (row) 들 및 열 (column) 들로 배치될 수도 있다. 또한, 블록들은 수직 방향에서와 동일한 수의 픽셀들을 수평 방향에서 반드시 가질 필요는 없다. 예를 들어, 블록들은 $N \times M$ 픽셀들을 포함할 수도 있으며, 여기서, M 은 반드시 N 과 동일하지는 않다.

[0056] CU 의 PU 들을 이용한 인트라-예측 또는 인터-예측 코딩에 후속하여, 비디오 인코더 (20) 는 CU 의 TU 들에 대한 잔차 데이터를 계산할 수도 있다. PU 들은 공간 도메인 (또한 픽셀 도메인으로서 지칭됨) 에서 예측 픽셀 데이터를 생성하는 방법 또는 모드를 설명하는 선택스 데이터를 포함할 수도 있고, TU 들은 변환, 예컨대, 이산 코사인 변환 (discrete cosine transform; DCT), 정수 변환, 웨이블릿 변환 (wavelet transform), 또는 개념적으로 유사한 변환의 잔차 비디오 데이터로의 적용에 후속하는 변환 도메인에서의 계수들을 포함할 수도 있다. 잔차 데이터는 인코딩되지 않은 픽처의 픽셀들과 PU 들에 대응하는 예측 값들과의 사이의 픽셀 차이들에 대응할 수도 있다. 비디오 인코더 (20) 는 CU 에 대한 잔차 데이터를 포함하는 TU 들을 형성할 수도 있고, 그 다음으로, CU 에 대한 변환 계수들을 생성하기 위하여 TU 들을 변환할 수도 있다.

[0057] 변환 계수들을 생성하기 위한 임의의 변환들에 후속하여, 비디오 인코더 (20) 는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 일반적으로, 계수들을 나타내기 위해 이용된 데이터의 양을 가능한 대로 감소시키기 위하여 변환 계수들이 양자화되어 추가의 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스는 계수들의 일부 또는 전부와 연관된 비트 심도 (bit depth) 를 감소시킬 수도 있다. 예를 들어, n-비트 값은

양자화 동안에 m-비트 값으로 버림 (round down) 될 수도 있고, 여기서, n 은 m 보다 더 크다.

- [0058] 양자화에 후속하여, 비디오 인코더 (20) 는 변환 계수들을 스캔 (scan) 하여, 양자화된 변환 계수들을 포함하는 2 차원 행렬로부터 1 차원 벡터를 생성할 수도 있다. 스캔은 더 높은 에너지 (및 이에 따라, 더 낮은 주파수) 계수들을 어레이의 전방에 배치하고 더 낮은 에너지 (및 이에 따라, 더 높은 주파수) 계수들을 어레이의 후방에 배치하도록 설계될 수도 있다.
- [0059] 일부의 예들에서, 비디오 인코더 (20) 는 양자화된 변환 계수들을 스캐닝하여, 엔트로피 인코딩될 수 있는 직렬화된 벡터 (serialized vector) 를 생성하기 위하여 미리 정의된 스캔 순서를 사용할 수도 있다. 다른 예들에서, 비디오 인코더 (20) 는 적응적 스캔 (adaptive scan) 을 수행할 수도 있다. 1 차원 벡터를 형성하기 위하여 양자화된 변환 계수들을 스캐닝한 후, 비디오 인코더 (20) 는 예컨대, 컨텍스트-적응 가변 길이 코딩 (context-adaptive variable length coding; CAVLC), 컨텍스트-적응 2 진 산술 코딩 (context-adaptive binary arithmetic coding; CABAC), 선택스-기반 컨텍스트-적응 2 진 산술 코딩 (syntax-based context-adaptive binary arithmetic coding; SBAC), 확률 간격 파티셔닝 엔트로피 (Probability Interval Partitioning Entropy; PIPE) 코딩 또는 또 다른 엔트로피 인코딩 방법론에 따라 1 차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (20) 는 또한, 비디오 데이터를 디코딩함에 있어서 비디오 디코더 (30) 에 의한 이용을 위한 인코딩된 비디오 데이터와 연관된 선택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.
- [0060] 비디오 인코더 (20) 는 예컨대, 픽처 헤더, 블록 헤더, 슬라이스 헤더, 또는 GOP 헤더에서, 블록-기반 선택스 데이터, 픽처-기반 선택스 데이터, 및 GOP-기반 선택스 데이터와 같은 선택스 데이터를 비디오 디코더 (30) 로 추가로 전송할 수도 있다. GOP 선택스 데이터는 각각의 GOP 에서 다수의 픽처들을 설명할 수도 있고, 픽처 선택스 데이터는 대응하는 픽처를 인코딩하기 위하여 이용된 인코딩/예측 모드를 표시할 수도 있다.
- [0061] 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 심도 데이터를 인트라 코딩할 수도 있다. 예를 들어, 3D-HEVC 에서, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 심도 맵들에 대한 인트라 모드들을 시그널링할 수도 있다. 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 이 개시물에서 설명된 바와 같이, 3D-HEVC 에서의 심도 맵들의 간략화된 인트라 모드들 시그널링을 위한 기법들을 수행하도록 구성될 수도 있는 디바이스들을 나타낸다.
- [0062] 하나의 예에서, 비디오 인코더 (20) 는 심도 모델링 테이블에서의 3D 비디오 데이터에 대한 심도 맵들을 코딩하기 위하여 이용된 심도 인트라 코딩 모드들을 시그널링하도록 구성될 수도 있고, 비디오 디코더 (30) 는 심도 모델링 테이블 선택스를 파싱함으로써 심도 맵들을 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 결정하도록 구성될 수도 있다. 심도 모델링 테이블은 HEVC 인트라 코딩 모드들에 대한 선택스로부터 추가적인 심도 인트라 코딩 모드들에 대한 선택스를 분리한다. 또 다른 예에서, 비디오 인코더 (20) 는 심도 코딩 모드들의 2 개 이상에 걸쳐 통합되는 메커니즘들을 이용하여 3D 비디오 데이터에 대한 심도 맵들의 잔차 정보를 시그널링하도록 구성될 수도 있고, 비디오 디코더 (30) 는 통합된 메커니즘들을 이용하여 2 개 이상의 심도 코딩 모드들 중의 임의의 것에 대한 심도 맵의 잔차 정보를 결정하도록 구성될 수도 있다. 심도 코딩 모드들은 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드의 양자를 포함할 수도 있다. 이 개시물의 기법들은 추가적인 심도 코딩 모드들과, 심도 코딩 모드들에 따라 코딩된 심도 데이터의 잔차 정보를 결정하기 위한 비디오 디코더 (30) 에 대한 파싱 부담을 감소시킬 수도 있다.
- [0063] 이 개시물에서 설명된 기법들은 비디오 데이터의 인코딩 및/또는 디코딩 (일반적으로, "코딩") 을 지원하여 3 차원 효과를 생성하기 위하여 이용될 수도 있다. 일 예로서, 하나의 이미지가 뷰어의 좌안에 의해 보여지고 다른 이미지가 뷰어의 우안에 의해 보여지도록, 상이한 뷰들의 2 개의 이미지들 (즉, 약간 상이한 수평 위치들을 가지는 2 개의 카메라 관점들에 대응함) 은 실질적으로 동시에 디스플레이될 수도 있다.
- [0064] 이 3D 효과는 예를 들어, 입체 디스플레이 (stereoscopic display) 들 또는 자동입체 디스플레이 (autostereoscopic display) 들을 이용하여 달성될 수도 있다. 입체 디스플레이들은 2 개의 이미지들을 그에 따라 필터링하는 안경류와 함께 이용될 수도 있다. 예를 들어, 수동형 안경은 적당한 눈이 적당한 이미지를 시청하는 것을 보장하기 위하여, 편광된 렌즈들 또는 상이한 착색된 렌즈들을 이용하여 이미지들을 필터링할 수도 있다. 또 다른 예로서, 능동형 안경은 입체 디스플레이와 협력하여 교대로 렌즈들을 급속하게 셔터링 (shutter) 할 수도 있고, 이것은 좌안 이미지 및 우안 이미지를 디스플레이하는 것 사이를 교대할 수도 있다. 자동입체 디스플레이들은 안경이 전혀 필요하지 않는 그러한 방법으로 2 개의 이미지들을 디스플레이한다. 예를 들어, 자동입체 디스플레이들은 각각의 이미지가 뷰어의 적절한 눈들로 투영되게 하도록 구성되는 미러 (mirror) 들 또는 프리즘 (prism) 들을 포함할 수도 있다.

- [0065] 3D 비디오 데이터를 코딩하는 것은 텍스처 및 심도 데이터를 코딩하는 것을 포함할 수도 있다. 일반적으로, 용어 "텍스처" 는 이미지의 휘도 (luminance) (즉, 밝기 또는 "루마 (luma)") 값들 및 이미지의 색도 (chrominance) (즉, 컬러 또는 "크로마 (chroma)") 를 설명하기 위하여 이용된다. 일부의 예들에서, 텍스처 이미지는 휘도 데이터의 하나의 세트와, 청색 색조 (hue) 들 (Cb) 및 적색 색조들 (Cr) 에 대한 색도 데이터의 2 개의 세트들을 포함할 수도 있다. 4:2:2 또는 4:2:0 과 같은 어떤 크로마 포맷들에서는, 크로마 데이터가 루마 데이터와 관련하여 다운샘플링된다. 즉, 색도 픽셀들의 공간 해상도는 대응하는 휘도 픽셀들의 공간 해상도보다 더 낮을 수도 있으며, 예컨대, 휘도 해상도의 1/2 또는 1/4 일 수도 있다.
- [0066] 심도 데이터는 일반적으로 대응하는 텍스처 데이터에 대한 심도 값들을 설명한다. 예를 들어, 심도 맵은 대응하는 텍스처 데이터에 대한 심도를 각각 설명하는 심도 픽셀들 (또는 심도 값들) 의 세트를 포함할 수도 있다. 픽셀은 하나 이상의 텍스처 값들 (예컨대, 휘도 및 색도) 을 가질 수도 있고, 또한, 하나 이상의 심도 값들을 가질 수도 있다. 심도 데이터는 대응하는 텍스처 데이터에 대한 수평 디스패리티를 결정하기 위하여 이용될 수도 있고, 일부의 경우들에는, 수직 디스패리티가 또한 이용될 수도 있다. 심도 데이터는 종종 루마 데이터와 관련하여 서브-샘플링되고, 이것은 다수의 루마 값들이 하나의 특별한 심도 값과 연관될 수도 있음을 의미한다. 루마 데이터와 관련된 심도 데이터의 서브-샘플링은 루마 데이터와 관련된 크로마 데이터의 서브-샘플링과는 유사하거나 상이할 수도 있다.
- [0067] 텍스처 및 심도 데이터를 수신하는 디바이스는 하나의 뷰 (예컨대, 좌안 뷰) 에 대한 제 1 텍스처 이미지를 디스플레이할 수도 있고, 제 1 텍스처 이미지를 수정하여, 심도 값들에 기초하여 결정된 수평 디스패리티 값들에 의해 제 1 이미지의 픽셀 값들을 오프셋 (offset) 함으로써 다른 뷰 (예컨대, 우안 뷰) 에 대한 제 2 텍스처 이미지를 생성하기 위하여 심도 데이터를 이용할 수도 있다. 일반적으로, 수평 디스패리티 (또는 간단하게 "디스패리티") 는 우측 뷰에서의 대응하는 픽셀까지의 제 1 뷰에서의 픽셀의 수평적 공간 오프셋 (horizontal spatial offset) 을 설명하며, 여기서, 2 개의 픽셀들은 2 개의 뷰들에서 나타낸 것과 동일한 객체의 동일한 부분에 대응한다.
- [0068] 또 다른 예들에서, 심도 데이터는 이미지 평면에 수직인 z-차원에서의 픽셀들에 대해 정의될 수도 있어서, 주어진 픽셀과 연관된 심도는 이미지에 대해 정의된 제로 디스패리티 평면 (zero disparity plane) 과 관련하여 정의된다. 이러한 심도는 픽셀을 디스플레이하기 위한 수평 디스패리티를 생성하기 위하여 이용될 수도 있어서, 픽셀은 제로-디스패리티 평면과 관련된 픽셀의 z-차원 심도 값에 따라 좌안 및 우안에 대해 상이하게 디스플레이된다.
- [0069] 제로-디스패리티 평면은 비디오 시퀀스의 상이한 부분들 대해 변화할 수도 있고, 제로-디스패리티 평면과 관련된 심도의 양이 또한 변화할 수도 있다. 제로-디스패리티 평면 상에 위치한 픽셀들은 좌안 및 우안에 대해 유사하게 정의될 수도 있다. 제로-디스패리티 평면의 전방에 위치한 픽셀들은, 픽셀이 이미지 평면에 수직인 z-방향에서의 이미지로부터 나오는 것으로 보인다는 지각 (perception) 을 만들도록 좌안 및 우안에 대해 상이한 로케이션들에서 (예컨대, 수평 디스패리티를 갖도록) 디스플레이될 수도 있다.
- [0070] 제로-디스패리티 평면의 후방에 위치한 픽셀들은 심도의 약간의 지각을 제공하기 위하여 약간의 블러 (blur) 를 갖도록 디스플레이될 수도 있거나, 좌안 및 우안에 대해 상이한 로케이션들에서 (예컨대, 제로-디스패리티 평면의 전방에 위치한 픽셀들의 것과 반대인 수평 디스패리티를 갖도록) 디스플레이될 수도 있다. 이미지에 대한 심도 데이터를 전달하거나 정의하기 위하여 다수의 다른 기법들이 또한 이용될 수도 있다.
- [0071] 2 차원 비디오 데이터는 일반적으로 개별 픽처들의 시퀀스로서 코딩되고, 그 각각은 특별한 시간적 인스턴스에 대응한다. 즉, 각각의 픽처는 시퀀스에서 다른 이미지들의 재생 시간들과 관련하여 연관된 재생 시간을 가진다. 이 픽처들은 텍스처 픽처들 또는 텍스처 이미지들로 고려될 수도 있다. 심도-기반 3D 비디오 코딩에서는, 시퀀스에서의 각각의 텍스처 픽처가 또한 심도 맵에 대응할 수도 있다. 즉, 텍스처 픽처에 대응하는 심도 맵은 대응하는 텍스처 픽처에 대한 심도 데이터를 설명한다. 멀티뷰 비디오 데이터는 다양한 상이한 뷰들에 대한 데이터를 포함할 수도 있으며, 여기서, 각각의 뷰는 텍스처 픽처들 및 대응하는 심도 픽처들의 각각의 시퀀스를 포함할 수도 있다.
- [0072] 위에서 언급된 바와 같이, 이미지들은 특별한 시간적 인스턴스에 대응할 수도 있다. 비디오 데이터는 액세스 유닛들의 시퀀스를 이용하여 표현될 수도 있으며, 여기서, 각각의 액세스 유닛은 특별한 시간적 인스턴스에 대응하는 모든 데이터를 포함한다. 이에 따라, 예를 들어, 멀티뷰 비디오 데이터 플러스 심도에 대하여, 공통의 시간적 인스턴스에 대한 각각의 뷰로부터의 텍스처 이미지들뿐만 아니라, 텍스처 이미지들의 각각에 대한 심도 맵들은 특별한 액세스 유닛 내에 모두 포함될 수도 있다. 액세스 유닛은 텍스처 이미지에 대응하는 텍

스처 컴포넌트와, 심도 맵에 대응하는 심도 컴포넌트에 대한 데이터를 포함할 수도 있다.

- [0073] 이러한 방식으로, 3D 비디오 데이터는, 캡처되거나 생성된 뷰들 (텍스처) 이 대응하는 심도 맵들과 연관되는 멀티뷰 비디오 플러스 심도 포맷을 이용하여 표현될 수도 있다. 또한, 3D 비디오 코딩에서는, 텍스처들 및 심도 맵들이 코딩될 수도 있고 3D 비디오 비트스트림으로 멀티플렉싱될 수도 있다. 심도 맵들은 그레이스케일 이미지들로서 코딩될 수도 있으며, 여기서, 심도 맵들의 "루마" 샘플들 (즉, 픽셀들) 은 심도 값들을 나타낸다. 일반적으로, 심도 데이터의 블록 (심도 맵의 샘플들의 블록) 은 심도 블록으로서 지칭될 수도 있다. 심도 값은 심도 샘플과 연관된 루마 값을 지칭할 수도 있다. 어떤 경우에도, 인트라-코딩 및 인터-코딩 방법들은 심도 맵 코딩을 위해 적용될 수도 있다.
- [0074] 심도 맵들은 보편적으로 예리한 에지들 및 일정한 에어리어들을 포함하고, 심도 맵들에서의 에지들은 전형적으로 대응하는 텍스처 데이터와의 강한 상관들을 제시한다. 텍스처 및 대응하는 심도 사이의 상이한 통계들 및 상관들로 인해, 상이한 코딩 방식들이 2D 비디오 코덱에 기초한 심도 맵들에 대해 설계될 수 있다.
- [0075] 이 개시물의 기법들은 일반적으로 심도 맵들의 간략화된 코딩 모드들 시그널링에 관한 것이고, 고효율 비디오 코딩 (HEVC) 표준에 적용가능할 수도 있다. 합동 비디오 팀 (Joint Video Team; JVT) 은 이전에 개발된 비디오 코딩 표준들보다 더 높은 효율을 제공하는 HEVC 의 기본 버전 (2D) 을 개발하였다. 3D 비디오 코딩에 관한 합동 협력 팀 (Joint Collaboration Team on 3D Video Coding; JCT-3V) 은 HEVC 에 대한 확장들로서 2 개의 3 차원 비디오 (3DV) 솔루션들의 연구를 현재 행하고 있다. 하나의 예의 3DV 솔루션은 MV-HEVC 로서 지칭되는 HEVC 의 멀티뷰 확장을 포함한다. 또 다른 예는 심도-강화된 (depth-enhanced) 3D 비디오 확장 (3D-HEVC) 을 포함한다.
- [0076] 3D-HEVC 에서, 각각의 액세스 유닛은, 고유의 뷰 id, 또는 뷰 순서 인덱스, 또는 계층 id 를 각각 포함하는 다수의 뷰 컴포넌트들을 포함한다. 뷰 컴포넌트는 심도 뷰 컴포넌트뿐만 아니라 텍스처 뷰 컴포넌트를 포함한다. 텍스처 뷰 컴포넌트는 하나 이상의 텍스처 슬라이스들로서 코딩될 수도 있는 반면, 심도 뷰 컴포넌트는 하나 이상의 심도 슬라이스들로서 코딩될 수도 있다.
- [0077] 일부의 사례들에서, 심도 데이터는 인트라 코딩될 수도 있고, 이것은 주어진 픽처 내의 공간적 중복성을 감소시키거나 제거하기 위하여 공간적 예측에 의존한다. 예를 들어, 3D-HEVC 에서, 비디오 코더 (예컨대, 비디오 인코더 또는 비디오 디코더) 는 심도 슬라이스의 인트라 예측 유닛을 코딩하기 위하여 기본 (2D) HEVC 표준으로부터의 인트라 예측 모드들을 이용할 수도 있다. HEVC 표준의 인트라 코딩 모드들은 도 4 및 도 5 에 대하여 이하에서 더욱 상세하게 설명된다.
- [0078] 또 다른 예에서, 비디오 코더는 심도 슬라이스의 인트라 예측 유닛을 코딩하기 위하여 영역 경계 체인 코딩 (region boundary chain coding) 을 이용할 수도 있다. 영역 경계 체인 코딩 모드 (간단하게 체인 코딩 모드 또는 CCM 으로서 지칭됨) 는 도 6a 및 도 6b 에 대하여 이하에서 더욱 상세하게 설명된다. 일반적으로, 비디오 코더는 심도 데이터의 블록을, 이후에 인트라 코딩될 수도 있는 불규칙적인 형상의 영역들로 파티셔닝하기 위하여 CCM 을 이용할 수도 있다.
- [0079] 또 다른 예에서, 비디오 코더는 심도 슬라이스의 인트라-예측 유닛을 코딩하기 위하여 간략화된 심도 코딩 (SDC) 모드를 이용할 수도 있다. SDC 모드는 이하에서 더욱 상세하게 설명된다. 다른 인트라-모드 코딩 방식들과 대조적으로, SDC 모드를 이용할 때, 비디오 코더는 잔차 심도 값들을 변환하거나 양자화하지 않는다. 오히려, 비디오 코더는 잔차 심도 값들을 직접적으로 코딩할 수도 있다.
- [0080] 또 다른 예에서, 비디오 코더는 심도 슬라이스의 인트라 예측 유닛을 코딩하기 위하여 심도 모델링 모드 (DMM) 를 이용할 수도 있다. 3D-HEVC 의 DMM 은 도 7a 및 도 7b 에 대하여 이하에서 더욱 상세하게 설명된다. DMM 에 의하여, 비디오 코더는 심도 데이터의 블록 (일반적으로, 심도 블록으로서 지칭됨) 을 예측 영역들로 파티셔닝할 수도 있다. 예를 들어, 비디오 코더는 심도 데이터의 블록을 통해 인출된 임의적인 라인에 의해 정해진 웨지렛 패턴, 또는 심도 블록을 2 개의 불규칙적인 형상의 영역들로 파티셔닝하는 윤곽 패턴을 이용하여 심도 데이터의 블록을 파티셔닝할 수도 있다. 웨지렛 패턴은 도 8a 및 도 8b, 그리고 도 9 에 대하여 이하에서 더욱 상세하게 설명된다.
- [0081] 이 개시물의 어떤 기법들은 일반적으로 3D-HEVC 에 대하여 설명될 수도 있지만, 기법들은 이러한 방법으로 반드시 제한되지는 않는다. 본원에서 설명된 기법들은 또한, 다른 현재의 표준들 또는 아직 개발되지 않은 미래의 표준들에 적용가능할 수도 있고, 특히, 여기서, 심도 맵들의 간략화된 인트라 모드들 시그널링을 제공하는 것이 유리할 수도 있다.

- [0082] 도 2 는 심도 데이터 코딩을 위한 기법들을 구현할 수도 있는 비디오 인코더 (20) 의 예를 예시하는 블록도이다. 비디오 인코더 (20) 는 이 개시물에서 설명된 바와 같이, 3D-HEVC 에서의 심도 맵들의 간략화된 인트라 모드들 시그널링을 위한 기법들을 수행하도록 구성될 수도 있는 디바이스를 나타낸다.
- [0083] 비디오 인코더 (20) 는 비디오 슬라이스들 내의 비디오 블록들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라-코딩은 주어진 비디오 프레임 또는 픽처 내의 비디오에 있어서의 공간적 중복성을 감소시키거나 제거하기 위하여 공간적 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 인접한 프레임들 또는 픽처들 내의 비디오에 있어서의 시간적 중복성을 감소시키거나 제거하기 위하여 시간적 예측에 의존한다. 인트라-모드 (I 모드) 는 몇몇 공간 기반 코딩 모드들 중의 임의의 것을 지칭할 수도 있다. 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 과 같은 인터-모드들은 몇몇 시간 기반 코딩 모드들 중의 임의의 것을 지칭할 수도 있다.
- [0084] 위에서 언급된 바와 같이, 비디오 인코더 (20) 는 멀티뷰 비디오 코딩을 수행하도록 구비될 수도 있다. 일부의 사례들에서, 비디오 인코더 (20) 는, 시간 인스턴스에서의 각각의 뷰가 비디오 디코더 (30) 와 같은 디코더에 의해 프로세싱될 수도 있도록, 멀티-뷰 HEVC 를 코딩하도록 구성될 수도 있다. 3D-HEVC 에 대하여, 각각의 뷰에 대한 텍스처 맵들 (즉, 루마 및 크로마 값들) 을 인코딩하는 것에 추가하여, 비디오 인코더 (20) 는 각각의 뷰에 대한 심도 맵을 추가로 인코딩할 수도 있다.
- [0085] 도 2 에서 도시된 바와 같이, 비디오 인코더 (20) 는 인코딩되어야 할 비디오 프레임 내의 현재의 비디오 블록을 수신한다. 도 2 의 예에서, 비디오 인코더 (20) 는 비디오 데이터 메모리 (41), 모드 선택 유닛 (40), 디코딩된 픽처 버퍼 (decoded picture buffer; DPB) (64), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 모드 선택 유닛 (40) 은 궁극적으로, 모션 보상 유닛 (44), 모션 추정 유닛 (42), 인트라 예측 유닛 (46), 및 파티션 유닛 (48) 을 포함한다. 비디오 블록 재구성을 위하여, 비디오 인코더 (20) 는 또한, 역양자화 유닛 (58), 역변환 유닛 (60), 및 합산기 (62) 를 포함한다. 디블록킹 필터 (deblocking filter; 도 2 에서 도시되지 않음) 는 또한, 블록 경계들을 필터링하여 재구성된 비디오로부터 블록화 아티팩트 (blockiness artifact) 들을 제거하기 위하여 포함될 수도 있다. 희망하는 경우, 디블록킹 필터는 전형적으로 합산기 (62) 의 출력을 필터링할 것이다. (루프 내의 또는 루프 이후의) 추가적인 필터들은 또한, 디블록킹 필터에 추가하여 이용될 수도 있다. 이러한 필터들은 간결함을 위하여 도시되어 있지 않지만, 희망하는 경우, (루프내 (in-loop) 필터로서) 합산기 (50) 의 출력을 필터링할 수도 있다. 일 예의 필터들은 적응적 루프 필터들, 샘플 적응적 오프셋 (sample adaptive offset; SAO) 필터들 또는 다른 타입들의 필터들을 포함할 수도 있다.
- [0086] 비디오 데이터 메모리 (41) 는 비디오 인코더 (20) 의 컴포넌트들에 의해 인코딩되어야 할 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (41) 에서 저장된 비디오 데이터는 예를 들어, 비디오 소스 (18) 로부터 얻어질 수도 있다. 디코딩된 픽처 버퍼 (64) 는 예컨대, 인트라-코딩 또는 인터-코딩 모드들에서 비디오 인코더 (20) 에 의해 비디오 데이터를 인코딩함에 있어서 이용하기 위한 참조 비디오 데이터를 저장하는 참조 픽처 메모리일 수도 있다. 비디오 데이터 메모리 (41) 및 디코딩된 픽처 버퍼 (64) 는 동기식 DRAM (synchronous DRAM; SDRAM), 자기저항 RAM (magnetoresistive RAM; MRAM), 저항 RAM (resistive RAM; RRAM), 또는 다른 타입들의 메모리 디바이스들을 포함하는 동적 랜덤 액세스 메모리 (dynamic random access memory; DRAM) 와 같은 다양한 메모리 디바이스들 중의 임의의 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (41) 및 디코딩된 픽처 버퍼 (64) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (41) 는 비디오 인코더 (20) 의 다른 컴포넌트들과 온-칩 (on-chip) 일 수도 있거나, 그러한 컴포넌트들과 관련하여 오프-칩 (off-chip) 일 수도 있다.
- [0087] 인코딩 프로세스 동안, 비디오 인코더 (20) 는 코딩되어야 할 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 다수의 비디오 블록들로 분할될 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 시간적 예측을 제공하기 위하여, 하나 이상의 참조 프레임들 내의 하나 이상의 블록들과 관련하여 수신된 비디오 블록의 인트라-예측 코딩을 수행한다. 인트라 예측 유닛 (46) 은 대안적으로, 공간적 예측을 제공하기 위하여, 코딩되어야 할 블록과 동일한 프레임 또는 슬라이스에서의 하나 이상의 이웃하는 블록들과 관련하여 수신된 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다. 비디오 인코더 (20) 는 예컨대, 비디오 데이터의 각각의 블록에 대한 적절한 코딩 모드를 선택하기 위하여, 다수의 코딩 패스 (coding pass) 들을 수행할 수도 있다.
- [0088] 또한, 파티션 유닛 (48) 은 이전의 코딩 패스들에서의 이전의 파티셔닝 방식들의 평가에 기초하여, 비디오 데이

터의 블록들을 서브-블록들로 파티셔닝할 수도 있다. 예를 들어, 파티션 유닛 (48) 은 먼저, 프레임 또는 슬라이스를 LCU 들로 파티셔닝할 수도 있고, 레이트-왜곡 (rate-distortion) 분석 (예컨대, 레이트-왜곡 최적화) 에 기초하여 LCU 들의 각각을 서브-CU 들로 파티셔닝할 수도 있다. 모드 선택 유닛 (40) 은 서브-CU 들로의 LCU 의 파티셔닝을 표시하는 쿼드트리 데이터 구조를 추가로 생성할 수도 있다. 쿼드트리의 리프-노드 (leaf-node) CU 들은 하나 이상의 PU 들 및 하나 이상의 TU 들을 포함할 수도 있다.

[0089] 모드 선택 유닛 (40) 은 예컨대, 여러 결과들에 기초하여 코딩 모드들 중의 하나, 인트라 또는 인터를 선택할 수도 있고, 결과적인 인트라-코딩된 또는 인터-코딩된 블록을, 잔차 블록 데이터를 생성하기 위하여 합산기 (50) 에, 그리고 참조 프레임으로서의 이용을 위한 인코딩된 블록을 재구성하기 위하여 합산기 (62) 에 제공한다. 모드 선택 유닛 (40) 은 또한, 모션 벡터들, 인트라-모드 표시자들, 파티션 정보, 및 다른 이러한 신택스 정보와 같은 신택스 엘리먼트들을 엔트로피 인코딩 유닛 (56) 에 제공한다.

[0090] 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적들을 위하여 별도로 예시되어 있다. 모션 추정 유닛 (42) 에 의해 수행된 모션 추정은 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 예를 들어, 모션 벡터는 현재의 프레임 (또는 다른 코딩된 유닛) 내에서 코딩되고 있는 현재의 블록과 관련된 참조 프레임 (또는 다른 코딩된 유닛) 내의 예측 블록과 관련된 현재의 비디오 프레임 또는 픽처 내의 비디오 블록의 PU 의 변위를 표시할 수도 있다.

[0091] 예측 블록은, 절대차의 합 (sum of absolute difference; SAD), 제곱차의 합 (sum of square difference; SSD), 또는 다른 차이 메트릭들에 의해 결정될 수도 있는 픽셀 차이의 측면에서, 코딩되어야 할 블록과 근접하게 일치하는 것으로 판명되는 블록이다. 일부의 예들에서, 비디오 인코더 (20) 는 디코딩된 픽처 버퍼 (64) 에 저장된 참조 픽처들의 정수 미만 (sub-integer) 픽셀 위치들에 대한 값들을 계산할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 참조 픽처의 1/4 픽셀 위치들, 1/8 픽셀 위치들, 또는 다른 분수 픽셀 위치들의 값들을 보간할 수도 있다. 그러므로, 모션 추정 유닛 (42) 은 전체 픽셀 위치들 및 분수 픽셀 위치들과 관련하여 모션 검색을 수행할 수도 있고, 분수 픽셀 정밀도를 갖는 모션 벡터를 출력할 수도 있다.

[0092] 모션 추정 유닛 (42) 은 PU 의 위치를 참조 픽처의 예측 블록의 위치와 비교함으로써, 인터-코딩된 슬라이스에서의 비디오 블록의 PU 에 대한 모션 벡터를 계산한다. 참조 픽처는 제 1 참조 픽처 리스트 (List 0) 또는 제 2 참조 픽처 리스트 (List 1) 로부터 선택될 수도 있고, 이들의 각각은 디코딩된 픽처 버퍼 (64) 에 저장된 하나 이상의 참조 픽처들을 식별한다. 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다.

[0093] 모션 보상 유닛 (44) 에 의해 수행된 모션 보상은 모션 추정 유닛 (42) 에 의해 결정된 모션 벡터에 기초하여 예측 블록을 페치 (fetch) 하거나 생성하는 것을 포함할 수도 있다. 다시, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 일부의 예들에서 기능적으로 통합될 수도 있다. 현재의 비디오 블록의 PU 에 대한 모션 벡터를 수신할 시에, 모션 보상 유닛 (44) 은 모션 벡터가 참조 픽처 리스트들 중의 하나에서 지시하는 예측 블록을 위치시킬 수도 있다. 합산기 (50) 는 이하에서 논의된 바와 같이, 코딩되고 있는 현재의 비디오 블록의 픽셀 값들로부터 예측 블록의 픽셀 값들을 감산하여 픽셀 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다. 일반적으로, 모션 추정 유닛 (42) 은 루마 컴포넌트들과 관련하여 모션 추정을 수행하고, 모션 보상 유닛 (44) 은 크로마 컴포넌트들 및 루마 컴포넌트들의 양자에 대한 루마 컴포넌트들에 기초하여 계산된 모션 벡터들을 이용한다. 모드 선택 유닛 (40) 은 또한, 비디오 슬라이스의 비디오 블록들을 디코딩함에 있어서 비디오 디코더 (30) 에 의해 이용하기 위한 비디오 블록들 및 비디오 슬라이스와 연관된 신택스 엘리먼트들을 생성할 수도 있다.

[0094] 인트라 예측 유닛 (46) 은 위에서 설명된 바와 같이, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 수행된 인터-예측에 대한 대안으로서, 현재의 블록을 인트라-예측할 수도 있다. 특히, 인트라 예측 유닛 (46) 은 현재의 블록을 인코딩하기 위하여 이용하기 위한 인트라-예측 모드를 결정할 수도 있다. 일부의 예들에서, 인트라 예측 유닛 (46) 은 예컨대, 별도의 인코딩 패스들 동안에 다양한 인트라-예측 모드들을 이용하여 현재의 블록을 인코딩할 수도 있고, 인트라 예측 유닛 (46) (또는 일부의 예들에서, 모드 선택 유닛 (40)) 은 테스트된 모드들로부터 이용하기 위한 적절한 인트라-예측 모드를 선택할 수도 있다.

[0095] 예를 들어, 인트라 예측 유닛 (46) 은 다양한 테스트된 인트라-예측 모드들에 대한 레이트-왜곡 분석을 이용하여 레이트-왜곡 값들을 계산할 수도 있고, 테스트된 모드들 중에서 최상의 레이트-왜곡 특성들을 가지는 인트라-예측 모드를 선택할 수도 있다. 레이트-왜곡 분석은 일반적으로, 인코딩된 블록을 생성하기 위해 이용된 비트레이트 (즉, 비트들의 수) 뿐만 아니라, 인코딩된 블록과, 인코딩된 블록을 생성하기 위하여 인코딩되었던

원래의 인코딩되지 않은 블록과의 사이의 왜곡 (또는 에러) 의 양을 결정한다. 인트라 예측 유닛 (46) 은 어느 인트라-예측 모드가 블록에 대한 최상의 레이트-왜곡 값을 나타내는지를 결정하기 위하여 다양한 인코딩된 블록들에 대한 왜곡들 및 레이트들로부터 비율 (ratio) 들을 계산할 수도 있다.

[0096] 게다가, 인트라 예측 유닛 (46) 은 심도 맵의 심도 블록들을 코딩하도록 구성될 수도 있다. 예를 들어, 인트라 예측 유닛 (46) 은 심도 슬라이스의 인트라 예측된 PU 를 코딩하기 위하여, 기본 (2D) HEVC 표준, 영역 경계 체인 코딩 모드 (CCM), 간략화된 심도 코딩 또는 세그먼테이션별 DC 코딩 (SDC), 또는 심도 모델링 모드 (DMM) 로부터의 인트라 예측 모드들을 이용할 수도 있다. 일부의 예들에서, 모션 예측 유닛 (42) 및 모션 보상 유닛 (44) 은 또한, 심도 맵의 심도 블록들을 코딩하도록 구성될 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 심도 슬라이스의 인트라 예측된 PU 를 코딩하기 위하여 기본 (2D) HEVC 표준 또는 인트라 SDC 모드에 따라 인트라 예측을 수행할 수도 있다.

[0097] 변환 프로세싱 유닛 (52) 은 변환을 잔차 블록에 적용하여, 잔차 변환 계수들의 블록을 생성한다. 변환은 잔차 정보를 픽셀 값 도메인으로부터, 주파수 도메인과 같은 변환 도메인으로 변환할 수도 있다. 변환 프로세싱 유닛 (52) 은 이산 코사인 변환 (DCT) 들, 또는 DCT 와 개념적으로 유사한 또는 다른 변환들과 같은 변환들을 수행할 수도 있다. 웨이블릿 변환 (wavelet transform) 들, 정수 변환 (integer transform) 들, 서브-대역 변환 (sub-band transform) 들 또는 다른 타입들의 변환들이 또한 이용될 수 있다. 변환 프로세싱 유닛 (52) 은 결과적인 변환 계수들을 양자화 유닛 (54) 으로 전송할 수도 있다. 일부의 예들에서, 변환 프로세스가 스킵될 수도 있다.

[0098] 양자화 유닛 (54) 은 비트 레이트를 추가로 감소시키기 위하여 변환 계수들을 양자화한다. 양자화 프로세스는 계수들의 일부 또는 전부와 연관된 비트 심도를 감소시킬 수도 있다. 양자화도는 양자화 파라미터를 조절함으로써 수정될 수도 있다. 일부의 예들에서, 다음으로, 양자화 유닛 (54) 은 양자화된 변환 계수들을 포함하는 행렬 (matrix) 의 스캔을 수행할 수도 있다. 대안적으로, 엔트로피 인코딩 유닛 (56) 은 스캔을 수행할 수도 있다.

[0099] 양자화에 후속하여, 엔트로피 인코딩 유닛 (56) 은 양자화된 변환 계수들을 엔트로피 코딩한다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 컨텍스트-적응 가변 길이 코딩 (context adaptive variable length coding; CAVLC), 신택스-기반 컨텍스트-적응 2 진 산술 코딩 (syntax-based context-adaptive binary arithmetic coding; SBAC), 또는 확률 간격 파티셔닝 엔트로피 (probability interval partitioning entropy; PIPE) 코딩과 같은 컨텍스트 적응 2 진 산술 코딩 (context adaptive binary arithmetic coding; CABAC) 또는 다른 엔트로피 코딩 프로세스들을 수행할 수도 있다. 컨텍스트-기반 엔트로피 인코딩의 경우, 컨텍스트는 이웃하는 블록들에 기초할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 코딩에 후속하여, 인코딩된 비트 스트림은 또 다른 디바이스 (예컨대, 비디오 디코더 (30)) 에 송신될 수도 있거나, 더 이후의 송신 또는 취출을 위해 아카이빙될 수도 있다.

[0100] 역양자화 유닛 (58) 및 역변환 유닛 (60) 은 예컨대, 참조 블록으로서의 더 이후의 이용을 위한, 픽셀 도메인에서의 잔차 블록을 재구성하기 위하여 역양자화 및 역변환을 각각 적용한다. 모션 보상 유닛 (44) 은 잔차 블록을 디코딩된 픽처 버퍼 (64) 의 프레임들 중의 하나의 프레임의 예측 블록에 추가함으로써 참조 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한, 모션 추정 시에 이용하기 위한 정수 미만 픽셀 값들을 계산하기 위하여 하나 이상의 보간 필터들을 재구성된 잔차 블록에 적용할 수도 있다.

[0101] 합산기 (62) 는 디코딩된 픽처 버퍼 (64) 에서의 저장을 위한 재구성된 비디오 블록을 생성하기 위하여, 재구성된 잔차 블록을 모션 보상 유닛 (44) 에 의해 생성된 모션 보상된 예측 블록에 추가한다. 재구성된 비디오 블록은 추후의 비디오 프레임에서 블록을 인트라-코딩하기 위하여, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 참조 블록으로서 이용될 수도 있다.

[0102] 이 개시물에서 설명된 기법들에 따르면, 비디오 인코더 (20) 는 HEVC 코딩 모드들과는 상이한 심도 코딩 모드들을 이용하여 멀티뷰 플러스 심도 포맷으로 표현된 3D 비디오 데이터에 대한 심도 데이터를 인코딩하도록 구성될 수도 있다. 3D-HEVC 프로세스와 같은 3D 비디오 코딩 프로세스에서 이용가능한 추가적인 심도 인트라 코딩 모드들의 예들은 DMM, SDC 모드, 및 CCM 중의 적어도 2 개를 포함한다. 게다가, 추가적인 심도 인트라 코딩 모드의 예는 인트라 SDC 모드를 포함한다.

[0103] 하나의 예에서, 비디오 인코더 (20) 는 심도 모델링 테이블에서 3D 비디오 데이터에 대한 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 시그널링한다. 심도 모델링 테이블은 HEVC 인트라 코딩 모드들

에 대한 선택으로부터 분리되어 있는 추가적인 심도 인트라 코딩 모드들에 대한 선택을 시그널링한다. 게다가, 비디오 인코더 (20) 는 HEVC 인트라 코딩 모드들과는 상이한 심도 인트라 코딩 모드들 중의 임의의 것이 3D 비디오 데이터에 대해 인에이블되는지 여부를 표시하기 위하여, 플래그와 같은 선택 엘리먼트를 시그널링한다. 일부의 예들에서, 선택 엘리먼트는 심도 인트라 코딩 모드들의 전부가 3D 비디오 데이터에 대해 인에이블되는지 여부를 표시할 수도 있다. 선택 엘리먼트는 비디오 데이터의 비디오 파라미터 세트 (video parameter set; VPS), 시퀀스 파라미터 세트 (sequence parameter set; SPS) 또는 픽처 파라미터 세트 (SPS) 중의 하나 내에 포함될 수도 있다. 심도 인트라 코딩 모드들이 인에이블될 때, 비디오 인코더 (20) 에 의해 시그널링된 플래그는 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들 중의 하나를 결정하기 위하여 심도 모델링 테이블을 디코딩할 것을 비디오 디코더 (30) 에 표시한다. 심도 모델링 테이블은 비디오 데이터의 코딩 유닛 (CU) 레벨 또는 예측 유닛 (PU) 레벨 중의 하나에서 인코딩될 수도 있다. 이 개시물의 기법들에 따르면, 비디오 인코더 (20) 의 시그널링은, HEVC 인트라 코딩 모드들로부터 분리되어 있는 심도 모델링 테이블에서의 심도 인트라 코딩 모드들을 시그널링함으로써, 그리고 심도 인트라 코딩 모드들이 언제 인에이블되는지와, 심도 모델링 테이블이 언제 디코딩되어야 하는지를 표시함으로써, 디코더 과잉 부담을 감소시킬 수도 있다.

[0104] 또 다른 예에서, 비디오 인코더 (20) 는 심도 코딩 모드들 중의 2 개 이상에 걸쳐 통합된 선택 엘리먼트들의 세트를 이용하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 시그널링하도록 구성될 수도 있다. 심도 코딩 모드들은 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드들을 포함할 수도 있다. 선택 엘리먼트들의 세트는 잔차 값들이 심도 코딩 모드들 중의 임의의 것에 대해 코딩되는지 여부를 표시하는 선택 엘리먼트를 포함할 수도 있다. 선택 엘리먼트들의 세트는 또한, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 값들의 절대값 및 부호를 표시하는 선택 엘리먼트들을 포함할 수도 있다. 심도 코딩 모드들 중의 2 개 이상에 대한 잔차 정보를 표시하기 위하여 선택 엘리먼트들의 동일한 세트를 공유함으로써, 비디오 인코더 (20) 는 또한, 공유된 선택 엘리먼트들의 각각에 대한 컨텍스트 모델 선택 및 이진화 방법들을 통합할 수도 있다. 게다가, 심도 인트라 코딩 모드들에 대하여, 비디오 인코더 (20) 는 심도 데이터의 잔차 값들의 생성 동안에 이웃하는 참조 샘플들을 필터링하지 않을 것을 결정할 수도 있다. 이 개시물의 기법들에 따르면, 비디오 인코더 (20) 의 인코딩 메커니즘들 및 시그널링은 심도 코딩 모드들에 대한 잔차 정보 선택을 통합함으로써 디코더 과잉 부담을 감소시킬 수도 있고 요구된 컨텍스트 모델들 및 이진화 방법들의 수를 감소시킬 수도 있다.

[0105] 도 3 은 심도 데이터 코딩을 위한 기법들을 구현할 수도 있는 비디오 디코더 (30) 의 예를 예시하는 블록도이다. 도 3 의 예에서, 비디오 디코더 (30) 는 비디오 데이터 메모리 (71), 엔트로피 디코딩 유닛 (70), 모션 보상 유닛 (72), 인트라 예측 유닛 (74), 역양자화 유닛 (76), 역변환 유닛 (78), 디코딩된 픽처 버퍼 (DPB; 82) 및 합산기 (80) 를 포함한다. 비디오 디코더 (30) 는 이 개시물에서 설명된 바와 같이, 3D-HEVC 에서의 심도 맵들의 간략화된 인트라 모드들 시그널링을 위한 기법들을 수행하도록 구성될 수도 있는 디바이스를 나타낸다.

[0106] 비디오 데이터 메모리 (71) 는 비디오 디코더 (30) 의 컴포넌트들에 의해 디코딩되어야 할, 인코딩된 비디오 비트스트림과 같은 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (71) 내에 저장된 비디오 데이터는 예를 들어, 컴퓨터-판독가능한 매체 (16) 로부터, 예컨대, 카메라와 같은 로컬 비디오 소스로부터, 비디오 데이터의 유선 또는 무선 네트워크 통신을 통해, 또는 물리적 데이터 저장 매체들을 액세스함으로써 얻어질 수도 있다. 비디오 데이터 메모리 (71) 는 인코딩된 비디오 비트스트림으로부터의 인코딩된 비디오 데이터를 저장하는 코딩된 픽처 버퍼 (coded picture buffer; CPB) 를 형성할 수도 있다. 디코딩된 픽처 버퍼 (82) 는 예컨대, 인트라-코딩 또는 인터-코딩 모드들에서 비디오 디코더 (30) 에 의해 비디오 데이터를 디코딩함에 있어서 사용하기 위한 참조 비디오 데이터를 저장하는 참조 픽처 메모리일 수도 있다. 비디오 데이터 메모리 (71) 및 디코딩된 픽처 버퍼 (82) 는 동기식 DRAM (SDRAM), 자기저항 RAM (MRAM), 저항 RAM (RRAM), 또는 다른 타입들의 메모리 디바이스들을 포함하는 동적 랜덤 액세스 메모리 (DRAM) 와 같은 다양한 메모리 디바이스들 중의 임의의 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (71) 및 디코딩된 픽처 버퍼 (82) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (71) 는 비디오 디코더 (30) 의 다른 컴포넌트들과 온-칩일 수도 있거나, 그러한 컴포넌트들과 관련하여 오프-칩일 수도 있다.

[0107] 디코딩 프로세스 동안, 비디오 디코더 (30) 는 인코딩된 비디오 슬라이스의 비디오 블록들 및 연관된 선택 엘리먼트들을 나타내는 인코딩된 비디오 비트스트림을 비디오 인코더 (20) 로부터 수신한다. 엔트로피 디코딩

유닛 (70) 은 양자화된 계수들, 모션 벡터들 또는 인트라-예측 모드 표시자들, 및 다른 신택스 엘리먼트들을 생성하기 위하여 비트스트림을 엔트로피 디코딩한다. 비디오 디코더 (30) 는 비디오 슬라이스 레벨 및/또는 비디오 블록 레벨에서 신택스 엘리먼트들을 수신할 수도 있다.

- [0108] 위에서 언급된 바와 같이, 비디오 디코더 (30) 는 멀티뷰 비디오 코딩을 수행하도록 구비될 수도 있다. 일부의 사례들에서, 비디오 디코더 (30) 는 멀티뷰 HEVC 를 디코딩하도록 구성될 수도 있다. HEVC-3D 에 대하여, 각각의 뷰에 대한 텍스처 맵들 (즉, 루마 및 크로마 값들) 을 디코딩하는 것에 추가하여, 비디오 디코더 (30) 는 각각의 뷰에 대한 심도 맵을 추가로 디코딩할 수도 있다.
- [0109] 비디오 슬라이스가 인트라-코딩된 (I) 슬라이스로서 코딩될 때, 인트라 예측 유닛 (74) 은 시그널링된 인트라 예측 모드와, 현재의 프레임 또는 픽처의 이전에 디코딩된 블록들로부터의 데이터에 기초하여, 현재의 비디오 슬라이스의 비디오 블록에 대한 예측 데이터를 생성할 수도 있다. 인트라 예측 유닛 (74) 은 또한, 심도 데이터를 인트라 코딩할 수도 있다. 예를 들어, 인트라 예측 유닛 (74) 은 심도 슬라이스의 인트라 예측된 PU 를 코딩하기 위하여, 기본 (2D) HEVC 표준, 영역 경계 체인 코딩 모드 (CCM), 간략화된 심도 코딩 (SDC) 모드, 또는 심도 모델링 모드 (DMM) 로부터의 인트라 예측 모드들을 이용할 수도 있다.
- [0110] 비디오 프레임이 인터-코딩된 (즉, B 또는 P) 슬라이스로서 코딩될 때, 모션 보상 유닛 (72) 은 모션 벡터들과, 엔트로피 디코딩 유닛 (70) 으로부터 수신된 다른 신택스 엘리먼트들에 기초하여, 현재의 비디오 슬라이스의 비디오 블록에 대한 예측 블록들을 생성한다. 예측 블록들은 참조 픽처 리스트들 중의 하나 내의 참조 픽처들 중의 하나로부터 생성될 수도 있다. 비디오 디코더 (30) 는 디코딩된 픽처 버퍼 (82) 에 저장된 참조 픽처들에 기초하여, 디폴트 구성 기법들을 이용하여 참조 프레임 리스트들, List 0 및 List 1 을 구성할 수도 있다.
- [0111] 모션 보상 유닛 (72) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 과싱함으로써 현재의 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 디코딩되고 있는 현재의 비디오 블록에 대한 예측 블록들을 생성하기 위하여 예측 정보를 이용한다. 예를 들어, 모션 보상 유닛 (72) 은 비디오 슬라이스의 비디오 블록들을 코딩하기 위해 이용된 예측 모드 (예컨대, 인트라-예측 또는 인터-예측), 인터-예측 슬라이스 타입 (예컨대, B 슬라이스 또는 P 슬라이스), 슬라이스에 대한 참조 픽처 리스트들 중의 하나 이상에 대한 구성 정보, 슬라이스의 각각의 인터 인코딩된 비디오 블록에 대한 모션 벡터들, 슬라이스의 각각의 인터-코딩된 비디오 블록에 대한 인터-예측 상태, 및 현재의 비디오 슬라이스에서 비디오 블록들을 디코딩하기 위한 다른 정보를 결정하기 위하여 수신된 신택스 엘리먼트들의 일부를 이용한다.
- [0112] 모션 보상 유닛 (72) 은 또한, 보간 필터들에 기초하여 보간을 수행할 수도 있다. 모션 보상 유닛 (72) 은 참조 블록들의 정수 미만 픽셀들에 대한 보간된 값들을 계산하기 위하여, 비디오 블록들의 인코딩 동안에 비디오 인코더 (20) 에 의해 이용된 바와 같은 보간 필터들을 이용할 수도 있다. 이 경우, 모션 보상 유닛 (72) 은 수신된 신택스 엘리먼트들로부터 비디오 인코더 (20) 에 의해 이용된 보간 필터들을 결정할 수도 있고, 예측 블록들을 생성하기 위하여 보간 필터들을 이용할 수도 있다.
- [0113] 모션 보상 유닛 (72) 은 또한, 심도 데이터를 인터 코딩할 수도 있다. 예를 들어, 모션 보상 유닛 (72) 은 심도 슬라이스의 인트라 예측된 PU 를 코딩하기 위하여 기본 (2D) HEVC 표준 또는 인터 SDC 모드에 따라 인터 예측을 수행할 수도 있다.
- [0114] 역양자화 유닛 (76) 은, 비트스트림에서 제공되며 엔트로피 디코딩 유닛 (70) 에 의해 디코딩된 양자화된 변환 계수들을 역양자화 (inverse quantize), 즉, 탈양자화(de-quantize) 한다. 역양자화 프로세스는 양자화도 및 마찬가지로, 적용되어야 할 역양자화도를 결정하기 위하여, 비디오 슬라이스에서의 각각의 비디오 블록에 대해 비디오 디코더 (30) 에 의해 계산된 양자화 파라미터 QP_Y 의 이용을 포함할 수도 있다. 역변환 유닛 (78) 은 픽셀 도메인에서 잔차 블록들을 생성하기 위하여, 역변환, 예컨대, 역 DCT, 역정수 변환, 또는 개념적으로 유사한 역변환 프로세스를 변환 계수들에 적용한다.
- [0115] 모션 보상 유닛 (72) 또는 인트라 예측 유닛 (74) 이 모션 벡터들 또는 다른 신택스 엘리먼트들에 기초하여 현재의 비디오 블록 (예컨대, 텍스처 블록 또는 심도 블록) 에 대한 예측 블록을 생성한 후, 비디오 디코더 (30) 는 역변환 유닛 (78) 으로부터의 잔차 블록들을, 모션 보상 유닛 (72) 또는 인트라 예측 유닛 (74) 에 의해 생성된 대응하는 예측 블록들과 합산함으로써 디코딩된 비디오 블록을 형성한다. 합산기 (80) 는 이 합산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다.
- [0116] 희망하는 경우, 디블록킹 필터는 또한, 블록화 아티팩트들을 제거하기 위하여 디코딩된 블록들을 필터링하도록 적용될 수도 있다. (코딩 루프 내 또는 코딩 루프 이후 중의 어느 하나에서의) 다른 루프 필터들은 또한,

픽셀 천이 (pixel transition) 들을 평활화하거나, 또는 이와 다르게 비디오 품질을 개선시키기 위하여 이용될 수도 있다. 다음으로, 주어진 프레임 또는 픽처에서의 디코딩된 비디오 블록들은, 후속 모션 보상을 위해 이용된 참조 픽처들을 저장하는 디코딩된 픽처 버퍼 (82) 내에 저장된다. 디코딩된 픽처 버퍼 (82) 는 또한, 도 1 의 디스플레이 디바이스 (32) 와 같은 디스플레이 디바이스 상에서의 더 이후의 제시를 위한 디코딩된 비디오를 저장한다.

[0117] 이 개시물에서 설명된 기법들에 따르면, 비디오 디코더 (30) 는 HEVC 코딩 모드들과는 상이한 심도 코딩 모드들을 이용하여 멀티뷰 플러스 심도 포맷으로 표현된 3D 비디오 데이터에 대한 심도 데이터를 디코딩하도록 구성된다. 3D-HEVC 프로세스와 같은 3D 비디오 코딩 프로세스에서 이용가능한 추가적인 심도 인트라 코딩 모드들의 예들은 DMM, SDC 모드, 및 CCM 중의 적어도 2 개를 포함한다. 게다가, 추가적인 심도 인트라 코딩 모드들의 예는 인트라 SDC 모드를 포함한다.

[0118] 하나의 예에서, 비디오 디코더 (30) 는 심도 모델링 테이블에서 3D 비디오 데이터에 대한 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 표시하는 신택스 엘리먼트들을 디코딩한다. 심도 모델링 테이블은 HEVC 인트라 코딩 모드들에 대한 신택스로부터 분리되어 있는 추가적인 심도 인트라 코딩 모드들에 대한 신택스를 포함한다. 게다가, 비디오 디코더 (30) 는 HEVC 인트라 코딩 모드들과는 상이한 심도 인트라 코딩 모드들 중의 임의의 것이 3D 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 플래그와 같은 신택스 엘리먼트를 디코딩한다. 일부의 예들에서, 신택스 엘리먼트는 심도 인트라 코딩 모드들의 전부가 3D 비디오 데이터에 대해 인에이블되는지 여부를 표시할 수도 있다. 신택스 엘리먼트는 비디오 데이터의 비디오 파라미터 세트 (VPS), 시퀀스 파라미터 세트 (SPS) 또는 픽처 파라미터 세트 (PPS) 중의 하나 내에 포함될 수도 있다. 비디오 디코더 (30) 가 디코딩된 신택스 엘리먼트에 기초하여 심도 인트라 코딩 모드들이 인에이블되는 것으로 결정할 때, 비디오 디코더 (30) 는 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들 중의 하나를 결정하기 위하여 심도 모델링 테이블을 디코딩한다. 심도 모델링 테이블은 비디오 데이터의 코딩 유닛 (CU) 레벨 또는 예측 유닛 (PU) 레벨 중의 하나에서 디코딩될 수도 있다. 이 개시물의 기법들에 따르면, 시그널링은 HEVC 인트라 코딩 모드들로부터 분리되어 있는 심도 모델링 테이블에서의 심도 인트라 코딩 모드들을 포함함으로써, 그리고 심도 인트라 코딩 모드들이 언제 인에이블되는지와, 심도 모델링 테이블이 언제 디코딩되어야 하는지를 표시함으로써, 비디오 디코더 (30) 에서 과잉 부담을 감소시킬 수도 있다.

[0119] 또 다른 예에서, 비디오 디코더 (30) 는 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 디코딩하도록 구성될 수도 있고, 여기서, 신택스 엘리먼트들의 세트는 심도 코딩 모드들 중의 2 개 이상에 걸쳐 통합된다. 심도 코딩 모드들은 심도 인트라 코딩 모드들 및 심도 인트라 코딩 모드들을 포함할 수도 있다. 신택스 엘리먼트들의 세트는 잔차 값들이 심도 코딩 모드들 중의 임의의 것에 대해 코딩되는지 여부를 표시하는 신택스 엘리먼트를 포함할 수도 있다. 신택스 엘리먼트들의 세트는 또한, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 값들의 절대값 및 부호를 표시하는 신택스 엘리먼트들을 포함할 수도 있다. 심도 코딩 모드들 중의 2 개 이상에 대한 잔차 정보를 표시하기 위하여 신택스 엘리먼트들의 동일한 세트를 공유함으로써, 비디오 디코더 (30) 는 또한, 공유된 신택스 엘리먼트들의 각각에 대한 컨텍스트 모델 선택 및 이진화 방법들을 통합할 수도 있다. 게다가, 심도 인트라 코딩 모드들에 대하여, 비디오 디코더 (30) 는 수신된 잔차 값들에 기초하여 비디오 데이터의 심도 값들의 재구성 동안에 이웃하는 참조 샘플들을 필터링하지 않을 것을 결정할 수도 있다. 이 개시물의 기법들에 따르면, 코딩 메커니즘들은 심도 코딩 모드들에 대한 잔차 정보 신택스를 통합함으로써 비디오 디코더 (30) 에서 과잉 부담을 감소시킬 수도 있고 요구된 컨텍스트 모델들 및 이진화 방법들의 수를 감소시킬 수도 있다.

[0120] 3D-HEVC 와 부합하는 심도 맵들의 간략화된 코딩 모드 시그널링을 위한 다양한 기법들이 지금부터 더욱 상세하게 설명될 것이다. 기법들은 심도 인트라 코딩 모드들과 관련될 수도 있고, 심도 모델링 모드 (DMM), 간략화된 심도 코딩 (SDC) 모드, 및 체인 코딩 모드 (CCM) 중의 2 개 이상을 포함한다. 기법들은 심도 인트라 코딩 모드들과, 인트라 SDC 모드와 같은 심도 인트라 코딩 모드들과 추가로 관련될 수도 있다. 이러한 기법들은 예를 들어, 도 1 내지 도 3 에서 예시된 디바이스들에 의해, 또는 이 디바이스들과 함께 수행될 수도 있다. 다음의 설명은 다양한 비디오 코딩 표준들 및 기법들의 논의를 포함한다.

[0121] 이 섹션에서는, 이 개시물과 관련된 비디오 코딩 기법들이 우선 검토된다. 비디오 코딩 표준들은 그 스케일러블 비디오 코딩 (Scalable Video Coding; SVC) 및 멀티뷰 비디오 코딩 (Multiview Video Coding; MVC) 확장들을 포함하는, ITU-T H.261, ISO/IEC MPEG-1 비주얼 (Visual), ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG-4 비주얼 및 ITU-T H.264 (또한, ISO/IEC MPEG-4 AVC 로서 알려짐) 를 포함한다. MVC 의 합동 초안은 "Advanced video coding for generic audiovisual services", 2010년 3 월자 ITU-T 추천

안 H.264 에서 설명되어 있다. 게다가, ITU-T VCEG 및 ISO/IEC MPEG 의 JCT-VC 에 의해 개발되고 있는 새로운 비디오 코딩 표준인 HEVC 가 있다. HEVC 의 초안은 http://phenix.it-sudparis.eu/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v5.zip 로부터 입수가 가능하다.

- [0122] JCT-3V 는 HEVC 에 대한 확장들로서 2 개의 3 차원 비디오 (3DV) 솔루션들의 연구를 현재 행하고 있다. 하나의 예의 3DV 솔루션은 MV-HEVC 로서 지칭되는 HEVC 의 멀티뷰 확장을 포함한다. 또 다른 예는 3D-HEVC 로서 지칭된 심도-강화된 3D 비디오 표준을 포함한다.
- [0123] 3D-HEVC 에서, 3 차원 (3D) 비디오 데이터는, 캡처된 뷰들 (예컨대, 텍스처들) 이 대응하는 심도 맵들과 연관되는 멀티뷰 비디오 플러스 심도 포맷을 이용하여 표현된다. 3D-HEVC 에서는, 텍스처들 및 심도 맵들이 코딩되며 3D 비디오 비트스트림으로 멀티플렉싱된다. 심도 맵들은 루마 샘플들이 심도 값들을 나타내는 그레이스케일 비디오로서 코딩되고, 기존의 인트라-코딩 및 인터-코딩 방법들은 심도 맵 코딩을 위해 적용될 수 있다.
- [0124] 심도 맵들은 예리한 에지들 및 일정한 에어리어들에 의해 특징되고, 심도 맵들에서의 에지들은 전형적으로 대응하는 텍스처 데이터와의 강한 상관들을 제시한다. 텍스처 및 대응하는 심도 사이의 상이한 통계들 및 상관들로 인해, 상이한 코딩 방식들이 2D 비디오 코덱에 기초한 심도 맵들에 대해 설계된다.
- [0125] 도 4 는 일반적으로, HEVC 에서의 방향성 인트라 예측 모드들과 연관된 예측 방향들을 예시한다. 기본 (2D) HEVC 표준에서는, 각각의 예측 유닛 (PU) 의 루마 컴포넌트들에 대하여, 인트라 예측 코딩 방법이 도 4 에서 도시된 바와 같이, 33 개의 각도 예측 모드들 (2 로부터 34 까지 인덱싱됨), DC 모드 (1 로 인덱싱됨) 및 평면 모드 (0 으로 인덱싱됨) 와 함께 사용된다. 예를 들어, 위에서 언급된 바와 같이, HEVC 표준은 평면 모드 (모드 0), DC 모드 (모드 1), 및 33 개의 방향성 예측된 모드들 (모드들 2 내지 34) 을 포함하는 35 개의 인트라-예측 모드들을 포함할 수도 있다.
- [0126] 평면 모드에 의하여, 예측은 소위 "평면" 함수를 이용하여 수행된다. DC 모드에 의하여, 예측은 블록 내의 픽셀 값들의 평균화에 기초하여 수행된다. 방향성 예측 모드에 의하여, 예측은 (모드에 의해 표시된 바와 같이) 특별한 방향을 따라 이웃하는 블록의 재구성된 픽셀들에 기초하여 수행된다. 일반적으로, 도 4 에서 도시된 화살표들의 꼬리 단부는 값이 추출되는 이웃하는 픽셀들 중의 상대적인 하나를 나타내는 반면, 화살표들의 머리는 추출된 값이 전파되어 예측 블록을 형성하는 방향을 나타낸다.
- [0127] MPEG 에서의 HEVC-기반 3D 비디오 코딩 (3D-HEVC) 코덱은 제출안들 m22570 및 m22571 에서 제안된 솔루션들에 기초하고 있다. m22570 에 대한 전체 인용은 Schwarz 등, Description of 3D Video Coding Technology Proposal by Fraunhofer HHI (프라운호퍼 (Fraunhofer) HHI 에 의한 3D 비디오 코딩 기술 제안의 설명) (HEVC 호환가능한 구성 A), MPEG 회의 ISO/IEC JTC1/SC29/WG11, Doc. MPEG11/M22570, Geneva, Switzerland, 2011 년 11월/12월이다. m22571 에 대한 전체 인용은 Schwarz 등, Description of 3D Video Technology Proposal by Fraunhofer HHI (프라운호퍼 (Fraunhofer) HHI 에 의한 3D 기술 제안의 설명) (HEVC 호환가능; 구성 B), MPEG 회의 - ISO/IEC JTC1/SC29/WG11, Doc. MPEG11/M22571, Geneva, Switzerland, 2011 년 11월/12월이다.
- [0128] 3D-HEVC 에 대한 참조 소프트웨어 HTM 버전 6.0 은 https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-6.0/ 으로부터 입수가 가능하다. 3D-HEVC 에 대한 소프트웨어 설명 (문서 번호: C1005) 은 http://phenix.it-sudparis.eu/jct2/doc_end_user/documents/3_Geneva/wg11/JCT3V-C1005-v1.zip 으로부터 입수가 가능하다. 예를 들어, 바로 위의 링크는 다음의 문서들: "HEVC Test Model 3" 문서 JCT3V-C1005_spec_d0, Tech 등, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 3D 비디오 코딩 확장 개발에 관한 합동 협력 팀, 3 차 회의: Geneva, CH, 2013 년 1 월 17 - 23, 및 "3D-HEVC Test Model 3", 문서 JCT3V-C1005_d0, Tech 등, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 3D 비디오 코딩 확장 개발에 관한 합동 협력 팀, 3 차 회의: Geneva, CH, 2013 년 17 - 23 일을 지칭한다.
- [0129] 3D-HEVC 에서, 각각의 액세스 유닛은, 고유의 뷰 id, 또는 뷰 순서 인덱스, 또는 계층 id 를 각각 포함하는 다수의 뷰 컴포넌트들을 포함한다. 뷰 컴포넌트는 심도 뷰 컴포넌트뿐만 아니라 텍스처 뷰 컴포넌트를 포함한다. 텍스처 뷰 컴포넌트는 하나 이상의 텍스처 슬라이스들로서 코딩되는 반면, 심도 뷰 컴포넌트는 하나 이상의 심도 슬라이스들로서 코딩된다.
- [0130] 심도 데이터 코딩을 위한 이용가능한 인트라 코딩 모드들은 HEVC 인트라 모드, HEVC 인트라 펄스 코드 변조 (I_PCM) 모드, 영역 경계 체인 코딩 모드 (CCM), 심도 모델링 모드 (DMM), 및 간략화된 심도 코딩 모드 (SDCM

또는 SDC 모드) 이다. 심도 데이터 코딩을 위한 이용가능한 인터 코딩 모드들은 HEVC 인터 모드 및 인터 SDC 모드이다. 이들의 각각은 다음의 하위섹션들에서 순서대로 설명되어 있다.

- [0131] 3D-HEVC 를 위한 HEVC 인트라 모드에서는, 기본 (2D) HEVC 표준에 대한 상기 설명된 바와 같은 인트라 예측 모드들의 동일한 정의가 사용된다. 즉, 도 4 에서 예시된 바와 같이, 총 35 개의 인트라 예측 모드들은 평면 인트라 예측 모드를 지칭하는 인트라 예측 모드 0, DC 예측을 지칭하는 모드 1, 및 상이한 방향성들을 갖는 각도 예측 모드들을 지칭하는 모드들 2 내지 34 로 지원된다.
- [0132] 도 5 는 각도 인트라 예측 모드들에 대한 V 및 H 의 값들을 도시하는 표이다. 방향성 인트라 예측 모드들의 각도는 (H, V) 로서 표시되고, 값은 대략적으로 H/V 와 동일하다. HEVC 인트라 모드들에 대한 가능한 (H, V) 조합들은 도 5 의 표에서 열거되고, 이 표에서, 2 내지 34 에 의해 인덱싱된 각각의 예측 모드는 (H, V) 조합들 중의 하나에 대응한다.
- [0133] HEVC I_PCM 은 또한, 인트라 코딩을 위하여 지원된다. HEVC I_PCM 모드에서는, 예측, 변환, 양자화 및 엔트로피 코딩의 코딩 메커니즘들이 우회되는 반면, 예측 샘플들은 미리 정의된 수의 비트들에 의해 코딩된다.
- [0134] 체인 코딩 모드 (CCM) 로서 지칭된 영역 경계 체인 코딩 모드는 심도 맵 코딩을 위한 하나의 특수한 HEVC 인트라 예측 모드로서 도입된다. CCM 은 JCT3V-A0070 ("CE.6.h Region Boundary Chain Coding for Depth-map" 으로 명명됨, Heo 등, JCT2-A0070, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 3D 비디오 코딩 확장 개발에 관한 합동 협력 팀, 1 차 회의: Stockholm, SE, 2012 년 7 월 16 - 20 일) 에서 설명되었다.
- [0135] CCM 에서, 심도 블록은 "체인들" 을 이용하여 시그널링되는 임의적인 파티션 패턴에 의해 특정된 2 개의 영역들로 파티셔닝된다. 체인은 샘플과, 8-접속성 (eight-connectivity) 샘플들 중의 하나와의 사이의 접속이다. 각각의 체인에는 방향 인덱스가 배정되고, 일련의 접속된 체인 인덱스들은 파티션 패턴을 정의하기 위하여 시그널링된다.
- [0136] 도 6a 및 도 6b 는 CCM 에서 일 예의 심도 파티션 패턴, 및 일 예의 코딩된 체인들을 각각 예시하는 개념도들이다. 도 6a 의 블록 (200) 에서 도시된 바와 같이, 0 로부터 7 까지의 범위인 방향 인덱스가 각각 배정된 8 개의 상이한 타입들의 체인들이 있다. 체인 방향 타입들은 심도 블록의 파티션들을 결정함에 있어서 비디오 코더를 보조할 수도 있다. 방향 인덱스 (0...7) 를 직접 코딩하는 대신에, 방향 인덱스를 시그널링하기 위하여 차분 코딩 (differential coding) 이 적용된다는 것에 주목해야 한다.
- [0137] CCM 에서, rem_intra_luma_pred_mode 가 31 과 동일하고 현재의 PU 사이즈가 4x4 이상이지만 64x64 보다 더 작을 때, 추가적인 플래그 (edge_intra_flag) 는 CCM 이 이용되는지 아닌지 여부를 시그널링하기 위하여 비트스트림에서 송신된다. 이 추가적인 플래그가 CCM 이 현재의 PU 에 적용됨을 표시할 때, (좌측 열 (column) 또는 상부 행 (row), edge_start_left_flag 로부터의) 영역 경계의 시작 포인트, 체인의 시작 위치 (edge_start_position), 체인 코드들의 수 마이너스 1 (num_edge_codes_minus1), 및 각각의 체인 코드에 대한 방향 인덱스 (edge_code) 가 비트스트림에서 추가로 시그널링된다.
- [0138] 도 6b 에서 예시된 일 예의 PU 는 파티셔닝 구조를 표시하는 체인 (206) 에 의해 분리된 제 1 파티션 (202) 및 제 2 파티션 (204) 을 포함한다. (비디오 인코더 (20) 와 같은) 비디오 인코더는 인코딩된 비트스트림에서 현재의 PU 에 대한 체인 (206) 을 결정하고 시그널링할 수도 있는 반면, (비디오 디코더 (30) 와 같은) 비디오 디코더는 인코딩된 비트스트림으로부터의 체인 (206) 을 나타내는 데이터를 파싱할 수도 있다.
- [0139] 일반적으로, 체인 (206) 은 시작 위치, 체인에서의 링크들의 수 (예컨대, 체인 코드들의 수) 의 표시, 및 각각의 체인 코드에 대한 방향 인덱스를 포함한다. 도 6b 의 예에서 도시된 임의적인 파티션 패턴을 시그널링하기 위하여, 비디오 인코더 (20) 는 체인 (206) 이 상부 경계로부터 시작함을 표시하기 위하여 1 비트 (예컨대, 0) 를 인코딩할 수도 있다. 비디오 인코더 (20) 는 체인 (206) 이 상부 경계의 제 3 심도 샘플 후에 시작함, 즉, "3" 의 시작 위치를 표시하기 위하여 3 비트들 (예컨대, 011) 을 추가로 인코딩할 수도 있다. 비디오 인코더 (20) 는 체인 (206) 에서 총 7 개의 링크들이 있음을 표시하기 위하여 4 비트들 (예컨대, 0110) 을 추가로 인코딩할 수도 있다. 비디오 인코더 (20) 는 또한, (예컨대, 블록 (200) 에 따라) 각각의 체인 링크의 방향을 표시하기 위하여 일련의 접속된 체인들 인덱스들 (예컨대, 3, 3, 3, 7, 1, 1, 1) 을 인코딩할 수도 있다. 일부의 예들에서, 비디오 인코더 (20) 는 룩업 테이블을 이용하여 각각의 인덱스를 코드 워드 (code word) 로 변환할 수도 있다. 비디오 디코더 (30) 와 같은 비디오 디코더는 블록의 파티셔닝 패턴을 결정하기 위하여 위에서 설명된 시그널링을 파싱할 수도 있다. 다음으로, 비디오 디코더 (30) 는 각각의 파티션에 대한 심도 값들을 디코딩할 수도 있다.

- [0140] 심도 모델링 모드 (DMM) 방법은 인터 코딩 동안에 심도 맵들에서의 예리한 에지들의 더욱 양호한 표현들을 제공하기 위하여 이용될 수도 있다. DMM 방법은 HEVC 에서 특정된 인트라 예측 모드들에 대한 대안으로서 통합될 수도 있다. 1-비트 플래그는 예를 들어, (HEVC 인트라 모드, HEVC I_PCM 모드 및, 일부의 경우에는, CCM 을 포함하는) DMM 또는 기존의 인트라 예측이 적용되는지 여부를 특정하기 위하여 각각의 PU 에 대해 시그널링될 수도 있다. 하나의 예에서, DMM 에서는 4 개의 인트라 모드들, 즉, DMM_1, DMM_2, DMM_3, 및 DMM_4 가 있다. 전부 4 개의 모드들에서, 심도 블록은 DMM 패턴에 의해 특정된 2 개의 영역들로 파티셔닝될 수도 있고, 여기서, 각각의 영역은 일정한 값에 의해 표현된다. DMM 패턴은 명시적으로 시그널링될 수 있거나 (DMM_1), 공간적으로 이웃하는 블록들에 의해 예측될 수 있거나 (DMM_2), 공동-위치된 텍스처 블록에 의해 예측될 수 있다 (DMM_3 및 DMM_4). 웨지렛 파티셔닝 및 윤곽 파티셔닝을 포함하는, DMM 에서 정의된 2 개의 파티셔닝 모델들이 있다.
- [0141] 도 7a 및 도 7b 는 DMM 에서, 8x8 블록에 대한 일 예의 웨지렛 패턴, 및 8x8 블록에 대한 일 예의 윤곽 패턴을 각각 예시하는 개념도들이다. 도 7a 는 8x8 블록에 대한 하나의 웨지렛 패턴을 도시한다. 웨지렛 파티션에 대하여, 심도 블록 (210) 은 도 7a 에서 예시된 바와 같이, (X_s, Y_s) 에서 위치한 시작 포인트 (218) 및 (X_e, Y_e) 에서 위치한 종료 포인트를 갖는 직선 라인 (216) 에 의해 2 개의 영역들 (212, 214) 로 파티셔닝되고, 여기서, 2 개의 영역들은 P₀ 및 P₁ 로 라벨이 붙여진다. 시작 포인트 (218) 는 포인트 (8, 0) 로서 정의될 수도 있고, 종료 포인트 (220) 는 포인트 (0, 8) 로서 정의될 수도 있다.
- [0142] 도 7a 의 예에서, 심도 블록 (210) 내의 각각의 개별적인 정사각형은 심도 블록 (210) 의 각각의 개별적인 픽셀을 나타낸다. 정사각형들 내의 수치 값들은 대응하는 픽셀이 영역 P₀ (212) (값 "0") 또는 영역 P₁ (214) (값 "1") 에 속하는지 여부를 나타낸다. 픽셀이 영역 (212) (백색 정사각형들) 또는 영역 (214) (회색 정사각형들) 에 속하는지 여부를 표시하기 위하여, 도 7a 에서는 음영처리 (shading) 가 또한 이용된다. 각각의 웨지렛 패턴은 대응하는 샘플이 영역 P₀ 또는 P₁ 에 속하는지 여부에 대한 사이즈 u_B×v_B 의 2 진 디지털 라벨링의 어레이로 구성되고, 여기서, u_B 및 v_B 는 각각 심도 블록 (210) 의 수평 및 수직 사이즈를 나타낸다.
- [0143] 이러한 방식으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 와 같은 비디오 코더는, 심도 블록 (210) 의 픽셀이 영역 (212) (또한, 영역 "P₀" 으로서 지칭될 수도 있음) 또는 영역 (214) (또한, 영역 "P₁" 으로서 지칭될 수도 있음) 에 속하는지 여부를 결정하기 위하여, 시작 포인트 (218) 및 종료 포인트 (220) 에 의해 정의된 바와 같은 라인 (216) 을 이용할 수도 있다. 웨지렛 패턴들은 인코딩 및 디코딩 양자의 시작 시에 초기화된다.
- [0144] 도 7b 는 8x8 블록에 대한 하나의 윤곽 패턴을 도시한다. 윤곽 파티션에 대하여, 심도 블록 (230) 은 2 개의 불규칙적인 영역들로 파티셔닝될 수 있다. 도 7b 의 예에서, 심도 블록 (230) 은 라인들 (236, 238) 에 의해 영역 (232) 및 영역 (234A, 234B) 으로 파티셔닝된다. 영역 (234A) 에서의 픽셀들이 영역 (234B) 에서의 픽셀들에 바로 인접하지는 않지만, 영역들 (234A 및 234B) 은 심도 블록 (230) 의 PU 를 예측할 목적들을 위하여, 하나의 단일 영역을 형성하도록 정의된다. 윤곽 파티셔닝은 웨지렛 파티셔닝보다 더욱 유연성이 있지만, 시그널링하기가 상대적으로 더욱 어려울 수도 있다. 서브-모드 DMM_4 에서는, 3D-HEVC 의 경우, 윤곽 파티셔닝 패턴이 공동-위치된 텍스처 블록의 재구성된 루마 샘플들을 이용하여 명시적으로 유도된다.
- [0145] 도 7b 의 예에서, 심도 블록 (230) 내의 각각의 개별적인 정사각형은 심도 블록 (230) 의 각각의 개별적인 픽셀을 나타낸다. 정사각형들 내의 수치 값들은 대응하는 픽셀이 영역 P₀ (232) (도 7b 의 예에서의 값 "0") 또는 영역 P₁ (234A, 234B) (도 7b 의 예에서의 값 "1") 에 속하는지 여부를 나타낸다. 픽셀이 영역 (232) (백색 정사각형들) 또는 영역 (234A, 234B) (회색 정사각형들) 에 속하는지 여부를 표시하기 위하여, 도 7b 에서는 음영처리가 또한 이용된다. 웨지렛 패턴들에 대하여 위에서 논의된 바와 같이, 각각의 윤곽 패턴은 대응하는 샘플 (즉, 픽셀) 이 영역 P₀ 또는 P₁ 에 속하는지 여부에 대한 사이즈 u_B×v_B 의 2 진 디지털 라벨링의 어레이에 의해 정의될 수도 있고, 여기서, u_B 및 v_B 는 각각 심도 블록 (230) 의 수평 및 수직 사이즈를 나타낸다.
- [0146] 이러한 방식으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 와 같은 비디오 코더는, 심도 블록 (230) 의 픽셀이 영역 (232) (또한, 영역 "P₀" 으로서 지칭될 수도 있음) 또는 영역 (234) (또한, 영역 "P₁" 으로서 지칭될 수도 있음) 에 속하는지 여부를 결정하기 위하여, 라인들 (236, 238) 을 이용할 수도 있다. 영역들 "P₁" 및

"P₂" 은 DMM 에 따라 파티셔닝된 상이한 영역들에 대한 디폴트 명명 규정들이고, 이에 따라, 심도 블록 (210) 의 영역 P₀ 은 심도 블록 (230) 의 영역 P₀ 과 동일한 영역으로 고려되지 않을 것이다.

- [0147] DMM 에서는, 4x4, 8x8, 16x16 및 32x32 블록들에 대해 각각 정의된 총 86, 782, 1394 및 1503 개의 상이한 웨지렛 패턴들이 있다. 이 패턴들은 인코딩 및 디코딩의 양자에서 생성되고 저장된다. DMM 패턴의 타입 (웨지렛 또는 윤곽 파티셔닝) 및 유도에 기초하여, 하기를 포함하는 4 개의 서브-모드들이 DMM 에서 정의된다: 이하에서 더욱 상세하게 각각 설명된 DMM_1 (명시적 웨지렛 시그널링), DMM_2 (인트라 예측된 웨지렛 파티셔닝), DMM_3 (인터-컴포넌트 웨지렛 파티셔닝), 및 DMM_4 (인터-컴포넌트 윤곽 파티셔닝).
- [0148] 서브-모드 DMM_1 은 명시적 웨지렛 시그널링을 정의한다. 서브-모드 DMM_1 에서는, 웨지렛 파티셔닝이 적용되고, 최적의 것은 어떤 왜곡 기준에 기초하여 비디오 인코더 (20) 와 같은 인코더에서 선택된다. 선택된 웨지렛 패턴의 인덱스는 비트스트림에서 시그널링된다. 4x4, 8x8 또는 8x8 보다 더 큰 것과 동일한 PU 사이즈에 대응하는 인덱스를 코딩하기 위하여, 7, 10 또는 11 개의 빈 (bin) 들이 이용된다.
- [0149] 서브-모드 DMM_2 는 인트라 예측된 웨지렛 파티셔닝을 정의한다. 서브-모드 DMM_2 에서는, 웨지렛 파티셔닝이 적용되고, 선택된 웨지렛 패턴은 이웃하는 인트라 예측 모드의 정보 및 웨지렛 패턴 정보를 이용하여 예측된다. 이 경우, 웨지렛 패턴 인덱스의 직접 시그널링은 회피되고, 예측된 웨지렛 패턴의 세분화가 시그널링된다.
- [0150] 서브-모드 DMM_3 은 인터-컴포넌트 웨지렛 파티셔닝을 정의한다. 서브-모드 DMM_3 에서는, 모든 이용가능한 웨지렛 패턴들이 동일한 액세스 유닛의 공동-위치된 루마 텍스처 블록 영역에 대해 철저히 시도되고, 텍스처 블록과 최상으로 일치하는 것이 심도 슬라이스에서의 PU 에 대한 웨지렛 파티션으로서 선택된다. 웨지렛 파티셔닝 패턴의 시그널링이 필요하지 않지만, 최상의 패턴의 철저한 검색은 비디오 디코더 (30) 와 같은 디코더에서 요구된다.
- [0151] 3D-HEVC 표준에서의 JCT3V-C0044 ("CE6.h results on Simplification of Depth Modeling Mode 3" 로 명명됨, Wang 등, JCT3V-C0044, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 3D 비디오 코딩 확장 개발에 관한 합동 협력 팀, 3 차 회의: Geneva, 2013 년 1 월 17 - 23 일) 의 채택으로 인해, 서브-모드 DMM_3 은 위에서 설명된 것과는 다소 상이하도록 수정되었다. 예를 들어, 수정된 서브-모드 DMM_3 에서는, 공동-위치된 텍스처 루마 블록이 각도 인트라 예측 모드들 2 내지 34 중의 하나로 인트라 코딩될 경우, 웨지렛 패턴들의 서브세트로부터 희망하는 또는 최종 웨지렛 패턴을 식별하기 위하여 추가적인 인덱스가 시그널링된다. 웨지렛 패턴들의 서브세트는 공동-위치된 텍스처 블록의 인트라 예측 방향에 의해 식별된다.
- [0152] 서브세트들의 구성의 상세한 도입은 다음과 같이 설명된다. 웨지렛 패턴의 시작 포인트가 S(X_s, Y_s) 이고 종료 포인트가 E(X_e, Y_e) 인 것으로 가정하면, 각각의 웨지렛 패턴에 대응하는 한 쌍의 (S, E) 가 있다. 각각의 인트라 예측 모드 (즉, 모드들 2 내지 34) 에 대하여, 방향 정보 (H_i, V_i) 는 도 5 에서 도시된 표에서와 같이 정의된다. ((X_s, Y_s) 및 E(X_e, Y_e) 에 의해 정의된) 웨지렛 패턴 및 ((H_i, V_i) 에 의해 정의된) 인트라 모드 사이의 적합성 (fitness) 은 하기와 같이 정의된다: D[i] = | V_i × (X_s - X_e) - H_i × (Y_e - Y_s) |. 다음으로, 각각의 웨지렛 패턴은 모든 인트라 예측 모드들 중에서 최소 D 를 검색함으로써 가장 근접한 인트라 예측 모드에 맵핑될 수 있다. 대응하게도, 각각의 인트라 예측 모드 j 에 대하여, 매 패턴의 가장 근접한 인트라 예측 모드가 j 인 웨지렛 패턴 서브세트 S(j) 가 존재한다.
- [0153] 이와 다르게, 공동-위치된 텍스처 루마 블록이 인트라 코딩되지 않거나 인트라 모드가 0 또는 1 인 경우에, 대략적 후보 웨지렛 패턴 서브세트에서의 최상의 웨지렛 패턴의 인덱스가 시그널링된다. 도 8a 및 도 8b 는 DMM 의 서브-모드 DMM_3 에서, 웨지렛 분리 라인 시작/종료 위치들의 일 예의 대략적 서브세트 및 하나의 웨지렛 패턴 서브세트 내의 일 예의 세분화들을 각각 예시하는 개념도들이다. 도 8a 및 도 8b 에서, 흑색-윤곽의 원들은 웨지렛 패턴의 시작/종료 위치들을 표시하고, 회색으로 채워진 흑색-윤곽 원들인 것들은 시작/종료 위치들의 대략적 서브세트를 표시한다. 도 8a 에서 도시된 바와 같이, 대략적 패턴들은 매 2 개의 시작 및 종료 위치들에 대해 생성된다. 각각의 대략적 후보 웨지렛 패턴 서브세트는 도 8b 에서 도시된 바와 같이 8 개에 이르는 참조들을 포함한다. 위에서 설명된 양자의 경우들에 대하여, 시그널링되는 추가적인 인덱스는 동일한 고정된 길이를 가지는 것에 주목해야 한다.
- [0154] 서브-모드 DMM_4 는 인터-컴포넌트 윤곽 파티셔닝을 정의한다. 서브-모드 DMM_4 에서, 윤곽 파티셔닝 패턴은 공동-위치된 텍스처 블록을 2 개의 부분들로 분리함으로써 디코딩 프로세스 동안에 생성되고: 제 1 부분은 평균 값보다 더 큰 루마 샘플 값들을 갖는 픽셀들을 포함하고, 제 2 부분은 평균 값 이하인 루마 샘플 값들을

갖는 픽셀들을 포함한다.

[0155] 도 9 는 상이한 PU 사이즈들에 대한 이용가능한 DMM 서브-모드들 및 웨지렛 패턴들의 수를 도시하는 표이다. 일 예의 구현에서, 4x4 이하인 PU 사이즈들에 대하여, 서브-모드 DMM_1 및 서브-모드 DMM_3 만이 인에이블되고, 4x4 보다 더 크지만 64x64 보다 더 작은, 즉, 8x8, 16x16 및 32x32 인 PU 사이즈들에 대하여, 모든 4 개의 모드들이 인에이블된다. 또한, 일 예의 구현에서는, 64x64 이상인 PU 사이즈에 대하여, 모든 4 개의 모드들이 디스에이블된다. 명료함을 위하여, 상이한 PU 사이즈들에 대한 이용가능한 DMM 서브-모드들 및 대응하는 수의 웨지렛 패턴들은 도 9 에서 도시된 표에서 열거된다.

[0156] 간략화된 심도 코딩 (SDC) 모드는 다른 HEVC 인트라 코딩 모드들에 대한 대안적인 인트라 코딩 모드로서 도입된다. SDC 모드는 JCT3V-B0036 ("3D-CE6.h: Simplified Depth Coding with an optional Depth Lookup Table" 으로 명명됨, Jager 등, ITU-T SG 16 WP 3 및 ISO/IEC JTC 1/SC 29/WG 11 의 3D 비디오 코딩 확장 개발에 관한 합동 협력 팀, 2 차 회의: Shanghai, CN, 2012 년 10 월 13 - 19 일). 현재의 CU 가 SDC 모드에 의해 코딩되는지 아닌지 여부를 특정하기 위하여, 추가적인 플래그가 각각의 인트라 심도 CU 에 대해 시그널링된다. 기존의 3D-HEVC 프로세스에서, SDC 모드는 2Nx2N 파티션 사이즈를 갖는 CU 들에 대해 적용되기만 한다.

[0157] SDC 모드에서는, 양자화된 변환 계수들을 코딩하는 대신에, 심도 블록은 일 타입의 파티션, 예컨대, 단일 DC 파티션, 2 개의 DMM_1 파티션들, 또는 단일 평면 파티션과, 각각의 파티션에 대해 시그널링된 (픽셀 도메인에서의) 잔차 값으로서 표현된다. 3 개의 서브-모드들은, DC, DMM_1, 및 평면의 파티션 타입들에 각각 대응하는 SDCM_DC, SDCM_DMM_1, 및 SDCM_Planar 을 포함하는 SDC 모드에서 정의된다. SDC 모드에서는, 변환 또는 양자화가 적용되지 않고, 심도 값들은, 전체 시퀀스를 인코딩하기 전에 제 1 인트라 주기 내의 프레임들을 분석함으로써 구성되는 심도 룩업 테이블 (Depth Lookup Table; DLT) 을 이용하여 인덱스들에 선택적으로 맵핑될 수 있다. DLT 가 이용될 경우, 전체적인 DLT 는 시퀀스 파라미터 세트 (SPS) 에서, 비디오 디코더 (30) 와 같은 디코더에 송신되고, 디코딩된 인덱스 값들은 DLT 에 기초하여 심도 값들에 다시 맵핑된다. DLT 의 사용을 설명하기 위하여, 이 개시물에서는, DLT 에 따라 심도 값을 인덱스 값으로 변환하기 위하여 표 DLT[·] 가 이용되고, 인덱스 값을 다시 심도 값으로 변환하기 위하여 표 IDLT[·] 가 이용된다.

[0158] 코딩 유닛 (CU) 에 대한 선택스 테이블은 이하에서 주어진다. 선택스의 회색 강조표시된 부분들은 일반적으로, 심도 뷰들에 대한 인트라 코딩의 측면에서 HEVC 표준 및 3D-HEVC 표준 사이의 차이들을 표시한다.

[0159] **G.7.3.9.1 일반적인 코딩 유닛 선택스**

coding_unit(x0, y0, log2CbSize, ciDepth) {	디스크립터
if(transquant_bypass_enable_flag) {	
cu_transquant_bypass_flag	ac(v)
}	
if(slice_type != I)	

[0160]

skip_flag [x0][y0]	ae(v)
if(skip_flag[x0][y0])	
prediction_unit(x0, y0, log2CbSize)	
if (icEnableFlag)	
ic_flag	ae(v)
else {	
nCbS = (1 << log2CbSize)	
if(slice_type != 1)	
pred_mode_flag	ae(v)
if(PredMode == MODE_INTRA && DepthFlag)	
sdc_flag [x0][y0]	ae(v)
[Ed. (GT): 소프트웨어에서, sdc_flag의 위치 및 슬라이스 타입의 종속성이 있는 것으로 보임.]	
if(sdc_flag[x0][y0]) {	
sdc_pred_mode	ae(v)
if(sdc_pred_mode == 1)	
wedge_full_tab_idx [x0][y0]	ae(v)
} else {	
if (PredMode[x0][y0] != MODE_INTRA log2CbSize == Log2MinCbSize) && !predPartModeFlag)	
part_mode	ae(v)
if(PredMode[x0][y0] == MODE_INTRA) {	
if(PartMode == PART_2Nx2N && pcm_enabled_flag && log2CbSize >= Log2MinIPCMCUSize && log2CbSize <= Log2MaxIPCMCUSize)	
pcm_flag	ae(v)
if(pcm_flag) {	
num_subsequent_pcm	tu(3)
NumPCMBlock = num_subsequent_pcm + 1	
while(!byte_aligned())	
pcm_alignment_zero_bit	f(1)
pcm_sample(x0, y0, log2CbSize)	
} else {	
pbOffset = (PartMode == PART_NxN) ? (nCbS / 2) : 0	
if (vps_dmm_flag[nuh_layer_id] && log2CbSize <= Log2MaxDmmCbSize) {	
log2PbSize = log2CbSize - (PartMode == PART_NxN ? 1 : 0)	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
dmm_flag [x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if(dmm_flag[x0 + i][y0 + j])	
dmm_mode [x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULL dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULLDELTA))	

[0161]

wedge_full_tab_idx [x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDDIR dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDDIRDELTA))	
wedge_predtex_tab_idx [x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDDIR dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDDIRDELTA))	
dmm_delta_end_flag [x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (dmm_delta_end_flag[x0 + i][y0 + i])	
dmm_delta_end_abs_minus1 [x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (dmm_delta_end_flag[x0 + i][y0 + i])	
dmm_delta_end_sign_flag [x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
DmmDeltaFlag[x0 + i][y0 + i] = (dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + i] == MODE_DMM_WFULDELTA dmm_mode[x0 + i][y0 + i] == MODE_DMM_WPREDDIRDELTA dmm_mode[x0 + i][y0 + i] == MODE_DMM_WPRETEXDELTA dmm_mode[x0 + i][y0 + i] == MODE_DMM_CPREDTEXDELTA))	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (DmmDeltaFlag[x0 + i][y0 + i])	
dmm_dc_1_abs [x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (DmmDeltaFlag[x0 + i][y0 + j] && dmm_dc_1_abs[x0 + i][y0 + i] != 0)	
dmm_dc_1_sign_flag [x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (DmmDeltaFlag[x0 + i][y0 + j])	
dmm_dc_2_abs [x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if (DmmDeltaFlag[x0 + i][y0 + j] && dmm_dc_2_abs[x0 + i][y0 + i] != 0)	

[0162]

dmm_dc_2_sign_flag[x0 + i][y0 + i]	ae(v)
} else {	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(!dmm_flag[x0 + i][y0 + j])	
prev_intra_luma_pred_flag[x0 + i][y0 + j]	ae(v)
}	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(!dmm_flag[x0 + i][y0 + j]) {	
if(prev_intra_luma_pred_flag[x0 + i][y0 + j])	
mpm_idx[x0 + i][y0 + j]	ae(v)
else {	
rem_intra_luma_pred_mode[x0 + i][y0 + j]	ae(v)
if(rem_intra_luma_pred_mode[x0 + i][y0 + j] == 31 && (log2CbSize + (PartMode == PART_NxN) ? - 1 : 0) <= 5 && (log2CbSize + (PartMode == PART_NxN) ? - 1 : 0) >= 2) {	
edge_intra_flag[x0 + i][y0 + j]	ae(v)
if(edge_intra_flag[x0 + i][y0 + j]) {	
edge_start_left_flag[x0 + i][y0 + j]	ae(v)
edge_start_position[x0 + i][y0 + j]	ae(v)
num_edge_codes_minus1[x0 + i][y0 + j]	ae(v)
for(k = 0; k <= num_edge_codes_minus1; k++)	
edge_code[k]	ae(v)
edge_dc_flag[x0 + i][y0 + j]	ae(v)
if(edge_dc_flag[x0 + i][y0 + j]) {	
edge_dc_1_abs[x0 + i][y0 + j]	ae(v)
if(edge_dc_1_abs[x0 + i][y0 + j] != 0)	
edge_dc_1_sign_flag[x0 + i][y0 + j]	ae(v)
edge_dc_2_abs[x0 + i][y0 + j]	ae(v)
if(edge_dc_2_abs[x0 + i][y0 + j] != 0)	
edge_dc_2_sign_flag[x0 + i][y0 + j]	ae(v)
}	
}	
}	
}	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else {	
if(PartMode == PART_2Nx2N)	
prediction_unit(x0, y0, nCbS, nCbS)	
else if(PartMode == PART_2NxN) {	
prediction_unit(x0, y0, nCbS, nCbS / 2)	

[0163]

prediction_unit(x0, y0 + (nCbS / 2), nCbS, nCbS / 2)	
} else if(PartMode == PART_Nx2N) {	
prediction_unit(x0, y0, nCbS / 2, nCbS)	
prediction_unit(x0 + (nCbS / 2), y0, nCbS / 2, nCbS)	
} else if(PartMode == PART_2NxN) {	
prediction_unit(x0, y0, nCbS, nCbS / 4)	
prediction_unit(x0, y0 + (nCbS / 4), nCbS, nCbS * 3 / 4)	
} else if(PartMode == PART_2NxND) {	
prediction_unit(x0, y0, nCbS, nCbS * 3 / 4)	
prediction_unit(x0, y0 + (nCbS * 3 / 4), nCbS, nCbS / 4)	
} else if(PartMode == PART_nLx2N) {	
prediction_unit(x0, y0, nCbS / 4, nCbS)	
prediction_unit(x0 + (nCbS / 4), y0, nCbS * 3 / 4, nCbS)	
} else if(PartMode == PART_nRx2N) {	
prediction_unit(x0, y0, nCbS * 3 / 4, nCbS)	
prediction_unit(x0 + (nCbS * 3 / 4), y0, nCbS / 4, nCbS)	
} else { /* PART_NxN */	
prediction_unit(x0, y0, nCbS / 2, nCbS / 2)	
prediction_unit(x0 + (nCbS / 2), y0, nCbS / 2, nCbS / 2)	
prediction_unit(x0, y0 + (nCbS / 2), nCbS / 2, nCbS / 2)	
prediction_unit(x0 + (nCbS / 2), y0 + (nCbS / 2), nCbS / 2, nCbS / 2)	
}	
}	
}	
if(icEnableFlag)	
ic_flag	ac(v)
if(!pcm_flag) {	
if(PredMode[x0][y0] != MODE_INTRA && !(PartMode == PART_2Nx2N && merge_flag[x0][y0]))	
no_residual_syntax_flag	ac(v)
if(!no_residual_syntax_flag && !sdc_flag) {	
MaxTrafoDepth = (PredMode[x0][y0] == MODE_INTRA ? max_transform_hierarchy_depth_intra + IntraSplitFlag : max_transform_hierarchy_depth_inter)	
transform_tree(x0, y0 x0, y0, log2CbSize, 0, 0)	
}	
if(sdc_flag) {	
sdcNumSegments = (sdc_pred_mode == 1 ? 2 : 1)	
for (i = 0; i < sdcNumSegments; i++) {	
sdc_residual_flag[x0][y0][i]	ac(v)
if(sdc_residual_flag [x0][y0][i]) {	
sdc_residual_sign_flag[x0][y0][i]	ac(v)
sdc_residual_abs_minus1[x0][y0][i]	ac(v)
}	
}	
}	
}	
}	

[0164]

[0165]

인터 SDC 모드에서는, SDC 모드의 개념이 심도 인터 코딩을 위하여 추가로 확장된다. 인터 SDC 모드는 JCT3V-E1001 (Tech 등, "3D-HEVC Draft Text 1 (Output Document of JCT-3V)", ITU-T SG 16 WP 3 및 ISO/IEC JTC1/SC29/WG11 의 3D 비디오 코딩 확장 개발에 관한 합동 협력 팀 (JCT-3V), 5 차 회의: Vienna, AT, 2013 년 7 월 27 일 - 8 월 2 일) 에서 설명되었다. 3D-HEVC 의 작업 초안 (문서 번호: JCT3V-E1001) 은 다음으로부터 입수가 가능하다: http://phenix.int-evry.fr/jct3v/doc_end_user/documents/5_Vienna/wg11/JCT3V-E1001-v3.zip.

[0166]

인터 SDC 모드는 단일의 고유한 파티션을 포함하는 각각의 PU 에 대한 하나의 DC 잔차 값만을 인코딩함으로써 추가적인 잔차 코딩 방법을 제공한다. 인터 SDC 에 의하여, 변환 및 양자화가 각각의 PU 에 대해 스킵되고, 잔차 블록은 대응하는 시그널링된 DC 잔차 값과 동일한 그 모든 샘플들을 가지며, 이것은 인트라 코딩에서 적용된 SDC 모드의 서브-모드 SDCM_DC 와 유사하다. 비-스킵 (non-skip) CU 에 대하여, 인터 SDC 가 적용되는지 여부를 표시하기 위하여 1-비트 플래그가 추가로 시그널링된다. 스킵 CU 에 대해서는, 현재의 3D-HEVC 에서 행해지는 것과 동일한 프로세스가 유지된다. 인터 SDC 모드가 적용될 경우, 하나의 DC 잔차 값만이 각

각의 PU 에 대해 시그널링되고, 이와 다를 경우, 전통적인 양자화된 변환 계수 블록이 시그널링된다.

[0167] 심도 코딩 모드들에서의 파티션들에 대한 잔차 코딩이 지금부터 더욱 상세하게 설명될 것이다. DMM, CCM, SDC 모드, 및 인터 SDC 모드를 포함하는 심도 코딩 모드들에서는, PU 가 1 개 또는 2 개의 파티션들을 포함하는 지 여부에 관계 없이, 전체의 파티션의 잔차 (또한, "델타 DC (delta DC)" 로서 지칭됨) 는 하나의 단일 값으로 시그널링될 수도 있다. 이러한 방법은 이 개시물에서 델타 DC 코딩으로 칭해진다.

[0168] DMM 을 이용한 델타 DC 코딩과 관련된 신택스 엘리먼트들 및 시맨틱들은 다음과 같이 정의된다.

depth_mode_parameters(x0 , y0 , log2CbSize) {	디스크립터
...	
if(DmmFlag[x0][y0]) {	
dmm_dc_flag [x0][y0]	ae(v)
if (dmm_dc_flag[x0][y0])	
for(i = 0; i < 2; i++) {	
dmm_dc_abs [x0][y0][i]	ae(v)
if (dmm_dc_abs[x0][y0][i])	
dmm_dc_sign_flag [x0][y0][i]	ae(v)
}	
}	
}	
...	
}	

[0169]

[0170] 신택스 엘리먼트 dmm_dc_flag 는 DMM 에 대한 심도 데이터의 잔차 값들이 존재하는지 여부를 표시한다. 1 과 동일한 dmm_dc_flag[x0][y0] 는 dmm_dc_abs[x0][y0][i] 및 dmm_dc_sign_flag[x0][y0][i] 가 존재하는 것을 특정한다. 0 과 동일한 dmm_dc_flag[x0][y0] 는 dmm_dc_abs[x0][y0][i] 및 dmm_dc_sign_flag[x0][y0][i] 가 존재하지 않는 것을 특정한다.

[0171] 잔차 값들이 존재할 때, dmm_dc_abs[x0][y0][i] 및 dmm_dc_sign_flag[x0][y0][i] 의 신택스 엘리먼트들은 DMM 에 대한 심도 데이터의 잔차 값들의 크기 (즉, 절대값) 및 부호를 각각 표시한다. 이 신택스 엘리먼트들은 다음과 같이 DcOffset[x0][y0][i] 를 유도하기 위하여 이용된다:

$$DcOffset[x0][y0][i] = (1 - 2 * dmm_dc_sign_flag[x0][y0][i]) * dmm_dc_abs[x0][y0][i].$$

[0172]

[0173] DMM 을 이용하여 코딩된 각각의 파티션에 대한 델타 DC 값의 유도는 다음과 같다. 비디오 인코더 (20) 와 같은 인코더에서는, 각각의 파티션 i (i=0, 1) 에 대하여, 예측된 DC 값, predDC[i] 가 이웃하는 참조 샘플들을 이용하여 유도된다. 파티션 i 내의 원래의 샘플들의 평균 값은 origDC[i] 로서 계산된다. DLT 가 적용되지 않을 경우, 잔차 값, deltaDC[i] 는 origDC[i] - predDC[i] 로서 유도된다. 이와 다를 경우, DLT 가 적용되고, deltaDC[i] 는 DLT[origDC[i]] - DLT[predDC[i]] 로서 유도된다. 어느 하나의 경우에 있어서, deltaDC[i] 는 인코더 최적화에 기초하여 추가로 조절될 수 있다.

[0174] 비디오 디코더 (30) 와 같은 디코더에서는, 각각의 파티션 i (i=0, 1) 에 대하여, 잔차 값, deltaDC[i] 가 수신되고, 예측된 DC 값, predDC[i] 가 이웃하는 참조 샘플들을 이용하여 유도된다. DLT 가 적용되지 않을 경우, 파티션 i 에서의 각각의 예측 샘플의 값 (즉, 재구성된 심도 값) 은 predDC[i] + deltaDC[i] 로서 유도된다. 이와 다를 경우, DLT 가 적용되고, 파티션 i 에서의 각각의 예측 샘플의 값은 IDLT[DLT[predDC[i]] + deltaDC[i]] 로서 유도된다.

[0175] CCM 을 이용한 델타 DC 코딩과 관련된 선택스 엘리먼트들 및 시맨틱들은 다음과 같이 정의된다.

depth_mode_parameters(x0 , y0 , log2CbSize) {	디스크립터
...	
else if(DepthIntraMode[x0][y0] == INTRA_DEP_CHAIN) {	
edge_dc_flag [x0][y0]	ac(v)
if(edge_dc_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
edge_dc_abs [x0][y0][i]	ac(v)
if(edge_dc_abs[x0][y0][i] != 0)	
edge_dc_sign_flag [x0][y0][i]	ac(v)
}	
}	
}	
...	
}	

[0176]

[0177] 선택스 엘리먼트 edge_dc_flag 는 CCM 에 대한 심도 데이터의 잔차 값들이 존재하는지 여부를 표시한다. 1 과 동일한 edge_dc_flag[x0][y0] 는 edge_dc_abs[x0][y0][i] 및 edge_dc_sign_flag[x0][y0][i] 가 존재하는 것을 특정하고, 0 과 동일한 edge_dc_flag[x0][y0] 는 edge_dc_abs [x0][y0][i] 및 edge_dc_sign_flag[x0][y0][i] 가 존재하지 않는 것을 특정한다.

[0178] 잔차 값들이 존재할 때, edge_dc_abs[x0][y0][i] 및 edge_dc_sign_flag[x0][y0][i] 의 선택스 엘리먼트들은 CCM 에 대한 심도 데이터의 잔차 값들의 크기 (즉, 절대값) 및 부호를 각각 표시한다. 이 선택스 엘리먼트들은 다음과 같이 DcOffset[x0][y0][i] 를 유도하기 위하여 이용된다:

$$DcOffset[x0][y0][i] = (1 - 2 * edge_dc_sign_flag[x0][y0][i]) * edge_dc_abs[x0][y0][i]$$

[0179]

[0180] CCM 을 이용하여 코딩된 각각의 파티션에 대한 델타 DC 값의 유도는 다음과 같다. CCM 에서는, DLT 가 절대로 적용되지 않고, 델타 DC 는 그것이 스케일링 인자 scaleF 에 의해 비디오 인코더 (20) 와 같은 인코더에서

양자화되는 방법으로 코딩되며, 여기서, scaleF 는 최소 값 1 및 최대 값 $2^{(BitDepth_Y-1)}$ 에 의해 $2^{(\frac{QP_Y}{10}-2)}$ 클리핑되고, QP_Y 는 루마 변환 계수 코딩에 대해 적용된 양자화 파라미터를 나타내고, $BitDepth_Y$ 는 루마 샘플 표현에 대해 적용된 비트 심도를 표시한다. 이러한 스케일링 인자는 비디오 디코더 (30) 와 같은 디코더에서 역양자화를 이용하여 다시 보상된다. 이러한 방법으로, CCM 에 대해 시그널링된 델타 DC 의 값은 훨씬 더 작을 수 있다.

[0181]

비디오 인코더 (20) 에서는, 각각의 파티션 i (i=0, 1) 에 대하여, 예측된 DC 값, predDC[i] 가 이웃하는 참조 샘플들을 이용하여 유도된다. 파티션 i 내의 원래의 샘플들의 평균 값은 origDC[i] 로서 계산된다. 각각의 파티션 i 에 대하여, 양자화된 잔차 값, deltaDC[i] 는 (origDC[i] - predDC[i]) / scaleF 로서 유도되고, deltaDC[i] 는 인코더 최적화에 기초하여 추가로 조절될 수 있다.

[0182]

비디오 디코더 (30) 에서는, 각각의 파티션 i 에 대하여, 양자화된 잔차 값, deltaDC[i] 가 수신되고, 잔차 값, deltaDC[i] 는 $deltaDC[i] = deltaDC[i] * scaleF$ 로서 역양자화된다. 그 후에는, 각각의 파티션 i 에 대하여, 예측된 DC 값, predDC[i] 가 이웃하는 참조 샘플들을 이용하여 유도된다. 파티션 i 내의 각각의 예측 샘플의 값 (즉, 재구성된 심도 값) 은 deltaDC[i] 및 predDC[i] 의 합으로서 유도된다. 이 문서에서는, CCM 에서의 델타 DC 값들의 유도에서 포함된 상기 양자화 및 역양자화 프로세스가 진보된 양자화 및 진보된 역양자화로서 지칭된다.

[0183] SDC 모드를 이용한 잔차 코딩과 관련된 신택스 엘리먼트들 및 시맨틱들은 다음과 같이 정의된다.

Depth_mode_parameters(x0 , y0 , log2CbSize) {	디스크립터
...	
else if(SdcFlag[x0][y0]) {	
sdcNumSegments =	
(DepthIntraMode[x0][y0] == INTRA_DEP_SDC_DMM_WFULL) ? 2 : 1	
for(i = 0; i < sdcNumSegments; i++) {	
sdc_residual_flag [x0][y0][i]	ae(v)
if(sdc_residual_flag[x0][y0][i]) {	
sdc_residual_sign_flag [x0][y0][i]	ae(v)
sdc_residual_abs_minus1 [x0][y0][i]	ae(v)
}	
}	
}	

[0184]

[0185] 신택스 엘리먼트 sdc_residual_flag 는 SDC 모드에 대한 심도 데이터의 잔차 값들이 존재하는지 여부를 표시한다. 0 과 동일한 sdc_residual_flag[x0][y0][i] 는 잔차가 세그먼트 i 에 대하여 제로인 것을 특정한다. 1 과 동일한 sdc_residual_flag 는 잔차가 비-제로 (non-zero) 이며 sdc_residual_sign_flag 및 sdc_residual_abs[i] 신택스 엘리먼트들이 세그먼트 i 에 대하여 존재하는 것을 특정한다.

[0186] 잔차 값들이 존재할 때, sdc_residual_abs_minus1[x0][y0][i] 및 sdc_residual_sign_flag[x0][y0][i] 의 신택스 엘리먼트들은 SDC 모드에 대한 심도 데이터의 잔차 값들의 크기 (즉, 절댓값) 및 부호를 각각 표시한다. 이 신택스 엘리먼트들은 다음과 같이 세그먼트 i 에 대하여 SdcResidual[x0][y0][i] 를 유도하기 위하여 이용된다:

$$SdcResidual[x0][y0][i] = (1 - 2 * sdc_residual_sign_flag[x0][y0][i]) * (sdc_residual_mag_minus1[x0][y0][i] + 1)$$

[0187]

[0188] 인터 SDC 모드를 이용한 잔차 코딩과 관련된 신택스 엘리먼트들 및 시맨틱들은 다음과 같이 정의된다.

coding_unit(x0 , y0 , log2CbSize , ctDepth) {	디스크립터
...	
if(DepthFlag && PredMode[x0][y0] != MODE_INTRA && !skip_flag[x0][y0])	
inter_sdc_flag	ae(v)
if(inter_sdc_flag) {	
puNum = (PartMode == PART_2Nx2N) ? 1 : (PartMode == PART_NxN ? 4 : 2)	
for(i = 0; i < puNum; i++) {	
inter_sdc_resi_abs_minus1 [x0][y0][i]	ae(v)
inter_sdc_resi_sign_flag [x0][y0][i]	ae(v)
}	
}	
...	
}	

[0189]

[0190] 신택스 엘리먼트 inter_sdc_flag 는 인터 SDC 모드에 대한 심도 데이터의 잔차 값들이 존재하는지 여부를 표시한다. 1 과 동일한 inter_sdc_flag 는 잔차 블록들의 간략화된 심도 코딩이 현재의 코딩 유닛에 대해 이용되는 것을 특정한다. 0 과 동일한 inter_sdc_flag 는 잔차 블록들의 간략화된 심도 코딩이 현재의 코딩 유닛에 대해 이용되지 않는 것을 특정한다. 존재하지 않을 때, inter_sdc_flag 는 0 과 동일한 것으로 추론된다.

[0191] 잔차 값들이 존재할 때, inter_sdc_resi_abs_minus1[x0][y0][i] 및 inter_sdc_resi_sign_flag[x0][y0][i] 의 신택스 엘리먼트들은 인터 SDC 모드에 대한 심도 데이터의 잔차 값들의 크기 (즉, 절

대값) 및 부호를 각각 표시한다. 이 선택스 엘리먼트들은 다음과 같이 InterSdcResi[x0][y0][i] 를 유도하기 위하여 이용된다:

$$\text{InterSdcResi}[x0][y0][i] = (1 - 2 * \text{inter_sdc_resi_sign_flag}[x0][y0][i]) * (\text{inter_sdc_resi_abs_minus1}[x0][y0][i] + 1)$$

[0192]

[0193]

SDC 모드 및 인터 SDC 모드를 이용하여 코딩된 각각의 파티션에 대한 델타 DC 값의 유도는 다음과 같다. SDC 모드 및 인터 SDC 모드의 양자에서, 각각의 파티션에 대한 델타 DC 값은 DMM 에 대하여 위에서 설명된 것과 유사한 방법으로 유도된다. 비디오 인코더 (20) 와 같은 인코더에서는, 각각의 파티션 i (i=0, 1) 에 대하여, 예측된 DC 값, predDC[i] 가 이웃하는 참조 샘플들을 이용하여 유도된다. 파티션 i 내의 원래의 샘플들의 평균 값은 origDC[i] 로서 계산된다. DLT 가 적용되지 않을 경우, 잔차 값, deltaDC[i] 는 origDC[i] - predDC[i] 로서 유도된다. 이와 다를 경우, DLT 가 적용되고, deltaDC[i] 는 DLT[origDC[i]] - DLT[predDC[i]] 로서 유도된다.

[0194]

비디오 디코더 (30) 와 같은 디코더에서는, 각각의 파티션 i (i=0, 1) 에 대하여, 잔차 값, deltaDC[i] 가 수신되고, 예측된 DC 값, predDC[i] 가 이웃하는 참조 샘플들을 이용하여 유도된다. DLT 가 적용되지 않을 경우, 파티션 i 에서의 각각의 예측 샘플의 값 (즉, 재구성된 심도 값) 은 predDC[i] + deltaDC[i] 로서 유도된다. 이와 다를 경우, DLT 가 적용되고, 파티션 i 에서의 각각의 예측 샘플의 값 (즉, 재구성된 심도 값) 은 IDLT[DLT[predDC[i]] + deltaDC[i]] 로서 유도된다.

[0195]

HEVC 및 3D-HEVC 에서는, 루마 컴포넌트에 대하여, 인트라 예측 샘플, 즉, 잔차 값의 생성을 위해 이용된 이웃하는 참조 샘플들이 비디오 인코더 (20) 와 같은 인코더에서의 생성 프로세스 전에, 그리고 비디오 디코더 (30) 와 같은 디코더에서의 심도 값 재구성 프로세스 전에 필터링된다. 필터링은 주어진 인트라 예측 모드 및 변환 블록 사이즈에 의해 제어된다. 예를 들어, 인트라 예측 모드가 DC 이거나 변환 블록 사이즈가 4x4 와 동일할 경우, 이웃하는 샘플들은 필터링되지 않는다. 또 다른 예로서, 주어진 인트라 예측 모드 및 수직 모드 (또는 수평 모드) 사이의 거리가 미리 정의된 문턱보다 더 클 경우, 필터링 프로세스가 된다. 일부의 예들에서, 미리 정의된 문턱은 변환 블록 사이즈들 8, 16 및 32 의 각각에 대해 각각 7, 1 및 0 이다.

[0196]

이웃하는 샘플 필터링을 위하여, [1, 2, 1] 필터 및 이중-선형 필터가 이용될 수도 있다. 이중-선형 필터링은 HEVC 에서 정의된 바와 같은 몇몇 조건들이 참일 경우에 조건부로 이용된다. 기존의 3D-HEVC 프로세스에서는, 이웃하는 참조 샘플들의 필터링이 DMM 에 대해 적용되는 것이 아니라, CCM 에 대해 적용된다.

[0197]

3D-HEVC 에서의 심도 코딩 모드들의 기존의 코딩 및 시그널링 메커니즘들은 비디오 디코더 (30) 와 같은 디코더에서 몇몇 쟁점들을 만들 수도 있다. 하나의 예로서, (DMM, SDC 모드, 및 CCM 과 같은) 상이한 심도 인트라 코딩 모드들의 사용의 개별적인 시그널링은 디코더 과잉 부담을 증가시킨다. 또 다른 예로서, 심도 인트라 코딩 모드들의 각각을 코딩하기 위하여 상이한 선택스 엘리먼트들이 도입된다. 그러므로, 추가적인 컨텍스트 모델들이 요구되고, 이것은 디코더에서 메모리 복잡성을 증가시킨다. 추가의 예에서, DMM 의 사용은 하나의 CU 에서의 모든 PU 들을 횡단하는 하나의 루프에서 시그널링된다. 게다가, 하나의 CU 에서의 모든 PU 들을 횡단하는 몇몇 루프들은 추가적인 정보를 코딩하기 위하여 이용되고, 이것은 필요한 것이 아니라, 디코더에서 과잉 복잡성을 증가시킨다.

[0198]

이 개시물에서 설명된 기법들은 HEVC 코딩 모드들과는 상이한 3D-HEVC 표준에서의 심도 코딩 모드들을 이용한 3D 비디오 데이터에 대한 심도 데이터의 인코딩 및 디코딩에 관한 것이다. 3D-HEVC 에서 이용가능한 추가적인 심도 인트라 코딩 모드들의 예들은 DMM, SDC 모드 및 CCM 중의 적어도 2 개를 포함한다. 게다가, 추가적인 심도 인터 코딩 모드의 예는 인터 SDC 모드를 포함한다. 이하에서 더욱 상세하게 설명된 기법들은 비디오 인코더 (20) (도 1 및 도 2) 및/또는 비디오 디코더 (30) (도 1 및 도 3) 에 의해 수행될 수도 있고, 이들 양자는 일반적으로 비디오 코더로서 지칭될 수도 있다.

[0199]

제 1 예에서, 이 개시물의 기법들은 심도 모델링 테이블에서 3D 비디오 데이터에 대한 심도 맵들을 코딩하기 위해 이용된 심도 인트라 코딩 모드들을 시그널링하는 것을 포함한다. 심도 모델링 테이블은 HEVC 인트라 코딩 모드들에 대한 선택스로부터 분리되어 있는 추가적인 심도 인트라 코딩 모드들에 대한 선택스를 시그널링한다. 이러한 방법으로, 심도 인트라 코딩 모드들에 대한 추가적인 선택스는 인트라 코딩 모드들에 대한 HEVC 선택스에서 원래의 것과 더 적은 상호작용을 가지도록 설계된다.

- [0200] 게다가, 기법들은 또한, HEVC 인트라 모드들과는 상이한 심도 인트라 코딩 모드들 중의 임의의 것이 3D 비디오 데이터에 대해 되는지 여부를 표시하기 위하여, 플래그와 같은 신택스 엘리먼트를 시그널링하는 것을 포함한다. 현재의 심도 계층이 DMM 을 이용하는지 여부를 표시하기 위하여 비디오 데이터의 비디오 파라미터 세트 (VPS) 에서 vps_dmm_flag 만을 가지는 대신에, 기법들은, DMM, CCM, 및 SDC 모드 중의 적어도 2 개를 포함하는 심도 인트라 코딩 모드들 중의 임의의 것이 인에이블되는지, 또는 심도 인트라 코딩 모드들의 전부가 디스에이블되는지 여부를 표시하기 위하여, 비디오 데이터의 VSP, SPS, 또는 PPS 중의 하나에서 단일 플래그 (single flag) 를 시그널링하는 것을 포함한다. 일부의 예들에서, 단일 플래그는 심도 인트라 코딩 모드들의 전부가 인에이블되는지 여부를 표시할 수도 있다. 이 개시물의 기법들은, HEVC 인트라 코딩 모드들로부터 분리되어 있는 심도 모델링 테이블에서의 심도 인트라 코딩 모드들을 시그널링함으로써, 그리고 심도 인트라 코딩 모드들이 언제 인에이블되는지와, 심도 모델링 테이블이 언제 파싱되어야 하는지를 표시함으로써, 비디오 디코더 (30) 에서 파싱 부담을 감소시킬 수도 있다.
- [0201] 심도 인트라 코딩 모드들이 디스에이블될 때, 비디오 디코더 (30) 는 심도 모델링 테이블을 디코딩하지 않고, 심도 인트라 코딩 모드들과는 상이한 HEVC 인트라 코딩 모드들 중의 하나, 예컨대, HEVC 인트라 코딩 모드 또는 HEVC I_PCM 모드를 선택한다. 심도 인트라 코딩 모드들이 인에이블될 때, 비디오 디코더 (30) 는 심도 데이터를 코딩하기 위하여 이용된 심도 인트라 코딩 모드들 중의 하나를 선택하기 위하여 심도 모델링 테이블을 수신하고 디코딩한다. 예를 들어, 심도 모델링 테이블은 심도 데이터를 디코딩하기 위한, DMM 의 서브-모드들 중의 하나, SDC 모드의 서브-모드들 중의 하나, 또는 CCM 을 표시하는 하나 이상의 신택스 엘리먼트들을 포함할 수도 있다. 일부의 예들에서, DMM 서브-모드들, SDC 서브-모드들 및 CCM 을 표시하는 상이한 신택스 엘리먼트들은 함께 그룹화될 수도 있어서, 상이한 모드들은 하나의 신택스 엘리먼트에 의해 표시된다.
- [0202] 대안적으로, DMM 의 서브-모드들 DMM_1 및 DMM_3 및 CCM 은 HEVC 인트라 코딩 모드들의 일부로서 취급될 수도 있다. 이 경우, 심도 모델링 테이블은 DMM 의 적어도 하나의 서브-모드, 예컨대, DMM_2 및/또는 DMM_4 와, SDC 모드의 3 개의 서브-모드들, 즉, SDCM_DM, SDCM_DMM_1, 및 SDCM_Planar 에 대한 신택스 엘리먼트들을 포함할 수도 있다. CCM 이 HEVC 인트라 코딩 모드로서 정의되는 경우, 기법들은 심도 모델링 테이블로부터 분리되어 있는 체인 코딩 테이블에서의 CCM 에 대한 심도 데이터의 잔차 정보를 시그널링하는 것을 포함한다.
- [0203] 게다가, 몇몇 루프들에서, 모든 PU 들의 dmm_flag, 모든 PU 들의 DMM 서브-모드들, 및 모든 PU 들의 다른 DMM 관련 정보를 별도로 시그널링하는 대신에, CU 신택스 테이블 또는 심도 모델링 테이블은 하나의 PU 의 모든 DMM 정보와 함께, 다음으로, 또 다른 PU 에 대한 정보 등등을 포함하도록 설계될 수도 있다. 그러므로, 현재의 CU 에서 모든 PU 들을 횡단하는 하나의 루프만이 요구된다.
- [0204] 제 2 예에서, 이 개시물의 기법들은 심도 코딩 모드들 중의 2 개 이상에 걸쳐 3D 비디오 데이터에 대한 심도 맵들의 잔차 정보의 시그널링을 통합하는 것을 포함한다. 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트가 정의되며, 여기서, 신택스 엘리먼트들의 동일한 세트는 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시한다. 신택스 엘리먼트들의 세트는 심도 코딩 모드들 중의 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시한다.
- [0205] 신택스 엘리먼트들의 통합된 세트는, 잔차 값들이 DMM, CCM, SDC 모드, 및 인터 SDC 모드 중의 적어도 2 개를 포함하는 심도 코딩 모드들 중의 임의의 것에 대해 코딩되는지 여부를 표시하는 신택스 엘리먼트를 포함할 수도 있다. 신택스 엘리먼트들의 통합된 세트는 또한, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 값들의 절댓값 및 부호를 표시하는 신택스 엘리먼트들을 포함할 수도 있다. 일부의 경우들에는, 심도 데이터를 코딩하기 위해 이용된 심도 인트라 코딩 모드들 중의 하나를 표시하기 위하여, 그리고 심도 데이터의 잔차 값들을 표시하기 위하여, 신택스 엘리먼트들의 통합된 세트가 위에서 설명된 심도 모델링 테이블 내에 포함될 수도 있다.
- [0206] 기존의 3D-HEVC 프로세스에서, DMM 및 CCM 에 대한 잔차 정보의 시그널링은 신택스 엘리먼트들 edge_dc_flag, edge_dc_1_abs, edge_dc_2_abs, dmm_dc_1_abs, dmm_dc_2_abs, edge_dc_1_sign_flag, edge_dc_2_sign_flag, dmm_dc_1_sign_flag 및 dmm_dc_2_sign_flag 에 의한 것이다. 하나의 예에서, 이 개시물에서 설명된 기법들에 따르면, 비디오 디코더 (30) 에서의 파싱 부담과, 잔차 정보 신택스 엘리먼트들에 대해 요구된 컨텍스트 모델들의 수를 감소시키기 위하여, DMM 및 CCM 에 대한 잔차 정보 신택스 엘리먼트들이 통합될 수도 있다.
- [0207] 또 다른 예에서, 기법들은 DMM, CCM, 및 SDC 모드에 대한 잔차 정보 시그널링을 통합한다. 추가의 예에서, 인터 SDC 모드들에 대한 잔차 정보의 시그널링은 또한, DMM, CCM, 및 SDC 모드들에 대한 잔차 정보의 시그널링과 통합될 수도 있다. 일반적으로, 모든 심도 코딩 모드들에 대하여, 잔차가 PU 의 2 개의 파티션들의 각각

또는 PU 의 단지 고유한 파티션에 대해 시그널링될 때, 시그널링 메커니즘은 심도 코딩 모드들이 선택스 엘리먼트들의 동일한 세트를 공유하고 있는 방법으로 통합될 수도 있다.

[0208] 심도 코딩 모드들 중의 2 개 이상에 대한 잔차 정보를 표시하기 위하여 선택스 엘리먼트들의 동일한 세트를 공유함으로써, 선택스 엘리먼트들에 대한 컨텍스트 모델 선택 및 이진화 방법들이 또한 통합될 수도 있다. 예를 들어, 심도 코딩 모드들 중의 2 개 이상에 걸쳐 공유되는 선택스 엘리먼트들의 각각에 대하여 하나의 컨텍스트 모델만이 요구될 수도 있다. 이러한 방법으로, 컨텍스트 모델들의 동일한 세트, 예컨대, 공유된 선택스 엘리먼트들의 각각에 대한 하나가 심도 코딩 모드들 중의 임의의 것에 적용될 수도 있다. 게다가, 단일 컨텍스트 모델 선택 방법은 심도 코딩 모드들 중의 임의의 것에 대해 이용될 수도 있다.

[0209] 기법들에 따르면, 컨텍스트 모델 선택은 현재의 심도 블록의 파티션들의 수에 기초할 수도 있다. 잔차 값들이 인터 SDC 모드에 대해 코딩되는지 여부를 시그널링하는 선택스 엘리먼트에 대하여, 컨텍스트 모델 선택은 현재의 CU 에 대해 적용된 파티션들의 수에 종속될 수도 있다. 다른 한편으로, 잔차 값들이 DMM, CCM, 또는 SDC 모드에 대해 코딩되는지 여부를 시그널링하는 동일한 선택스 엘리먼트에 대하여, 컨텍스트 모델 선택은 현재의 PU 에 대해 적용된 파티션들의 수에 종속될 수도 있다.

[0210] 기존의 3D-HEVC 프로세스에서는, DMM, CCM, 및 SDC 모드에 대한 잔차 값들의 절대값들을 시그널링하는 선택스 엘리먼트들의 이진화가 선택스 엘리먼트들 `edge_dc_abs`, `dmm_dc_abs`, 및 `sdc_residual_abs_minus1` 에 의해 현재 각각 표현된다. 이 개시물의 기법들은 비디오 디코더 (30) 에서 과성 복잡성을 감소시키기 위하여 동일한 이진화 방법을 이용함으로써 이 선택스 엘리먼트들을 통합하는 것을 포함한다. 이 선택스 엘리먼트들의 통합된 이진화 방법은 SDC 모드, 즉, `sdc_residual_abs_minus1`, 또는 DMM 즉, `dmm_dc_abs` 에서의 선택스 엘리먼트들에 대해 기존에 이용된 이진화 방법들 중의 임의의 것과 동일할 수도 있다.

[0211] 인터 SDC 모드에 대한 잔차 값들의 절대값들을 시그널링하는 선택스 엘리먼트들의 이진화는 또한, SDC 모드 또는 DMM 에 대한 이진화 방법과 동일하도록 통합될 수도 있다. 일반적으로, 심도 CU (또는 심도 PU) 를 몇몇 세그먼트들로 파티셔닝하며 하나의 잔차 값만이 각각의 세그먼트에 대해 시그널링되는 모든 심도 코딩 모드들에 대하여, 잔차 값들의 절대값 및 부호를 시그널링하는 선택스 엘리먼트들의 이진화는 비디오 디코더 (30) 에서 과성 복잡성을 감소시키기 위하여 통합될 수도 있다. 이 개시물의 기법들은 심도 코딩 모드들에 대한 잔차 정보 코딩 메커니즘들을 통합함으로써, 디코더 과성 부담을 감소시킬 수도 있고, 요구된 컨텍스트 모델들 및 이진화 방법들의 수를 감소시킬 수도 있다.

[0212] 게다가, 심도 인트라 모드들에 대하여, 이웃하는 참조 샘플들은 잔차 값들로서 지칭된 인트라 예측 샘플들의 생성 동안, 또는 잔차 값들에 기초한 심도 값들의 재구성 동안에 필터링되지 않을 수도 있다. 예를 들어, 이 개시물의 기법들에 따르면, 인트라 예측 샘플 생성 동안의 이웃하는 참조 샘플들의 필터링은 디스에이블될 수도 있고, DMM, CCM, 및 SDC 모드의 서브-모드 SDCM_DMM_1 에서 적용되지 않을 수도 있다.

[0213] 위에서 설명된 바와 같이, 기존의 3D-HEVC 프로세스에서는, 이웃하는 참조 샘플들의 필터링이 SDC 모드의 서브-모드 SDCM_DC 를 포함하는 DC 인트라 예측 모드들에 대해 이미 디스에이블된다. 또 다른 예에서, 인트라 예측 샘플 생성 동안의 이웃하는 참조 샘플들의 필터링은 SDC 모드의 서브-모드 SDCM_Planar 에 대해 추가로 디스에이블될 수도 있다. 이러한 방법으로, 비디오 디코더 (30) 는 이웃하는 참조 샘플들을 필터링하지 않으면서, 수신된 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 심도 코딩 모드들 중의 임의의 것에 따라 비디오 데이터의 심도 값들을 재구성하기 위하여 심도 데이터를 디코딩할 수도 있다. 유사하게, 비디오 인코더 (20) 는 이웃하는 참조 샘플들을 필터링하지 않으면서, 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 심도 코딩 모드들 중의 임의의 것에 따라 심도 데이터의 잔차 값들을 생성할 수도 있다.

[0214] 대안으로서, DMM, CCM, 및 SDC 모드의 서브 모드들 SDCM_DMM_1 및 SDCM_Planar 에 대하여, 인트라 예측 샘플 생성 동안의 이웃하는 참조 샘플들의 필터링은 항상 적용될 수도 있다. 이 경우, 필터링 방법들은 적용된 심도 코딩 모드에 종속적일 수도 있다.

[0215] 또한, 기법들은 심도 록업 테이블 (DLT) 을 이용하여, DMM, SDC 모드, 인터 SDC 모드, 및 CMM 중의 임의의 것에 대한 파티션에 대해 하나의 단일 송신된 잔차 값에 기초하여 심도 값들을 재구성하기 위하여 이용된 프로세스를 통합하는 것을 포함한다. 심도 코딩 모드들 중의 임의의 것에 대하여, 비디오 디코더 (30) 에서는, DLT 가 예측 값 인덱스들을 파티션의 예측 값들, 즉, 재구성된 심도 값들로 변환하기 위하여 이용될 수도 있으며, 여기서, 예측 값 인덱스는 송신된 잔차 값 및 예측된 DC 값 인덱스의 합이다. CCM 에 따라 파티션에 대한 하나의 단일 송신된 잔차 값 및 예측 값들에 기초하여 잔차 정보를 생성할 때, 단일 잔차 값의 비디오 인코더 (20)

에서의 진보된 양자화 및 비디오 디코더 (30) 에서의 진보된 역양자화는 제거될 수도 있다. CCM 으로부터 진보된 양자화를 제거함으로써, 디코딩 프로세스는 DLT 를 이용하기 위한 DMM 및 SDC 모드와 통합될 수도 있다.

[0216] 이러한 방법으로, 비디오 디코더 (30) 는 코딩 파티션에 대해 수신된 단일 잔차 값과, 이웃하는 참조 샘플들로부터 유도된 예측된 값을 식별하는 DLT 인덱스 값과의 합과 동일한 인덱스 값을 갖는 값들을 록업하기 위한 DLT 를 이용하여 코딩 파티션 내의 심도 값들을 재구성한다. 비디오 디코더 (30) 는 수신된 단일 잔차 값의 진보된 역양자화를 수행하지 않는다. 유사하게, 비디오 인코더 (20) 는, 코딩 파티션에 대한 단일 잔차 값이 코딩 파티션 내의 원래의 심도 값들의 평균 값을 식별하는 제 1 DLT 인덱스 값과, 이웃하는 참조 샘플들로부터 유도된 예측된 값을 식별하는 제 2 DLT 인덱스 값과의 차이와 동일한 DLT 를 이용하여 코딩 파티션에 대한 단일 잔차 값을 생성한다. 비디오 디코더 (20) 는 생성된 단일 잔차 값의 진보된 양자화를 수행하지 않는다.

[0217] 3D-HEVC 에서의 3D 비디오 데이터의 심도 데이터의 심도 코딩 모드들 시그널링의 몇몇 상세한 구현 예들이 이하에서 설명된다.

[0218] 이 개시물의 기법들의 제 1 예에서, 비디오 인코더 (20) 는 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 플래그를 인코딩하고, 심도 데이터를 인코딩하기 위해 이용된 심도 인트라 코딩 모드들 중의 선택된 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 인코딩한다. 시그널링된 플래그에 의해 표시된 바와 같이, 심도 인트라 코딩 모드들이 인에이블될 때, 비디오 디코더 (30) 는 심도 데이터를 디코딩하기 위한 심도 인트라 코딩 모드를 선택하기 위하여 심도 모델링 테이블을 수신하고 디코딩한다.

[0219] 이 예에서는, SDC 모드, DMM, 또는 HEVC 인트라 코딩 모드 (CCM 을 포함함) 의 사용을 표시하기 위하여 심도 코딩 모드에서의 다수의 신택스 엘리먼트들을 이용하는 대신에, 하나의 신택스 엘리먼트는 DMM, SDC 모드 및 CCM 중의 적어도 2 개를 포함하는 심도 인트라 코딩 모드들의 가능성 (enablement) 을 표시한다. 심도 인트라 코딩 모드들 중의 선택된 하나를 표시하는 하나 이상의 신택스 엘리먼트들은 HEVC 신택스와 분리되어 있는 심도 모델링 테이블 내에 포함되고, 일부의 조건들은 CU 에 대한 HEVC 신택스로부터 제거될 수도 있다.

[0220] 심도 모델링 테이블을 시그널링하는 제 1 예의 기법의 제 1 대안에서는, 하나의 신택스 엘리먼트, depth_intra_mode 가 CU 신택스 테이블 및 심도 모델링 테이블에서 심도 뷰들의 인트라 코딩 모드들을 표시하기 위하여 추가되고, CCM 은 HEVC 인트라 예측 모드들의 일부가 되는 대신에, 하나의 추가적인 심도 인트라 코딩 모드로서 시그널링된다. 다른 예들에서는, 하나를 초과하는 신택스 엘리먼트가 CU 신택스 테이블 및 심도 모델링 테이블에서 심도 인트라 코딩 모드들을 표시할 수도 있다. 3D-HEVC 표준에 대한 추가들은 이하에서 회색으로 강조표시되고, 3D-HEVC 표준으로부터의 삭제들은 지움표시를 이용하여 표시된다.

[0221] **G.7.3.9.1 일반적인 코딩 유닛 신택스**

coding_unit(x0, y0, log2CbSize , ctDepth) {	디스크립터
if(transquant_bypass_enable_flag) {	
cu_transquant_bypass_flag	ac(v)
}	
if(slice_type != I)	
skip_flag[x0][y0]	ac(v)

[0222]

if(skip_flag[x0][y0])	
prediction_unit(x0, y0, log2CbSize)	
if (icEnableFlag)	
ic_flag	ae(v)
else {	
nCbS = (1 << log2CbSize)	
if(slice_type != I)	
pred_mode_flag	ae(v)
if(PredMode == MODE_INTRA && DepthFlag)	
sde_flag [x0][y0]	ae(v)
if(sde_flag[x0][y0]) {	
sde_pred_mode	ae(v)
if(sde_pred_mode == 1)	
wedge_full_tab_idx [x0][y0]	ae(v)
} else {	
if((PredMode[x0][y0] != MODE_INTRA log2CbSize == Log2MinCbSize) && !predPartModeFlag)	
part_mode	ae(v)
if(PredMode[x0][y0] == MODE_INTRA) {	
if(PartMode == PART_2Nx2N && pcm_enabled_flag && log2CbSize >= Log2MinIPCMCUSize && log2CbSize <= Log2MaxIPCMCUSize)	
pcm_flag	ae(v)
if(pcm_flag) {	
num_subsequent_pcm	tu(3)
NumPCMBlock = num_subsequent_pcm + 1	
while(!byte_aligned())	
pcm_alignment_zero_bit	f(1)
pcm_sample(x0, y0, log2CbSize)	
} else {	
pbOffset = (PartMode == PART_NxN) ? (nCbS / 2) : 0	
if(vps_dmm_flag[nuh_layer_id] && log2CbSize <= Log2MaxDmmCbSize) {	
log2PbSize = log2CbSize - (PartMode == PART_NxN ? 1 : 0)	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
dmm_flag [x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if(dmm_flag[x0 + i][y0 + j])	
dmm_mode [x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if(dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULL dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULLDELTA))	
wedge_full_tab_idx [x0 + i][y0 + j]	ae(v)

[0223]

for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(dmm_flag[x0+i][y0+j] && (dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIR dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIRDELTA))	
wedge_predtex_tab_idx[x0+i][y0+i]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(dmm_flag[x0+i][y0+j] && (dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIR dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIRDELTA))	
dmm_delta_end_flag[x0+i][y0+i]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(dmm_delta_end_flag[x0+i][y0+i])	
dmm_delta_end_abs_minus[x0+i][y0+i]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(dmm_delta_end_flag[x0+i][y0+i])	
dmm_delta_end_sign_flag[x0][y0]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
DmmDeltaFlag[x0+i][y0+i] = (dmm_flag[x0+i][y0+j] && (dmm_mode[x0+i][y0+i] == MODE_DMM_WFULLDELTA dmm_mode[x0+i][y0+i] == MODE_DMM_WPREDDIRDELTA dmm_mode[x0+i][y0+i] == MODE_DMM_WPREDTXDELTA dmm_mode[x0+i][y0+i] == MODE_DMM_CPREDTEXDELTA))	
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(DmmDeltaFlag[x0+i][y0+i])	
dmm_de_1_abs[x0+i][y0+i]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(DmmDeltaFlag[x0+i][y0+j] && dmm_de_1_abs[x0+i][y0+i] != 0)	
dmm_de_1_sign_flag[x0+i][y0+i]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(DmmDeltaFlag[x0+i][y0+j])	
dmm_de_2_abs[x0+i][y0+i]	ae(v)
for(j=0; j<=pbOffset; j=j+pbOffset)	
for(i=0; i<=pbOffset; i=i+pbOffset)	
if(DmmDeltaFlag[x0+i][y0+j] && dmm_de_2_abs[x0+i][y0+i] != 0)	
dmm_de_2_sign_flag[x0+i][y0+i]	ae(v)

[0224]

<pre> } else { /* PART_NxN */ prediction_unit(x0, y0, nCbS / 2, nCbS / 2) prediction_unit(x0 + (nCbS / 2), y0, nCbS / 2, nCbS / 2) prediction_unit(x0, y0 + (nCbS / 2), nCbS / 2, nCbS / 2) prediction_unit(x0 + (nCbS / 2), y0 + (nCbS / 2), nCbS / 2, nCbS / 2) } } } if (icEnableFlag) ic_flag ac(v) if(!pcm_flag) { if(PredMode[x0][y0] != MODE_INTRA && !(PartMode == PART_2Nx2N && merge_flag[x0][y0])) no_residual_syntax_flag ac(v) if(!no_residual_syntax_flag !sde_flag) { MaxTrafoDepth = (PredMode[x0][y0] == MODE_INTRA ? max_transform_hierarchy_depth_intra + IntraSplitFlag : max_transform_hierarchy_depth_inter) transform_tree(x0, y0 x0, y0, log2CbSize, 0, 0) } if(sde_flag) { sdeNumSegments = (sde_pred_mode == -1) ? 2 : 1 for (i = 0; i < sdeNumSegments; i++) { sde_residual_flag[x0][y0][i] ae(v) if(sde_residual_flag[x0][y0][i]) { sde_residual_sign_flag[x0][y0][i] ae(v) sde_residual_abs_minus1[x0][y0][i] ae(v) } } } } } } } </pre>	
---	--

[0226]

[0227] 새롭게 도입된 신택스 "depth_modeling_table" 는 다음과 같이 정의될 수도 있다.

[0228] H.xxxx 심도 모델링 테이블 신택스

[0229]

depth_modeling_table(x0 , y0) {	디스크립터
depth_intra_mode [x0][y0]	ae(v)
dmmFlag[x0][y0] = (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR depth_intra_mode[x0][y0] == MODE_DMM_WPREDTX depth_intra_mode[x0][y0] == MODE_DMM_CPREDTEX)	

sdcFlag[x0][y0] = (depth_intra_mode[x0][y0] == SDCM_DMM_1 depth_intra_mode[x0][y0] == SDCM_DC depth_intra_mode[x0][y0] == SDCM_Planar)	
if (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == SDCM_DMM_1)	
wedge_full_tab_idx[x0][y0]	ae(v)
else if(depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEX)	
wedge_predtex_tab_idx[x0][y0]	ae(v)
else if (depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR) {	
dmm_delta_end_flag[x0][y0]	ae(v)
if (dmm_delta_end_flag[x0][y0]) {	
dmm_delta_end_abs_minus1[x0][y0]	ae(v)
dmm_delta_end_sign_flag[x0][y0]	ae(v)
}	
} else if (depth_intra_mode[x0][y0] == CCM) {	
edge_start_left_flag[x0][y0]	ae(v)
edge_start_position[x0][y0]	ae(v)
num_edge_codes_minus1[x0][y0]	ae(v)
for(k = 0; k <= num_edge_codes_minus1; k++)	
edge_code[k]	ae(v)
}	
if(dmmFlag[x0][y0]) {	
dmm_delta_flag[x0][y0]	ae(v)
if (dmm_delta_flag [x0][y0])	
for(i = 0; i < 2; i ++) {	
dmm_dc_abs[x0][y0][i]	ae(v)
if (dmm_dc_abs[x0][y0][i])	
dmm_dc_sign_flag[x0][y0][i]	ae(v)
}	
}	
else if (depth_intra_mode[x0][y0] == CCM) {	
edge_dc_flag[x0][y0]	ae(v)
if(edge_dc_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
edge_dc_abs[x0][y0][i]	ae(v)
if(edge_dc_abs[x0][y0][i] != 0)	
edge_dc_sign_flag[x0][y0][i]	ae(v)
}	
}	
}	
else if(sdcFlag[x0][y0]) {	
sdcNumSegments = depth_intra_mode[x0][y0] == SDCM_DMM_1 ? 2 : 1	
for (i = 0; i < sdcNumSegments ; i++) {	
sdc_residual_flag[x0][y0][i]	ae(v)
if(sdc_residual_flag [x0][y0][i]) {	
sdc_residual_sign_flag[x0][y0][i]	ae(v)
sdc_residual_abs_minus1[x0][y0][i]	ae(v)
}	
}	
}	
}	

[0230]

[0231] CU 신택스 테이블 및 심도 모델링 테이블에 대한 시맨틱들은 다음과 같이 정의된다.

[0232] DepthModesFlag 는 VPS 에서의 각각의 심도 계층에 대해 시그널링된 신택스 엘리먼트 depth_modes_flag 와 동일하게 설정될 수도 있다. 1 과 동일한 depth_modes_flag[i] 는 DMM, CCM, 및 SDC 모드가 i 와 동일한 layer_id 를 갖는 심도 계층에 대해 인에이블되는 것을 표시한다. 0 과 동일한 depth_modes_flag[i] 는 DMM, CCM, 및 SDC 모드가 i 와 동일한 layer_id 를 갖는 심도 계층에 대해 디스에이블되는 것을 표시한다.

[0233] 1 과 동일한 no_residual_syntax_flag (HEVC 사양에서의 rqt_root_cbf) 는 transform_tree() syntax 가 현재의 코딩 유닛에 대해 존재하는 것을 특정한다. 0 과 동일한 rqt_root_cbf 는 transform_tree() 신택스가 현재의 코딩 유닛에 대해 존재하지 않는 것을 특정한다. rqt_root_cbf 가 존재하지 않고 sdcFlag[x0][y0] 가 0 과 동일할 때, 그 값은 1 과 동일한 것으로 추론된다. rqt_root_cbf 가 존재하지 않고 sdcFlag[x0][y0] 가 1 과 동일할 때, 그 값은 0 과 동일한 것으로 추론된다.

[0234] 대안적으로, no_residual_syntax_flag 의 시맨틱들 및 SDC 모드에 대한 잔차 정보 (sdc_residual_flag,

sdc_residual_sign_flag 및 sdc_residual_abs_minus1 을 포함함) 의 시그널링은 변화되지 않은 채로 유지된다.

이 경우, 위에서 제시된 심도 모델링 테이블의 선택스의 최후 부분 ("else if(sdcFlag[x0][y0]) {" 로 시작하고 테이블의 최후에서 두 번째 라인에서 "}" 로 종료함) 이 제거될 수도 있다.

[0235] intra_chroma_pred_mode[x0][y0] 는 크로마 샘플들에 대한 인트라 예측 모드를 특정한다. 어레이 인덱스들 x0, y0 는 픽처의 상부-좌측 루마 샘플과 관련하여 고려된 예측 블록의 상부-좌측 루마 샘플의 로케이션 (x0, y0) 을 특정한다. 현재의 뷰 컴포넌트가 심도 뷰 컴포넌트일 때, 이 선택스는 존재하지 않는다. 대안적으로, SDC 모드가 현재의 코딩 유닛에 대해 적용될 때, 이 선택스는 존재하지 않는다.

[0236] depth_intra_mode[x0][y0] 는 이하의 표 1 에서 도시된 값들 중의 하나일 수도 있다. 3D-HEVC 표준에 대한 추가들은 이하에서 회색으로 강조표시되고, 3D-HEVC 표준으로부터의 삭제들은 지움표시를 이용하여 표시된다.

[0237] 표 1 : depth_intra_mode 의 해독

depth_intra_mode	DepthIntraMode	빈 스트링				
		(cLog2CbSize == 3 && partition size == SIZE_2Nx2N) (cLog2CbSize > 3 && cLog2CbSize < 6)		cLog2CbSize == 3 && partition size == SIZE_NxN		cLog2CbSize == 6
		프리픽스	서브픽스	프리픽스	서브픽스	
0	SDCM Planar	0	0	-	-	0
1	HEVC 인트라 (CGM 포함) (HEVC_INTRA_PRED)	0	1 0	0	-	1 0
2	SDCM DMM 1	0	1 1	-	-	-
3	DMM 1 (MODE_DMM_WFULL)	1	0 0	1	0	-
4	DMM 4 (MODE_DMM_CPREDTEX)	1	0 1	-	-	-
5	DMM 3 (MODE_DMM_WFULL)	1	1 0	1	1 0	-
6	SDCM DC	1	1 1 0	-	-	1 1
7	DMM 2 (MODE_DMM_WPREDDIR)	1	1 1 1 1	-	-	-
8	CCM	1	1 1 1 0	1	1 1	-
선택스 인덱스 오프셋		3	4,5,6,6	0	1,2	6,7

[0238] 일부의 대안들에서는, DepthIntraMode 를 제시하기 위한 다른 빈 스트링 (bin string) 들이 적용될 수도 있다.

게다가, 현재의 모드들이 DMM 일 때, 예컨대, depth_intra_mode 가 SDCM_DMM_1 및 DMM_1 과 동일할 때, prev_intra_luma_pred_flag 가 존재한다. 대안적으로, 표 1 에서, 서브-모드 DMM_2 는 제거될 수도 있다.

또한, 표 1 에서, 심도 인트라 코딩 모드들의 임의의 다른 순서가 적용될 수도 있다.

[0240] 추가의 대안으로서, 서브-모드 DMM_2 는 CCM 에 의해 대체될 수도 있고, 상기 표 1 은 이하의 표 2 에서와 같이 다시 정의될 수도 있다.

[0241] 표 2 : depth_intra_mode 의 해독

depth_intra_mode	DepthIntraMode	빈 스트링				
		(cLog2CbSize == 3 && partition size == SIZE_2Nx2N) (cLog2CbSize > 3 && cLog2CbSize < 6)		cLog2CbSize == 3 && partition size == SIZE_NxN		cLog2CbSize == 6
		프리픽스	서브픽스	프리픽스	서브픽스	
0	SDCM Planar	0	0	-	-	0
1	HEVC 인트라 (CGM 포함) (HEVC_INTRA_PRED)	0	1 0	0	-	1 0
2	SDCM DMM 1	0	1 1	-	-	-
3	DMM 1 (MODE_DMM_WFULL)	1	0 0	1	0	-
4	DMM 4 (MODE_DMM_CPREDTEX)	1	0 1	-	-	-
5	DMM 3 (MODE_DMM_WFULL)	1	1 0	1	1 0	-
6	SDCM DC	1	1 1 0	-	-	1 1
7	CCM	1	1 1 1	-	-	-

[0242] 서브-모드 DMM_2 가 CCM 과 대체되는 경우, 상기 표 2 에서 도시된 바와 같이, 심도 모델링 테이블의 선택스는 도시된 바와 같이 라인들을 제거함으로써 수정될 수도 있다. 위에서 제시된 심도 모델링 테이블로부터의 삭제들은 지움표시를 이용하여 표시된다.

[0244] H.xxxx 심도 모델링 테이블 선택스

depth_modeling_table(x0 , y0) {	디스크립터
depth_intra_mode [x0][y0]	ae(v)
dmmFlag[x0][y0] = (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEX depth_intra_mode[x0][y0] == MODE_DMM_CPREDTEX)	
sdcFlag[x0][y0] = (depth_intra_mode[x0][y0] == SDCM_DMM_1 depth_intra_mode[x0][y0] == SDCM_DC depth_intra_mode[x0][y0] == SDCM_Planar)	
if (depth_intra_mode[x0][y0] == MODE_DMM_WFULL)	
wedge_full_tab_idx [x0][y0]	ae(v)
else if (depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEX)	
wedge_predtex_tab_idx [x0][y0]	ae(v)
— else if (depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR) {	
— dmm_delta_end_flag [x0][y0]	ae(v)
— if (dmm_delta_end_flag[x0][y0]) {	
— dmm_delta_end_abs_minus1 [x0][y0]	ae(v)
— dmm_delta_end_sign_flag [x0][y0]	ae(v)
— }	
— } else if (depth_intra_mode[x0][y0] == CCM) {	
edge_start_left_flag [x0][y0]	ae(v)
edge_start_position [x0][y0]	ae(v)
num_edge_codes_minus1 [x0][y0]	ae(v)
for(k = 0; k <= num_edge_codes_minus1; k++)	
edge_code [k]	ae(v)
}	
...	
}	

[0245]

[0246] 위에서 제시된 수정된 심도 모델링 테이블에 대한 시맨틱들은 다음과 같이 정의된다.

[0247] 0 과 동일한 dmm_delta_flag[x0][y0] 는 DC 오프셋 값들이 이용불가능한 것을 특정하고, 1 과 동일한 dmm_delta_flag[x0][y0] 는 DC 오프셋 값들이 이용가능한 것을 특정한다. dmm_delta_flag[x0][y0] 가 1 과 동일할 때, 각각의 파티션에 대한 DC 오프셋 값들 즉, 잔차 값들은 dmm_dc_abs 및 dmm_dc_sign_flag 에 의해 추가로 시그널링된다.

$dmm_dc_1_sign_flag[x0][y0]$ 는 다음과 같이 $QuantDCOffsetP0[x0][y0]$ 및 $QuantDCOffsetP1[x0][y0]$ 값을 유도하기 위하여 이용된다:

$$QuantDCOffsetP0[x0][y0] = (1 - 2 * dmm_dc_0_sign_flag[x0][y0]) * dmm_dc_0_abs[x0][y0] \quad (G-19)$$

$$QuantDCOffsetP1[x0][y0] = (1 - 2 * dmm_dc_1_sign_flag[x0][y0]) * dmm_dc_1_abs[x0][y0] \quad (G-20)$$

0 과 동일한 $edge_dc_flag[x0][y0]$ 는 역양자화된 오프셋이 이용되지 않는 것을 특징한다. 1 과 동일한 $edge_dc_flag[x0][y0]$ 는 역양자화된 오프셋이 이용되는 것을 특징한다. $edge_dc_0_abs[x0][y0]$, $edge_dc_0_sign_flag[x0][y0]$, $edge_dc_1_abs[x0][y0]$, $edge_dc_1_sign_flag[x0][y0]$ 는 다음과 같이 $QuantDCOffsetP0[x0][y0]$ 및 $QuantDCOffsetP1[x0][y0]$ 값을 유도하기 위하여 이용된다.

$$QuantDCOffsetP0[x0][y0] = (1 - 2 * edge_dc_0_sign_flag[x0][y0]) * edge_dc_1_abs[x0][y0] \quad (G-21)$$

$$QuantDCOffsetP1[x0][y0] = (1 - 2 * edge_dc_0_sign_flag[x0][y0]) * edge_dc_1_abs[x0][y0] \quad (G-22)$$

[0248]

[0249] $dmm_dc_abs[x0][y0][i]$ 및 $dmm_dc_sign_flag[x0][y0][i]$ 은 다음과 같이 $QuantDCOffsetPi[x0][y0]$ 값을 유도하기 위하여 이용된다.

$$QuantDCOffsetP0[x0][y0] = (1 - 2 * dmm_dc_sign_flag[x0][y0][0]) * dmm_dc_abs[x0][y0][0] \quad (H-x)$$

$$QuantDCOffsetP1[x0][y0] = (1 - 2 * dmm_dc_sign_flag[x0][y0][1]) * dmm_dc_abs[x0][y0][1] \quad (H-x)$$

[0250]

[0251] $edge_dc_abs[x0][y0][i]$ 및 $edge_dc_sign_flag[x0][y0][i]$ 및 $DcOffsetPi[x0][y0]$ 값은 다음과 같다.

$$DcOffsetP0[x0][y0] = (1 - 2 * edge_dc_sign_flag[x0][y0][0]) * edge_dc_abs[x0][y0][0] \quad (H-X)$$

$$DcOffsetP1[x0][y0] = (1 - 2 * edge_dc_sign_flag[x0][y0][1]) * edge_dc_abs[x0][y0][1] \quad (H-X)$$

[0252]

[0253] 선택스 엘리먼트 $depth_intra_mode$ 를 디코딩할 때, 이하의 표 3 에서 도시된 다음의 컨텍스트 모델들은 제 1 및/또는 제 2 번을 코딩하기 위하여 적용될 수도 있다:

[0254] 표 3 : 좌측 및 상부 선택스 엘리먼트들을 이용한 ctxIdxInc 의 사양

선택스 엘리먼트 (depth_intra_m ode)	condL	condA	ctxIdxInc
제 1 빈	depth_intra_mode [xL][yL] < 3	depth_intra_mode[xA][yA] < 3	(condL && availableL) + (condA && availableA)
제 2 빈	sdc_flag[xL][yL]	sdc_flag[xA][yA]	(condL && availableL) + (condA && availableA)

[0255]

[0256]

표 3 에서 도시된 컨텍스트 모델 사양에 대한 하나의 대안에서는, 제 2 빈을 코딩하기 위한 condL 및 condA 의 사용은 제 1 빈이 0 과 동일할 때, 즉, (depth_intra_mode < 3) 에서 적용되기만 할 수도 있다. 또 다른 대안에서는, 하나의 컨텍스트만이 선택스 엘리먼트 depth_intra_mode 의 제 1 의 2 개의 빈들의 각각에 대해 이용될 수도 있다. 추가의 대안에서는, 하나의 컨텍스트만이 제 1 빈에 대하여 이용될 수도 있고, 제 2 빈은 우회 코딩될 수도 있다. 선택스 엘리먼트 depth_intra_mode 의 나머지 빈들에 대하여, 하나의 컨텍스트는 이들의 각각에 대하여 이용될 수도 있다. 다른 예들에서, depth_intra_mode 의 나머지 빈들은 우회 코딩될 수도 있다. 선택스 엘리먼트 depth_intra_mode 를 코딩하기 위해 이용된 컨텍스트들의 초기화는 슬라이스 타입에 종속적일 수도 있다.

[0257]

심도 모델링 테이블을 시그널링하는 제 1 예의 기법의 제 2 대안은 위에서 설명된 제 1 대안과 유사하다. 그러나, 제 2 대안에서는, DMM 및 CCM 심도 인트라 코딩 모드 및 연관된 잔차 정보의 시그널링, 즉, 심도 모델링 테이블에서의 선택스 엘리먼트들 dmm_delta_flag, edge_dc_flag, edge_dc_abs, dmm_dc_abs, edge_dc_sign_flag, 및 dmm_dc_sign_flag 를 이용하는 것이 다음과 같이 추가로 통합될 수도 있다. 위에서 설명된 제 1 대안을 통한 심도 모델링 테이블에 대한 추가들은 회색으로 강조표시되고, 삭제들은 지움표시를 이용하여 표시된다.

[0258] H.xxxx 심도 모델링 테이블 선택스

depth_modeling_table(x0 , y0) {	디스크립터
depth_intra_mode[x0][y0]	ae(v)
dmmFlag[x0][y0+j] = (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR depth_intra_mode[x0][y0] == MODE_DMM_WPREDTX depth_intra_mode[x0][y0] == MODE_DMM_CPREDTEX)	
...	
if(dmmFlag[x0][y0] depth_intra_mode[x0][y0] == CCM) {	
dmm_cmm_delta_flag[x0][y0]	ae(v)
if (dmm_cmm_delta_flag [x0][y0]) {	
for(i = 0; i < 2; i++) {	
dmm_cmm_dc_abs [x0][y0][i]	ae(v)
if (dmm_cmm_dc_abs [x0][y0][i] != 0)	
dmm_cmm_dc_sign_flag[x0][y0][i]	ae(v)
}	
}	
else if(sdcFlag[x0][y0]) {	
sdcNumSegments = depth_intra_mode[x0][y0] == SDCM_DMM_1 ? 2 : 1	
for (i = 0; i < sdcNumSegments ; i++) {	
sdc_residual_flag[x0][y0][i]	ae(v)
if(sdc_residual_flag [x0][y0][i]) {	
sdc_residual_sign_flag[x0][y0][i]	ae(v)
sdc_residual_abs_minus1[x0][y0][i]	ae(v)
}	
}	
— else if(depth_intra_mode[x0][y0] == CCM) {	
edge_de_flag[x0][y0]	ae(+)
if(edge_de_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
edge_de_abs[x0][y0][i]	ae(+)
if(edge_de_abs[x0][y0][i] != 0)	
edge_de_sign_flag[x0][y0][i]	ae(+)
}	
}	
}	
}	

[0259]

[0260] 수정된 심도 모델링 테이블에 대한 시맨틱들은 다음과 같이 정의된다.

[0261] 0 과 동일한 dmm_cmm_delta_flag[x0][y0] 는 DC 오프셋 값들이 이용불가능한 것을 특정하고, 1 과 동일한 dmm_cmm_delta_flag[x0][y0] 는 DC 오프셋 값들이 CCM 또는 DMM 코딩된 PU 들에 대해 이용가능한 것을 특정한다. dmm_cmm_delta_flag[x0][y0] 가 1 과 동일할 때, 각각의 파티션에 대한 DC 오프셋 값들은 dmm_cmm_dc_abs 및 dmm_cmm_dc_sign_flag 에 의해 추가로 시그널링된다.

~~0 과 동일한 dmm_delta_flag[x0][y0] 는 DC 오프셋 값들이 이용불가능한 것을 특정하고, 1 과 동일한 dmm_delta_flag[x0][y0] 는 DC 오프셋 값들이 이용가능한 것을 특정한다. dmm_delta_flag[x0][y0] 가 1 과 동일할 때, 각각의 파티션에 대한 DC 오프셋 값들은 dmm_dc_abs 및 dmm_dc_sign_flag 에 의해 추가로 시그널링된다. 0 과 동일한 edge_de_flag[x0][y0] 는 역양자화된 오프셋이 이용되지 않는 것을 특정한다. 1 과 동일한 edge_de_flag[x0][y0] 는 역양자화된 오프셋이 이용되는 것을 특정한다.~~

[0262]

[0263] dmm_cmm_dc_abs[x0][y0][i] 및 dmm_cmm_dc_sign_flag[x0][y0][i] 는 다음과 같이 QuantDCOffsetPi[x0][y0] 값을 유도하기 위하여 이용되고, 여기서, i 는 0 내지 1 이다:

$$\text{QuantDCOffsetPi}[x0][y0] = (1 - 2 * \text{dmm_cmm_dc_sign_flag}[x0][y0][1]) * \text{dmm_cmm_dc_abs}[x0][y0][1] \quad (\text{H-x})$$

~~$\text{dmm_dc_abs}[x0][y0][i]$ 및 $\text{dmm_dc_sign_flag}[x0][y0][i]$ 는 다음과 같이 $\text{QuantDCOffsetPi}[x0][y0]$ 값을 유도하기 위하여 이용된다:~~

$$\text{QuantDCOffsetP0}[x0][y0] = (1 - 2 * \text{dmm_dc_sign_flag}[x0][y0][0]) * \text{dmm_dc_abs}[x0][y0][0] \quad (\text{H-x})$$

$$\text{QuantDCOffsetP1}[x0][y0] = (1 - 2 * \text{dmm_dc_sign_flag}[x0][y0][1]) * \text{dmm_dc_abs}[x0][y0][1] \quad (\text{H-x})$$

~~$\text{edge_dc_abs}[x0][y0][i]$ 및 $\text{edge_dc_sign_flag}[x0][y0][i]$ 및 $\text{DeOffsetPi}[x0][y0]$ 값은 다음과 같다.~~

$$\text{DeOffsetP0}[x0][y0] = (1 - 2 * \text{edge_dc_sign_flag}[x0][y0][0]) * \text{edge_dc_abs}[x0][y0][0] \quad (\text{H-X})$$

$$\text{DeOffsetP1}[x0][y0] = (1 - 2 * \text{edge_dc_sign_flag}[x0][y0][1]) * \text{edge_dc_abs}[x0][y0][1] \quad (\text{H-X})$$

심도 모델링 테이블을 시그널링하는 제 1 예의 기법의 제 3 대안은 위에서 설명된 제 1 대안과 유사하다. 그러나, 제 3 대안에서는, 이용가능한 심도 인트라 모드들의 수가 기존의 3D-HEVC 표준으로부터 변화되지 않은 채로 유지되는 반면, 하나의 선택스 엘리먼트, 즉, `depth_intra_mode` 는 CU 선택스 테이블 및 심도 모델링 테이블에서의 하나의 PU 의 인트라 코딩 모드를 시그널링하기 위하여 이용된다. 심도 인트라 코딩 모드들, 즉, DMM 및 SDC 모드와 관련된 추가적인 선택스 엘리먼트들은 CU 선택스 테이블에서 HEVC 인트라 코딩 모드 선택스로부터 분리되어 있는 심도 모델링 테이블에 더해진다. 이 예에서는, CCM 이 HEVC 인트라 코딩 모드들의 일부로서 정의되므로, 심도 모델링 테이블로부터 분리되어 있는 체인 코딩 선택스 테이블이 도입된다. 위에서 설명된 제 1 대안을 통한 CU 및 심도 모델링 테이블에 대한 추가들은 회색으로 강조표시되고, 삭제들은 지움표시를 이용하여 표시된다.

[0267] G.7.3.9.1 일반적인 코딩 유닛 선택스

coding_unit(x0, y0, log2CbSize , ctDepth) {	디스크립터
if(transquant_bypass_enable_flag) {	
cu_transquant_bypass_flag	ae(v)
}	
...	
if(pcm_flag) {	
num_subsequent_pcm	tu(3)
NumPCMBlock = num_subsequent_pcm + 1	
while(!byte_aligned())	
pcm_alignment_zero_bit	f(1)
pcm_sample(x0, y0, log2CbSize)	
} else {	
pbOffset = (PartMode == PART_NxN) ? (nCbS / 2) : 0	
log2PbSize = log2CbSize - (PartMode == PART_NxN ? 1 : 0)	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(DepthModesFlag)	
depth_modeling_table(x0 + i , y0 + j)	
if(depth_intra_mode[x0 + i][y0 + j] == HEVC_INTRA_PRED)	
prev_intra_luma_pred_flag[x0 + i][y0 + j]	ae(v)
}	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(depth_intra_mode[x0 + i][y0 + j] == HEVC_INTRA_PRED) {	
if(prev_intra_luma_pred_flag[x0 + i][y0 + j])	
mpm_idx[x0 + i][y0 + j]	ae(v)
} else {	
rem_intra_luma_pred_mode[x0 + i][y0 + j]	ae(v)
if(rem_intra_luma_pred_mode[x0 + i][y0 + j] == 31 && (log2CbSize + (PartMode == PART_NxN) ? -1 : 0) <= 5 && (log2CbSize + (PartMode == PART_NxN) ? -1 : 0) >= 2)	
ccm_info(x0 + i , y0 + j)	
}	
}	
}	
}	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else {	
...	
}	
}	
}	

[0268]

[0269] 제 3 대안에서는, CCM 관련 선택스 엘리먼트들이 심도 모델링 테이블로부터 제거되고, 그 대신에, 새로운 체인 코딩 선택스 테이블 "ccm_info" 이 도입된다.

[0270] H.xxxx 심도 모델링 테이블 선택스

depth_modeling_table(x0 , y0) {	디스크립터
depth_intra_mode[x0][y0]	ae(v)
dmmFlag[x0][y0] = (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEX depth_intra_mode[x0][y0] == MODE_DMM_CPREDTEX)	
sdcFlag[x0][y0] = (depth_intra_mode[x0][y0] == SDCM_DMM_1 depth_intra_mode[x0][y0] == SDCM_DC depth_intra_mode[x0][y0] == SDCM_Planar)	
if(depth_intra_mode[x0][y0] == MODE_DMM_WFULL)	

[0271]

wedge_full_tab_idx [x0][y0]	ac(v)
else if(depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEx)	
wedge_predtex_tab_idx [x0][y0]	ac(v)
else if (depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR) {	
dmm_delta_end_flag [x0][y0]	ac(v)
if (dmm_delta_end_flag[x0][y0]) {	
dmm_delta_end_abs_minus1 [x0][y0]	ac(v)
dmm_delta_end_sign_flag [x0][y0]	ac(v)
}	
} else if (depth_intra_mode[x0][y0] == CCM) {	
edge_start_left_flag[x0][y0]	ae(+)
edge_start_position[x0][y0]	ae(+)
num_edge_codes_minus1[x0][y0]	ae(+)
for(k = 0; k <= num_edge_codes_minus1; k++)	
edge_code[k]	ae(+)
}	
if(dmmFlag[x0][y0]) {	
dmm_delta_flag [x0][y0]	ac(v)
if (dmm_delta_flag [x0][y0]) {	
for(i = 0; i < 2; i++) {	
dmm_dc_abs [x0][y0][i]	ac(v)
if (dmm_dc_abs [x0][y0][i] != 0)	
dmm_dc_sign_flag [x0][y0][i]	ac(v)
}	
}	
} else if (depth_intra_mode[x0][y0] == CCM) {	
edge_de_flag[x0][y0]	ae(+)
if(edge_de_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
edge_de_abs[x0][y0][i]	ae(+)
if(edge_de_abs[x0][y0][i] != 0)	
edge_de_sign_flag[x0][y0][i]	ae(+)
}	
}	
}	
else if(sdcFlag[x0][y0]) {	
sdcNumSegments = depth_intra_mode[x0][y0] == SDCM_DMM_1 ? 2 : 1	
for (i = 0; i < sdcNumSegments ; i++) {	
sdc_residual_flag [x0][y0][i]	ac(v)
if(sdc_residual_flag [x0][y0][i]) {	
sdc_residual_sign_flag [x0][y0][i]	ac(v)
sdc_residual_abs_minus1 [x0][y0][i]	ac(v)
}	
}	
}	
}	

[0272]

[0273] H.xxxx 체인 코딩 모드 선택스

ccm_info (x0 , y0) {	디스크립터
edge_intra_flag[x0][y0]	ae(v)
if(edge_intra_flag[x0][y0]) {	
edge_start_left_flag[x0][y0]	ae(v)
edge_start_position[x0][y0]	ae(v)
num_edge_codes_minus1[x0][y0]	ae(v)
for(k = 0; k <= num_edge_codes_minus1; k++)	
edge_code[k]	ae(v)
edge_dc_flag[x0][y0]	ae(v)
if(edge_dc_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
edge_dc_abs[x0][y0][i]	ae(v)
if(edge_dc_abs[x0][y0][i] != 0)	
edge_dc_sign_flag[x0][y0][i]	ae(v)
}	
}	
}	
}	

[0274]

[0275] 위에서 제시된 수정된 심도 모델링 테이블에 대한 시맨틱들은 다음과 같이 정의된다. depth_intra_mode[x0][y0] 는 이하의 표 4 에서 도시된 값들 중의 하나일 수도 있다. 표 4 는 기존의 3D-HEVC 표준으로부터 변화하지 않은 이용가능한 심도 인트라 모드들을 나타낸다.

[0276]

표 4 : depth_intra_mode 의 해독

depth_intra_mode	DepthIntraMode	빈 스트림				
		(cLog2CbSize == 3 && partition size == SIZE_2Nx2N) (cLog2CbSize > 3 && cLog2CbSize < 6)		cLog2CbSize = 3 && partition size == SIZE_NxN		cLog2CbSize == 6
		프리픽스	서브픽스	프리픽스	서브픽스	
0	SDCM Planar	0	0	-		0
1	HEVC 인트라 (CCM 포함) (HEVC INTRA_PRED)	0	1 0	0		1 0
2	SDCM_DMM_1	0	1 1	-		-
3	DMM_1 (MODE_DMM_WFULL)	1	0 0	1	0	-
4	DMM_4 (MODE_DMM_CPREDTEX)	1	0 1	-		-
5	DMM_3 (MODE_DMM_WFULL)	1	1 0	1	1	-
6	SDCM_DC	1	1 1 0			1 1
7	DMM_2 (MODE_DMM_WPREDDIR)	1	1 1 1			

[0277]

[0278] 하나의 대안에서, 0, 1 및 2 (즉, SDCM Planar, CCM 을 포함하는 HEVC 인트라, 및 SDCM_DMM_1 을 표시함) 과 동일한 선택스 엘리먼트 depth_intra_mode 에 대한 빈 스트림은 각각 "000", "001" 및 "01" 로 설정될 수도 있다. 또 다른 대안에서, 0, 1 및 2 와 동일한 선택스 엘리먼트 depth_intra_mode 에 대한 빈 스트림은 각각 "000", "01" 및 "001" 로 설정될 수도 있다. 추가의 대안에서는, 제 1 의 3 개의 심도 인트라 모드들의 다른 순서들이 이용될 수도 있다. 일반적으로, DepthIntraMode 를 제시하기 위한 다른 빈 스트림들이 적용될 수도 있다.

[0279]

예를 들어, 3D-HEVC 표준에서의 심도 인트라 모드들의 이용가능한 수는 1 만큼 감소될 수도 있으며, 즉, DMM_2 가 제거될 수도 있다. 위에서 제시된 심도 모델링 테이블로부터의 시맨틱들 및 선택스 테이블들에 있어서의 변화들은 표 5 에서의 지움표시와, 이하에서 제시된 수정된 심도 모델링 선택스 테이블을 이용하여 표시된다.

표 5 : depth_intra_mode 의 해독

depth_intra_mode	DepthIntraMode	빈 스트링				
		(cLog2CbSize == 3 && partition size == SIZE_2Nx2N) (cLog2CbSize > 3 && cLog2CbSize < 6)		cLog2CbSize = 3 && partition size == SIZE_NxN		cLog2CbSize == 6
		프리픽스	서브픽스	프리픽스	서브픽스	
0	SDCM_Planar	0	0	-		0
1	HEVC 인트라 (CCM 포함)	0	1 0	0		1 0
2	SDCM_DMM_1	0	1 1	-		-
3	DMM_1(MODE_DMM_WFULL)	1	0 0	1	0	-
4	DMM_4(MODE_DMM_CPREDTEX)	1	0 1	-		-
5	DMM_3(MODE_DMM_WFULL)	1	1 0	1	1	-
6	SDCM_DC	1	1 1 0			1 1
7	DMM_2(MODE_DMM_WPREDDIR)	1	1 1 1			

H.xxxx 심도 모델링 테이블 선택스

depth_modeling_table(x0 , y0) {	디스크립터
depth_intra_mode[x0][y0]	ae(v)
dmmFlag[x0][y0] = (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEX depth_intra_mode[x0][y0] == MODE_DMM_CPREDTEX)	
sdcmFlag[x0][y0] = (depth_intra_mode[x0][y0] == SDCM_DMM_1 depth_intra_mode[x0][y0] == SDCM_DC depth_intra_mode[x0][y0] == SDCM_Planar)	
if (depth_intra_mode[x0][y0] == MODE_DMM_WFULL)	
wedge_full_tab_idx[x0][y0]	ae(v)
else if (depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEX)	
wedge_predtex_tab_idx[x0][y0]	ae(v)
else if (depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR) {	
dmm_delta_end_flag[x0][y0]	ae(+)
if (dmm_delta_end_flag[x0][y0]) {	
dmm_delta_end_abs_minus1[x0][y0]	ae(+)
dmm_delta_end_sign_flag[x0][y0]	ae(+)
}	
if(dmmFlag[x0][y0]) {	
...	
}	

[0284] 또 다른 예에서, SDC 모드에서 서브-모드 SDCM_DC 는 심도 인트라 코딩에서 고려되지 않을 수도 있다. 그 경우에, 6 과 동일한 depth_intra_mode 는 CCM 에 의해 대체될 수도 있다.

[0285] 심도 모델링 테이블을 시그널링하는 제 1 예의 기법의 제 4 대안은 위에서 설명된 제 3 대안과 유사하다. 그러나, 제 4 대안에서는, DMM 의 서브-모드들 DMM_1 및 DMM_3 은 HEVC 인트라 코딩 모드의 일부로서 취급된다. 위에서 설명된 제 3 대안을 통한 CU 및 심도 모델링 테이블의 선택스 및 시맨틱들에 대한 추가들은 회색으로 강조표시되고, 삭제들은 지움표시를 이용하여 표시된다.

[0286]

G.7.3.9.1 일반적인 코딩 유닛 선택스

coding_unit(x0, y0, log2CbSize , ctDepth) {	디스크립터
if(transquant_bypass_enable_flag) {	
cu_transquant_bypass_flag	ae(v)
}	
...	
if(pcm_flag) {	
...	
} else {	
pbOffset = (PartMode == PART_NxN) ? (nCbS / 2) : 0	
log2PbSize = log2CbSize - (PartMode == PART_NxN ? 1 : 0)	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(DepthModesFlag)	
depth_intra_mode [x0 + i][y0 + j]	ae(v)
if(depth_intra_mode[x0 + i][y0 + j] != HEVC_INTRA_PRED)	
prev_intra_luma_pred_flag [x0 + i][y0 + j]	ae(v)
}	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(depth_intra_mode[x0 + i][y0 + j] == HEVC_INTRA_PRED) {	
if(prev_intra_luma_pred_flag[x0 + i][y0 + j])	
mpm_idx [x0 + i][y0 + j]	ae(v)
else {	
rem_intra_luma_pred_mode [x0 + i][y0 + j]	ae(v)
if(rem_intra_luma_pred_mode[x0 + i][y0 + j] == 31 && (log2CbSize + (PartMode == PART_NxN) ? - 1 : 0) <= 5 && (log2CbSize + (PartMode == PART_NxN) ? - 1 : 0) >= 2)	
dmm_ccm_directional_intra_idc [x0 + i][y0 + j]	ae(v)
if(dmm_ccm_directional_intra_idc[x0 + i][y0 + j] == 3) {	
ccm_info (x0 + i , y0 + j)	
} else if(dmm_ccm_directional_intra_idc[x0 + i][y0 + j] != 0)	
wedge_predtex_tab_idx [x0 + i][y0 + j]	ae(v)
}	
if(dmm_ccm_directional_intra_idc[x0 + i][y0 + j] != 0 depth_intra_mode[x0 + i][y0 + j] == MODE_DMM_CPREDTEX)	
{	
for(k = 0; k < 2; k++) {	
edge_dc_abs [x0 + i][y0 + j][k]	ae(v)
if(edge_dc_abs[x0 + i][y0 + j][k] != 0)	
edge_dc_sign_flag [x0 + i][y0 + j][k]	ae(v)
}	
}	
}	
}	
intra_chroma_pred_mode [x0][y0]	ae(v)
}	
} else {	
if(PartMode == PART_2Nx2N)	
...	
}	

[0287]

[0288] H.xxxx 심도 모델링 테이블 선택스

depth_modeling_table(x0 , y0) {	디스크립터
depth_intra_mode[x0][y0]	ae(v)
dmmFlag[x0][y0] = (depth_intra_mode[x0][y0] == MODE_DMM_WFULL depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEx depth_intra_mode[x0][y0] == MODE_DMM_CPREDTEx)	
sdcmFlag[x0][y0] = (depth_intra_mode[x0][y0] == SDCM_DMM_1 depth_intra_mode[x0][y0] == SDCM_DC depth_intra_mode[x0][y0] == SDCM_Planar)	
if (depth_intra_mode[x0][y0] == MODE_DMM_WFULL)	
wedge_full_tab_idx[x0][y0]	ae(+)
else if(depth_intra_mode[x0][y0] == MODE_DMM_WPREDTEx)	
wedge_predtex_tab_idx[x0][y0]	ae(+)
else if(depth_intra_mode[x0][y0] == MODE_DMM_WPREDDIR) {	
dmm_delta_end_flag[x0][y0]	ae(+)
if (dmm_delta_end_flag[x0][y0]) {	
dmm_delta_end_abs_minus1[x0][y0]	ae(+)
dmm_delta_end_sign_flag[x0][y0]	ae(+)
}	
}	
if(dmmFlag[x0][y0]) {	
dmm_delta_flag[x0][y0]	ae(+)
if (dmm_delta_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
dmm_de_abs[x0][y0][i]	ae(+)
if (dmm_de_abs[x0][y0][i] != 0)	
dmm_de_sign_flag[x0][y0][i]	ae(+)
}	
}	
}	
else if(sdcmFlag[x0][y0]) {	
sdcmNumSegments = depth_intra_mode[x0][y0] == SDCM_DMM_1 ? 2 : 1	
for (i = 0; i < sdcmNumSegments ; i++) {	
sdcm_residual_flag[x0][y0][i]	ae(v)
if(sdcm_residual_flag [x0][y0][i]) {	
sdcm_residual_sign_flag[x0][y0][i]	ae(v)
sdcm_residual_abs_minus1[x0][y0][i]	ae(v)
}	
}	
}	
}	

[0289]

[0290] H.xxxx 체인 코딩 모드 선택스

ccm_info (x0 , y0) {	디스크립터
edge_intra_flag[x0][y0]	ae(+)
if(edge_intra_flag[x0][y0]) {	
edge_start_left_flag[x0][y0]	ae(v)
edge_start_position[x0][y0]	ae(v)
num_edge_codes_minus1[x0][y0]	ae(v)
for(k = 0; k <= num_edge_codes_minus1; k++)	
edge_code[k]	ae(v)
edge_de_flag[x0][y0]	ae(+)
if(edge_de_flag[x0][y0]) {	
for(i = 0; i < 2; i++) {	
edge_de_abs[x0][y0][i]	ae(+)
if(edge_de_abs[x0][y0][i] != 0)	
edge_de_sign_flag[x0][y0][i]	ae(+)
}	
}	
}	
}	

[0291]

[0292] CU 선택스 테이블 및 위에서 제시된 수정된 심도 모델링 테이블에 대한 시맨틱들은 다음과 같이 정의된다.

depth_intra_mode[x0][y0] 는 이하의 표 6 에서 도시된 값들 중의 하나일 수도 있다.

[0293] 표 6 : depth_intra_mode 의 해독

depth_intra_mode	DepthIntraMode	빈 스트림				
		(cLog2CbSize == 3 && partition size == SIZE_2Nx2N) (cLog2CbSize > 3 && cLog2CbSize < 6)		cLog2CbSize == 3 && partition size == SIZE_NxN		cLog2CbSize == 6
		프리픽스	서브픽스	프리픽스	서브픽스	
0	SDCM_Planar	0	0	-	-	0
1	HEVC 인트라 (CCM, DMM_1, DMM_3 을 포함함)	0	1 0	-	-	1 0
2	SDCM_DMM_1	0	1 1	-	-	-
3	DMM_4 (MODE_DMM_CPREDTEX)	1	0	-	-	-
4	SDCM_DC	1	1	-	-	1 1

[0294]

[0295] dmm_ccm_directional_intra_idc[x0 + i][y0 + i] 는 이하의 표 7 에서 도시된 값들 중의 하나일 수도 있다.

[0296] 표 7 : dmm_ccm_directional_intra_idc 의 해독

dmm_ccm_directional_intra_idc	DmmCcmDirectionalIntraIdc	빈 스트림			
		rem_intra_luma_pred_mode[x0 + i][y0 + j] == 31		rem_intra_luma_pred_mode[x0 + i][y0 + j] != 31	
0	HEVC_INTRA	0	0	0	
1	DMM_1	1	0	1	0
2	DMM_3	1	1 0	1	1
3	CCM	1	1 1	-	-

[0297]

[0298] 이 개시물의 기법들의 제 2 예에서는, DMM, HEVC 인트라 모드, 및 CCM 에 대한 선택스 테이블들 및 선택스 엘리먼트들이 간략화된다. 이 예에서는, 몇몇 루프들에서, 모든 PU 들의 dmm_flag, 모든 PU 들의 DMM 서브-모드들, 및 모든 PU 들의 다른 DMM 관련 정보를 별도로 시그널링하는 대신에, CU 선택스 테이블 또는 심도 모델링 테이블은 하나의 PU 의 모든 DMM 정보와 함께, 다음으로, 또 다른 PU 에 대한 정보 등등을 포함하도록 설계될 수도 있다. 그러므로, 현재의 CU 에서 모든 PU 들을 횡단하는 하나의 루프만이 요구된다.

[0299] 심도 인트라 코딩 모드들에 대한 선택스 테이블들 및 엘리먼트들을 간략화하는 제 2 예의 기법의 제 1 대안에서는, 관련된 시맨틱들이 변화되지 않은 채로 유지되는 반면, CU 선택스 테이블은 비디오 디코더 (30) 에서의 파싱 프로세스에 대해 이를 더욱 간단하게 하도록 수정된다. 3D-HEVC 표준에 대한 추가들은 이하에서 회색으로 강조표시되고, 3D-HEVC 표준으로부터의 삭제들은 지움표시를 이용하여 표시된다.

[0300] G.7.3.9.1 일반적인 코딩 유닛 선택스

coding_unit(x0, y0, log2CbSize, Depth) {	디스크립터
if(transquant_bypass_enable_flag) {	
cu_transquant_bypass_flag	ae(v)
}	
if(slice_type != 1 && !MotionIntrFlag[x0][y0])	
skip_flag[x0][y0]	ae(v)
if(skip_flag[x0][y0])	
prediction_unit(x0, y0, log2CbSize)	
else {	
if(!MotionIntrFlag[x0][y0]) {	
nCbS = (1 << log2CbSize)	
if(slice_type != 1)	
pred_mode_flag	ae(v)
if(PredMode == MODE_INTRA && DepthFlag)	
sdc_flag[x0][y0]	ae(v)
if(sdc_flag[x0][y0]) {	

[0301]

sdc_pred_mode	ac(v)
if(sdc_pred_mode == 1)	
wedge_full_tab_idx[x0][y0]	ac(v)
} else {	
if((PredMode[x0][y0] != MODE_INTRA log2CbSize == Log2MinCbSize) && !predPartModeFlag)	
part_mode	ac(v)
if(PredMode[x0][y0] == MODE_INTRA) {	
if(PartMode == PART_2Nx2N && pcm_enabled_flag && log2CbSize >= Log2MinIPCMCUSize && log2CbSize <= Log2MaxIPCMCUSize)	
pcm_flag	ac(v)
if(pcm_flag) {	
num_subsequent_pcm	tu(3)
NumPCMBlock = num_subsequent_pcm + 1	
while(!byte_aligned())	
pcm_alignment_zero_bit	f(1)
pcm_sample(x0, y0, log2CbSize)	
} else {	
pbOffset = (PartMode == PART_NxN) ? (nCbS / 2) : 0	
if(vps_dmm_flag[nuh_layer_id] && log2CbSize <= Log2MaxDmmCbSize)	
{	
log2PbSize = log2CbSize - (PartMode == PART_NxN ? 1 : 0)	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
dmm_flag[x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if(dmm_flag[x0 + i][y0 + j])	
dmm_mode[x0 + i][y0 + j]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if(dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULL dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULLDELTA))	
}	
wedge_full_tab_idx[x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	
if(dmm_flag[x0 + i][y0 + j] && (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDTEx dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDTExDELTA))	
wedge_full_tab_idx[x0 + i][y0 + i]	ae(v)
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset)	

[0302]

<pre> if(dmm_flag[x0+i][y0+j] && (dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIR dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIRDELTA)) </pre>	
<pre> dmm_delta_end_flag[x0+i][y0+i] </pre>	ae(v)
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) if (dmm_delta_end_flag[x0+i][y0+i]) dmm_delta_end_abs_minus1[x0+i][y0+i] </pre>	ae(v)
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) if (dmm_delta_end_flag[x0+i][y0+i]) dmm_delta_end_sign_flag[x0][y0] </pre>	ae(v)
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) DmmDeltaFlag[x0+i][y0+i] = (dmm_flag[x0+i][y0+j] && (dmm_mode[x0+i][y0+i] == MODE_DMM_WFULLEDELTA dmm_mode[x0+i][y0+i] == MODE_DMM_WPREDDIRDELTA dmm_mode[x0+i][y0+i] == MODE_DMM_WPRETEXDELTA dmm_mode[x0+i][y0+i] == MODE_DMM_CPREDTEXDELTA)) </pre>	
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) if (DmmDeltaFlag[x0+i][y0+i]) dmm_de_1_abs[x0+i][y0+i] </pre>	ae(v)
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) if (DmmDeltaFlag[x0+i][y0+j] && dmm_de_1_abs[x0+i][y0+i] != 0) dmm_de_1_sign_flag[x0+i][y0+i] </pre>	ae(v)
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) if (DmmDeltaFlag[x0+i][y0+j]) dmm_de_2_abs[x0+i][y0+i] </pre>	ae(v)
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) if (DmmDeltaFlag[x0+i][y0+j] && dmm_de_2_abs[x0+i][y0+i] != 0) dmm_de_2_sign_flag[x0+i][y0+i] </pre>	ae(v)
<pre> } else { for(j = 0; j <= pbOffset; j = j + pbOffset) for(i = 0; i <= pbOffset; i = i + pbOffset) { if (vps_dmm_flag[nuh_layer_id] && log2CbSize <= Log2MaxDmmCbSize) </pre>	
<pre> dmm_flag[x0+i][y0+j] </pre>	ae(v)
<pre> if (!dmm_flag[x0+i][y0+j]) prev_intra_luma_pred_flag[x0+i][y0+j] </pre>	ae(v)
<pre> else { dmm_mode[x0+i][y0+j] </pre>	ae(v)

[0303]

<pre> DmmDeltaFlag[x0+i][y0+j]={ dmm_mode[x0+i][y0+j] == MODE_DMM_WFULLDELTA dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIRDELTA dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDTEXDELTA dmm_mode[x0+i][y0+j] == MODE_DMM_CPREDTEXDELTA } </pre>	
<pre> if (dmm_mode[x0+i][y0+j] == MODE_DMM_WFULL dmm_mode[x0+i][y0+j] == MODE_DMM_WFULLDELTA) </pre>	
<pre> wedge_full_tab_idx[x0+i][y0+j] </pre>	ae(v)
<pre> else if (dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIR dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDTEXDELTA) </pre>	
<pre> wedge_predtex_tab_idx[x0+i][y0+j] </pre>	ae(v)
<pre> else if (dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIR dmm_mode[x0+i][y0+j] == MODE_DMM_WPREDDIRDELTA) { </pre>	
<pre> dmm_delta_end_flag[x0+i][y0+j] </pre>	ae(v)
<pre> if(dmm_delta_end_flag[x0+i][y0+j]) { </pre>	
<pre> dmm_delta_end_abs_minus1[x0+i][y0+j] </pre>	ae(v)
<pre> dmm_delta_end_sign_flag[x0+i][y0+j] </pre>	ae(v)
<pre> } } </pre>	
<pre> if (DmmDeltaFlag[x0+i][y0+j]) { </pre>	
<pre> dmm_dc_1_abs[x0+i][y0+j] </pre>	ae(v)
<pre> dmm_dc_2_abs[x0+i][y0+j] </pre>	ae(v)
<pre> if (dmm_dc_1_abs[x0+i][y0+j]) </pre>	
<pre> dmm_dc_1_sign_flag[x0+i][y0+j] </pre>	ae(v)
<pre> if (dmm_dc_2_abs[x0+i][y0+j]) </pre>	
<pre> dmm_dc_2_sign_flag[x0+i][y0+j] </pre>	ae(v)
<pre> } } } </pre>	
<pre> for(j = 0; j <= pbOffset; j = j + pbOffset) </pre>	
<pre> for(i = 0; i <= pbOffset; i = i + pbOffset) { </pre>	
<pre> if(!dmm_flag[x0+i][y0+j]) { </pre>	
<pre> if(prev_intra_luma_pred_flag[x0+i][y0+j]) </pre>	
<pre> mpm_idx[x0+i][y0+j] </pre>	ae(v)
<pre> else { </pre>	
<pre> rem_intra_luma_pred_mode[x0+i][y0+j] </pre>	ae(v)
<pre> if(rem_intra_luma_pred_mode[x0+i][y0+j] == 31 && (log2CbSize + (PartMode == PART_NxN) ? -1 : 0) <= 5 && (log2CbSize + (PartMode == PART_NsN) ? -1 : 0) >= 2) { </pre>	
<pre> edge_intra_flag[x0+i][y0+j] </pre>	ae(v)
<pre> if(edge_intra_flag[x0+i][y0+j]) { </pre>	
<pre> edge_start_left_flag[x0+i][y0+j] </pre>	ae(v)
<pre> edge_start_position[x0+i][y0+j] </pre>	ae(v)
<pre> num_edge_codes_minus1[x0+i][y0+j] </pre>	ae(v)
<pre> for(k=0; k <= num_edge_codes_minus1; k++) </pre>	
<pre> edge_code[k] </pre>	ae(v)

[0304]

dmm_mode[x0 + i][y0 + j]	ac(v)
DmmDeltaFlag[x0 + i][y0 + i] = (dmm_mode[x0 + i][y0 + i] == MODE_DMM_WFULLDELTA dmm_mode[x0 + i][y0 + i] == MODE_DMM_WPREDDIRDELTA dmm_mode[x0 + i][y0 + i] == MODE_DMM_WPRETEXDELTA dmm_mode[x0 + i][y0 + i] == MODE_DMM_CPREDTEXDELTA)	
if (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULL dmm_mode[x0 + i][y0 + j] == MODE_DMM_WFULLDELTA)	
wedge_full_tab_idx[x0 + i][y0 + i]	ac(v)
else if (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPRETEX dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPRETEXDELTA)	
wedge_predtex_tab_idx[x0][y0]	ac(v)
else if (dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDDIR dmm_mode[x0 + i][y0 + j] == MODE_DMM_WPREDDIRDELTA) {	
dmm_delta_end_flag[x0 + i][y0 + i]	ac(v)
if(dmm_delta_end_flag[x0 + i][y0 + i]) {	
dmm_delta_end_abs_minus1[x0 + i][y0 + i]	ac(v)
dmm_delta_end_sign_flag[x0][y0]	ac(v)
}	
}	
if (DmmDeltaFlag[x0 + i][y0 + j]) {	
dmm_de_1_abs[x0 + i][y0 + j]	ac(v)
dmm_de_2_abs[x0 + i][y0 + j]	ac(v)
if (dmm_de_1_abs[x0 + i][y0 + j])	
dmm_de_1_sign_flag[x0 + i][y0 + j]	ac(v)
if (dmm_de_2_abs[x0 + i][y0 + j])	
dmm_de_2_sign_flag[x0 + i][y0 + j]	ac(v)
}	
}	
for(j = 0; j <= pbOffset; j = j + pbOffset)	
for(i = 0; i <= pbOffset; i = i + pbOffset) {	
if(!dmm_flag[x0 + i][y0 + j]) {	
if(prev_intra_luma_pred_flag[x0 + i][y0 + j])	
mpm_idx[x0 + i][y0 + j]	ac(v)
else {	
rem_intra_luma_pred_mode[x0 + i][y0 + j]	ac(v)
if(rem_intra_luma_pred_mode[x0 + i][y0 + j] == 31 && (log2CbSize + (PartMode == PART_NxN) ? - 1 : 0) <= 5 && (log2CbSize + (PartMode == PART_NxN) ? - 1 : 0) >= 2) {	
edge_intra_flag[x0 + i][y0 + j]	ac(v)
if(edge_intra_flag[x0 + i][y0 + j]) {	
edge_start_left_flag[x0 + i][y0 + j]	ac(v)
edge_start_position[x0 + i][y0 + j]	ac(v)
num_edge_codes_minus1[x0 + i][y0 + j]	ac(v)
for(k = 0; k <= num_edge_codes_minus1; k++)	

[0309]

edge_code[k]	ac(v)
edge_dc_flag[x0 + i][y0 + j]	ac(v)
if(edge_dc_flag[x0 + i][y0 + j]) {	
edge_dc_1_abs[x0 + i][y0 + j]	ae(+)
if(edge_dc_1_abs[x0 + i][y0 + j] != 0)	
edge_dc_1_sign_flag[x0 + i][y0 + j]	ae(+)
edge_dc_2_abs[x0 + i][y0 + j]	ae(+)
if(edge_dc_2_abs[x0 + i][y0 + j] != 0)	
edge_dc_2_sign_flag[x0 + i][y0 + j]	ae(+)
}	
}	
}	
if(edge_dc_flag[x0 + i][y0 + j] DmmDeltaFlag[x0 + i][y0 + i]) {	
dc_1_abs[x0 + i][y0 + j]	ae(v)
if(dc_1_abs[x0 + i][y0 + j] != 0)	
dc_1_sign_flag[x0 + i][y0 + j]	ae(v)
dc_2_abs[x0 + i][y0 + j]	ae(v)
if(dc_2_abs[x0 + i][y0 + j] != 0)	
dc_2_sign_flag[x0 + i][y0 + j]	ae(v)
}	
}	
}	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else {	
...	
}	

[0310]

[0311] 위에서 제시된 CU 선택스 테이블에 대한 시맨틱들은 다음과 같이 정의된다.

$dmm_dc_0_abs[x0][y0]$, $dmm_dc_0_sign_flag[x0][y0]$,
 $dmm_dc_1_abs[x0][y0]$, 및 $dmm_dc_1_sign_flag[x0][y0]$ 는 다음과
 같이 QuantDCOffsetP0[x0][y0] 및 QuantDCOffsetP1[x0][y0] 값들을
 유도하기 위하여 이용된다:

$$QuantDCOffsetP0[x0][y0] = (1 - 2 * dmm_dc_0_sign_flag[x0][y0]) * dmm_dc_0_abs[x0][y0] \tag{G-19}$$

$$QuantDCOffsetP1[x0][y0] = (1 - 2 * dmm_dc_1_sign_flag[x0][y0]) * dmm_dc_1_abs[x0][y0] \tag{G-20}$$

~~edge_dc_0_abs[x0][y0], edge_dc_0_sign_flag[x0][y0],~~
~~edge_dc_1_abs[x0][y0], 및 edge_dc_1_sign_flag[x0][y0]~~ 는
 다음과 같이 QuantDCOffsetP0[x0][y0] 및 QuantDCOffsetP1[x0][y0]
 값들을 유도하기 위하여 이용된다.

~~$$QuantDCOffsetP0[x0][y0] = (1 - 2 * edge_dc_0_sign_flag[x0][y0]) * edge_dc_1_abs[x0][y0] \tag{G-21}$$~~

~~$$QuantDCOffsetP1[x0][y0] = (1 - 2 * edge_dc_0_sign_flag[x0][y0]) * edge_dc_1_abs[x0][y0] \tag{G-22}$$~~

[0312]

[0313] 이 개시물의 기법들의 제 3 예에서, 비디오 인코더 (20) 는 심도 코딩 모드들 중의 2 개 이상에 걸쳐 통합되는 시그널링 메커니즘들을 이용하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 시그널링한다. 이러

한 방법으로, 비디오 디코더 (30) 는 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 결정하기 위하여 선택스 엘리먼트들의 동일한 세트를 디코딩할 수도 있다. 선택스 엘리먼트들의 통합된 세트는 잔차 값들이 심도 코딩 모드들 중의 임의의 것에 대해 코딩되는지 여부를 표시하는 선택스 엘리먼트를 포함할 수도 있다. 선택스 엘리먼트들의 통합된 세트는 또한, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 값들의 절대값 및 부호를 표시하는 선택스 엘리먼트들을 포함할 수도 있다. 심도 코딩 모드들 중의 2 개 이상에 대한 잔차 정보를 표시하기 위하여 선택스 엘리먼트들의 동일한 세트를 공유함으로써, 선택스 엘리먼트들에 대한 컨텍스트 모델 선택 및 이진화 방법들이 또한 통합될 수도 있다.

[0314]

3D-HEVC 표준에 대한 추가들은 이하에서 회색으로 강조표시되고, 3D-HEVC 표준으로부터의 삭제들은 지움표시를 이용하여 표시된다. 3D-HEVC 의 작업 초안 (문서 번호: JCT3V-E1001) 은 다음으로부터 입수가능하다: http://phenix.int-evry.fr/jct3v/doc_end_user/documents/5_Vienna/wg11/JCT3V-E1001-v3.zip. 3D-HEVC 의 소프트웨어 설명 (문서 번호: JCT3V-E1005) 은 다음으로부터 입수가능하다: http://phenix.int-evry.fr/jct3v/doc_end_user/documents/5_Vienna/wg11/JCT3V-E1005-v1.zip.

[0318] H.7.4.9.5.1 심도 모드 파라미터 시맨틱들

1 과 동일한 $depth_dc_flag[x0][y0]$ 는 $depth_dc_abs[x0][y0][i]$ 및 $depth_dc_sign_flag[x0][y0][i]$ 가 존재하는 것을 특징한다. 0 과 동일한 $depth_dc_flag[x0][y0]$ 는 $depth_dc_abs[x0][y0][i]$ 및 $depth_dc_sign_flag[x0][y0][i]$ 가 존재하지 않으며 0 으로서 유도되는 것을 특징한다. $depth_dc_abs[x0][y0][i]$ 는 다음과 같이 $depth_dc_abs[x0][y0][i]$ 를 유도하기 위하여 이용된다.

$$depth_dc_abs[x0][y0][i] += (dcNumSeg == 1 ? 1 : 0)$$

$depth_dc_sign_flag[x0][y0][i]$ 는 다음과 같이 $DcOffset[x0][y0][i]$ 를 유도하기 위하여 이용된다.

$$DcOffset[x0][y0][i] =$$

$$(1 - 2 * depth_dc_sign_flag[x0][y0][i]) * depth_dc_abs[x0][y0][i] \quad (H-24)$$

[0319]

1 과 동일한 $dmm_dc_flag[x0][y0]$ 는 $dmm_dc_abs[x0][y0][i]$ 및 $dmm_dc_sign_flag[x0][y0][i]$ 가 존재하는 것을 특징한다. 0 과 동일한 $dmm_dc_flag[x0][y0]$ 는 $dmm_dc_abs[x0][y0][i]$ 및 $dmm_dc_sign_flag[x0][y0][i]$ 가 존재하지 않는 것을 특징한다. $dmm_dc_abs[x0][y0][i]$, $dmm_dc_sign_flag[x0][y0][i]$ 는 다음과 같이 $DeOffset[x0][y0][i]$ 를 유도하기 위하여 이용된다.

$$DeOffset[x0][y0][i] = (1 - 2 * dmm_dc_sign_flag[x0][y0][i]) * dmm_dc_abs[x0][y0][i] \quad (H-24)$$

1 과 동일한 $edge_dc_flag[x0][y0]$ 는 $edge_dc_abs[x0][y0][i]$ 및 $edge_dc_sign_flag[x0][y0][i]$ 가 존재하는 것을 특징한다. 0 과 동일한 $edge_dc_flag[x0][y0]$ 는 $edge_dc_abs[x0][y0][i]$ 및 $edge_dc_sign_flag[x0][y0][i]$ 가 존재하지 않는 것을 특징한다. $edge_dc_abs[x0][y0][i]$, $edge_dc_sign_flag[x0][y0][i]$ 는 다음과 같이 $DeOffset[x0][y0][i]$ 를 유도하기 위하여 이용된다.

$$(1 - 2 * edge_dc_sign_flag[x0][y0][i]) * edge_dc_abs[x0][y0][i] \quad (H-25)$$

0 과 동일한 $sdc_residual_flag[x0][y0][i]$ 는 잔차가 세그먼트 i 에 대하여 제로인 것을 특징한다. 1 과 동일한 $sdc_residual_flag$ 는 잔차가 비-제로이며 $sdc_residual_sign_flag$ 및 $sdc_residual_abs[i]$ 선택스 엘리먼트들이 세그먼트 i 에 대하여 존재하는 것을 특징한다. $sdc_residual_abs_minus1[x0][y0][i]$ 및 $sdc_residual_sign_flag[x0][y0][i]$ 는 다음과 같이 세그먼트 i 에 대하여 $SdcResidual[x0][y0][i]$ 를 유도하기 위하여 이용된다.

$$SdcResidual[x0][y0][i] = (1 - 2 * sdc_residual_sign_flag[x0][y0][i]) * (sdc_residual_mag_minus1[x0][y0][i] + 1) \quad (H-26)$$

[0320]

[0321]

H.8.4.4.2.11 심도 파티션 값 유도 및 배정 프로세스

[0322]

이 프로세스에 대한 입력들은 다음과 같다.

[0323]

- $x = -1, y = -1..nTbS * 2 - 1$, 그리고 $x = 0..nTbS * 2 - 1, y = -1$ 인 이웃하는 샘플들 $p[x][y]$

[0324]

- 파티션 0 및 파티션 1 에서의 예측 블록의 파티셔닝을 특징하는, $x, y = 0..nTbS - 1$ 인 2 진 어레이 $partitionPattern[x][y]$.

[0325]

- 변환 블록 사이즈를 특징하는 변수 $nTbS$

- [0326] - DC 오프셋 값들이 이용가능한지 여부를 특징하는 플래그 dcOffsetAvailFlag
 - ~~현재의 인트라 예측 모드가 INTRA_CHAIN 과 동일한지 여부를~~
 특징하는, 플래그 ~~intraChainFlag~~
- [0327]
- [0328] - 블록 파티션들에 대한 DC 오프셋들을 특징하는 변수들 dcOffsetP0 및 dcOffsetP1
- [0329] 이 프로세스의 출력은 다음과 같다.
- [0330] - $x, y = 0..nTbS - 1$ 인 예측된 샘플들 $predSamples[x][y]$
- [0331] 변수들 $vertEdgeFlag$ 및 $horEdgeFlag$ 는 다음에서 특징된 바와 같이 유도된다.
- [0332] $vertEdgeFlag = (partitionPattern[0][0] !=$
- [0333] $partitionPattern[nTbS - 1][0]) ? 1 : 0$ (H-45)
- [0334] $horEdgeFlag = (partitionPattern[0][0] !=$
- [0335] $partitionPattern[0][nTbS - 1]) ? 1 : 0$ (H-46)
- [0336] 변수들 $dcVal0$ 및 $dcVal1$ 은 다음에서 특징된 바와 같이 유도된다.
- [0337] - $vertEdgeFlag$ 가 $horEdgeFlag$ 와 동일할 경우, 다음이 적용된다:
- [0338] $dcValBR = horEdgeFlag ? ((p[-1][nTbS - 1] + p[nTbS - 1][-1]) >> 1) : (1$
 $<< (BitDepth_y - 1))$ (H-47)
- [0339] $dcValLT = (p[-1][0] + p[0][-1]) >> 1$ (H-48)
- [0340] - 이와 다를 경우 ($horEdgeFlag$ 가 $vertEdgeFlag$ 와 동일하지 않음), 다음이 적용된다:
- [0341] $dcValBR = horEdgeFlag ? p[-1][nTbS - 1] : p[nTbS - 1][-1]$ (H-49)
- [0342] $dcValLT = horEdgeFlag ? p[(nTbS - 1) >> 1][-1] :$
- [0343] $p[-1][(nTbS - 1) >> 1]$ (H-50)
- ~~변수 dcOffsetScale 는 다음과 같이 유도된다:~~
- ~~$dcOffsetScale = intraChainFlag ? Clip3(1, (1 << BitDepth_x) - 1, Round(2^{(QP_x$
 $40) - 2})) : 1$ (H-51)~~
- [0344]
- [0345] 예측된 샘플 값들 $predSamples[x][y]$ 은 다음에서 특징된 바와 같이 유도된다.
- [0346] - 0 내지 ($nTbS - 1$) 까지의 범위에서의 x 에 대하여, 다음이 적용된다:
- [0347] - 0 내지 ($nTbS - 1$) 까지의 범위에서의 y 에 대하여, 다음이 적용된다:
- [0348] - 변수들 $predDcVal$ 및 $dcOffset$ 은 다음에서 특징된 바와 같이 유도된다:
- $predDcVal = (partitionPattern[x][y] == partitionPattern[0][0]) ?$
- $dcValLT : dcValBR$ (H-52)
- $dcOffset = dcOffsetAvailFlag ? (partitionPattern[x][y] == 0 ?$
- $dcOffsetP0 : dcOffsetP1) : 0$ (H-53)
- ~~$dcOffset = dcOffset * dcOffsetScale$ (H-54)~~
- [0349]

- ~~intraChainFlag~~ 가 1 과 동일하거나 ~~dlt_flag[nuh_layer_id]~~ 가 0 과 동일할 경우, 다음이 적용된다:

$$\text{predSamples}[x][y] = \text{predDcVal} + \text{dcOffset} \quad (\text{H-55})$$

- 이와 다를 경우 (~~intraChainFlag~~ 가 0 과 동일하고 ~~dlt_flag[nuh_layer_id]~~ 가 1 과 동일함), 다음이 적용된다:

$$\text{predSamples}[x][y] = \text{Idx2DepthValue}[\text{DepthValue2Idx}[\text{predDcVal}] + \text{dcOffset}] \quad (\text{H-56})$$

[0350]

[0351] **H.8.4.4.3 심도 값 재구성 프로세스**

[0352] 이 프로세스에 대한 입력들은 다음과 같다.

[0353] - 현재의 픽처의 상부-좌측 루마 샘플과 관련하여 현재의 블록의 상부-좌측 루마 샘플을 특정하는 루마 로케이션 (x_{Tb}, y_{Tb})

[0354] - 변환 블록 사이즈를 특정하는 변수 n_{TbS}

[0355] - $x, y = 0..n_{TbS} - 1$ 인 예측된 샘플들 $\text{predSamples}[x][y]$

[0356] - 인트라 예측 모드 predModeIntra

[0357] 이 프로세스의 출력은 다음과 같다.

[0358] - $x, y = 0.. n_{TbS} - 1$ 인 재구성된 심도 값 샘플들 $\text{resSamples}[x][y]$

[0359] predModeIntra 에 따라서는, 2 진 세그먼트이션 패턴을 특정하는 $x, y = 0..n_{TbS} - 1$ 인 어레이 $\text{wedgePattern}[x][y]$ 가 다음과 같이 유도된다.

[0360] - predModeIntra 가 INTRA_DMM_WFULL 과 동일할 경우, 다음이 적용된다.

[0361] $\text{wedgePattern} =$

$$\text{WedgePatternTable}[\text{Log2}(n_{TbS})][\text{wedge_full_tab_idx}[x_{Tb}][y_{Tb}]]$$

[0363] - 이와 다를 경우 (predModeIntra 가 INTRA_DMM_WFULL 과 동일하지 않음), 다음이 적용된다:

[0364] - $x, y = 0..n_{TbS} - 1$ 에 대하여, $\text{wedgePattern}[x][y]$ 은 0 과 동일하게 설정된다

[0365] $\text{dlt_flag}[nuh_layer_id]$ 에 따라서는, 재구성된 심도 값 샘플들 $\text{resSamples}[x][y]$ 이 다음에서 특정된 바와 같이 유도된다.

[0366] - $\text{dlt_flag}[nuh_layer_id]$ 가 0 과 동일할 경우, 다음이 적용된다:

[0367] - $x, y = 0..n_{TbS} - 1$ 에 대하여, 재구성된 심도 값 샘플들 $\text{resSamples}[x][y]$ 은 다음에서 특정된 바와 같이 유도된다:

$$\text{resSamples}[x][y] = \text{predSamples}[x][y] +$$

$$\text{SdcResidualDcOffset}[x_{Tb}][y_{Tb}][\text{wedgePattern}[x][y]] \quad (\text{H-59})$$

[0370] - 이와 다를 경우 ($\text{dlt_flag}[nuh_layer_id]$ 는 1 과 동일함), 다음이 적용된다:

[0371] - 변수들 $\text{dcPred}[0]$ 및 $\text{dcPred}[1]$ 는 다음에서 특정된 바와 같이 유도된다:

[0372] - predModeIntra 가 INTRA_DC 와 동일할 경우, 다음이 적용된다:

[0373] $dcPred[0] = predSamples[nTbS - 1][nTbS - 1]$ (H-60)

[0374] - 이와 다르게, $predModeIntra$ 가 INTRA_PLANAR 와 동일할 경우에는, 다음이 적용된다:

[0375] $dcPred[0] = (predSamples[0][0] + predSamples[0][nTbS - 1] + predSamples[nTbS - 1][0] + predSamples[nTbS - 1][nTbS - 1] + 2) \gg 2$ (H-61)

[0376] - 이와 다른 경우 ($predModeIntra$ 가 INTRA_DMML_WFULL 과 동일함), 다음이 적용된다:

$dcPred[wedgePattern[0][0]] = predSamples[0][0]$ (H-62)

$dcPred[wedgePattern[nTbS - 1][0]] =$

$predSamples[nTbS - 1][0]$ (H-63)

$dcPred[wedgePattern[0][nTbS - 1]] =$

$predSamples[0][nTbS - 1]$ (H-64)

$dcPred[wedgePattern[nTbS - 1][nTbS - 1]] =$

[0377] $predSamples[nTbS - 1][nTbS - 1]$ (H-65)

[0378] - $x, y = 0..nTbS - 1$ 에 대하여, 재구성된 심도 값 샘플들 $resSamples[x][y]$ 은 다음에서 특정된 바와 같이 유도된다:

$dltIdxPred = DepthValue2Idx[dcPred[wedgePattern[x][y]]]$ (H-66)

$dltIdxResi =$

~~SdResidualDcOffset~~ $[xTb][yTb][wedgePattern[x][y]]$ (H-67)

$resSamples[x][y] = predSamples[x][y] +$

$Idx2DepthValue[dltIdxPred + dltIdxResi] -$

[0379] $dcPred[wedgePattern[x][y]]$ (H-68)

[0380] **H.9.3.2.2 컨텍스트 변수들에 대한 초기화 프로세스**

[0381] 하위조항 9.3.2.2 에서의 사양들은 다음의 수정들과 함께 적용된다.

[0382] - 하위조항들 7.3.8.1 내지 7.3.8.11 에서 특정된 프로세스에 대한 모든 참조들은 하위조항들 H.7.3.8.1 내지 H.7.3.8.11 에서 특정된 프로세스에 대한 참조들로 대체된다.

[0383] - 표 H-12 는 표 9-4 의 중반부에 첨부된다.

[0384] - 표 H-13 내지 표 H-24 는 하위조항의 중반부에 첨부된다.

[0385] 표 H-12 : 초기화 프로세스에서의 각각의 초기화 타입에 대한 ctxIdx 및 선택스 엘리먼트들의 연관

선택스 구조	선택스 엘리먼트	ctxTable	initType		
			0	1	2
coding_unit() depth_mode_parameters()	depth_intra_mode	표 H-21	0..7	8..15	16..23
	wedge_full_tab_idx	표 H-13	0	1	2
	wedge_predtex_tab_idx	표 H-14	0	1	2
	dmmdepth_dc_flag	표 H-22	0	1	2
	dmmdepth_dc_abs	표 H-15	0	1	2
	edge_code	표 H-16	0	1	2
	iv_res_pred_weight_idx	표 H-17		0..3	4..7
	ic_flag	표 H-18		0	1
	edge_dc_flag	표 H-23	0	1	2
	edge_dc_abs	표 H-24	0	1	2
	sdc_residual_flag	표 H-19	0	1	2
	sdc_residual_abs_minus1	프릭스: 표 H-20 서픽스: na	프릭스:0 서픽스: na	프릭스:1 서픽스: na	프릭스:2 서픽스: na
	inter_sdc_flag	표 H-25	0	1	2
	inter_sdc_resi_abs_minus1	표 H-26	0	1	2
	inter_sdc_resi_sign_flag	표 H-27	0	1	2

[0386]

[0387] 표 H-19: sdc_residual_flag ctxIdx 에 대한 initValue 의 값들

초기화 변수	sdc_residual_flag 의 ctxIdx		
	0	1	2
initValue	154	154	154

[0388]

[0389] 표 H-20: sdc_residual_abs_minus1 ctxIdx 에 대한 initValue 의 값들

초기화 변수	sdc_residual_abs_minus1 의 ctxIdx		
	0	1	2
initValue	프릭스: 155 서픽스: na	프릭스: 155 서픽스: na	프릭스: 155 서픽스: na

[0390]

[0391] 표 H-21 : depth_intra_mode ctxIdx 에 대한 initValue 의 값들

초기화 변수	depth_intra_mode 의 ctxIdx							
	0	1	2	3	4	5	6	7
initValue	0	0	64	168	168	124	154	0
	8	9	10	11	12	13	14	15
initValue	0	64	0	183	154	108	0	0
	16	17	18	19	20	21	22	23
initValue	64	0	154	154	168	109	0	0

[0392]

[0393] 표 H-22 : dmmdepth_dc_flag ctxIdx 에 대한 initValue 의 값들

초기화 변수	dmm_dc_flag 의 ctxIdx		
	0	1	2
initValue	0	0	64

[0394]

[0395] 표 H-23: ~~edge_dc_flag ctxIdx~~ 에 대한 ~~initValue~~ 의 값들

초기화 변수	edge_dc_flag 의 ctxIdx		
	0	1	2
initValue	154	154	154

[0396]

[0397] 표 H-24: ~~edge_dc_abs ctxIdx~~ 에 대한 ~~initValue~~ 의 값들

초기화 변수	edge_dc_abs 의 ctxIdx		
	0	1	2
initValue	154	154	154

[0398]

[0399] 표 H-25 : inter_sdc_flag ctxIdx 에 대한 initValue 의 값들

초기화 변수	inter_sdc_flag 의 ctxIdx		
	0	1	2
initValue	154	154	154

[0400]

[0401] 표 H-26 : inter_sdc_resi_abs_minus1 ctxIdx 에 대한 initValue 의 값들

초기화 변수	inter_sdc_resi_abs_minus1 의 ctxIdx		
	0	1	2
initValue	154	154	154

[0402]

[0403] 표 H-27 : inter_sdc_resi_sign_flag ctxIdx 에 대한 initValue 의 값들

초기화 변수	inter_sdc_resi_sign_flag 의 ctxIdx		
	0	1	2
initValue	154	154	154

[0404]

[0405] H.9.3.3 이진화 프로세스

[0406] H.9.3.3.1 일반

[0407] 하위조항 9.3.3.1 에서의 사양들은 다음의 수정들과 함께 적용된다.

[0408] - 표 H-28 은 표 9-32 의 종반부에 첨부된다.

[0409] 표 H-28 : 신택스 엘리먼트들 및 연관된 이진화들

신택스 구조	신택스 엘리먼트	이진화	
		프로세스	입력 파라미터들
coding_unit()	iv_res_pred_weight_idx	TR	cMax = 2, cRiceParam = 0
	ic_flag	FL	cMax = 1
	iv_res_pred_weight_idx	TR	cMax = 2, cRiceParam = 0
	inter_sdc_flag	FL	cMax = 1
	inter_sdc_resi_abs_minus1	UEG0	[특정되어야 할 Ed. (GT)]
	inter_sdc_resi_sign_flag	FL	cMax = 1
depth_mode_parameters()	depth_intra_mode_set_indication_flag	FL	cMax = 1
	depth_intra_mode	TR	cMax = depthIntraModeMaxLen, cRiceParam = 0
	wedge_full_tab_idx	FL	cMax = wedgeFullTabIdxBits[log2PbSize] (표 H-29 에서 정의됨)
	wedge_predtex_tab_idx	FL	cMax = wedgePredTexTabIdxBits[log2PbSize] (표 H-30 에서 정의됨)
	edge_start_left_flag [Ed. (GT) 누락]		
	edge_start_position [Ed. (GT) 누락]		
	num_edge_codes_minus1 [Ed. (GT) 누락]		
	edge_code	TR	cMax = 6, cRiceParam = 0
	depth_dc_flag	FL	cMax = 1
	depth_dc_abs	UEG0	[특정되어야 할 Ed. (GT)]
	depth_dc_sign_flag	FL	cMax = 1
	edge_dc_flag	FL	eMax = 1
	edge_dc_abs	FL	eMax = 1
	edge_dc_sign_flag	FL	eMax = 1
	sdc_residual_flag	FL	eMax = 1
	sdc_residual_sign_flag	FL	eMax = 1
	sdc_residual_abs_minus1	H.9.3.3.10	

[0410]

H.9.3.3.10 sdc_residual_abs_minus1 에 대한 이진화 프로세스

이 프로세스에 대한 입력은 신택스 엘리먼트 sdc_residual_abs_minus1 에 대한 요청이다.

이 프로세스의 출력은 신택스 엘리먼트의 이진화이다.

빈 스트링은 프리픽스 빈 스트링 (prefix bin string) 과, 존재할 경우, 서픽스 빈 스트링 (suffix bin string) 의 연결체이다.

변수 numDepthValues 는 다음과 같이 유도된다:

$$\begin{aligned} \text{numDepthValues} &= \text{dlt_flag}[\text{nuh_layer_id}]? \\ &\text{num_depth_values_in_dlt}[\text{nuh_layer_id}] : (1 \ll \text{BitDepth}_x) - 1 \end{aligned} \quad \text{(H.262)}$$

변수 cMaxPrefix 는 다음과 같이 유도된다:

$$\text{cMaxPrefix} = (\text{numDepthValues} * 3) \gg 2$$

[0411]

프리픽스 빈 스트링의 유도를 위하여, 다음이 적용된다:

= sdc_residual_abs_minus1 이 cMaxPrefix 보다 작을 경우, 프리픽스 빈 스트링은 binIdx 에 의해 인덱싱된 길이 sdc_residual_abs_minus1+ 1 의 비트 스트링이다. sdc_residual_abs_minus1 보다 더 작은 binIdx 에 대한 빈들은 1 과 동일하다. sdc_residual_abs_minus1 과 동일한 binIdx 를 갖는 빈은 0 과 동일하다.

= 이와 다른 경우, 프리픽스 빈 스트링은 모든 빈들이 1 과 동일한 길이 eMaxPrefix 의 비트 스트링이다.

sdc_residual_abs_minus1 이 cMaxPrefix 보다 더 클 때, 빈 스트링의 서픽스가 존재하고, 그것은 다음과 같이 유도된다:

= 서픽스 값 suffixVal 은 다음과 같이 유도된다:

$$\text{suffixVal} = \text{sdc_residual_abs_minus1} - \text{cMaxPrefix} \quad (\text{H-263})$$

= 빈 스트링의 서픽스는 하위조항에서 특정된 바와 같은 고정-길이 (Fixed-length; FL) 이진화 프로세스에 의해, 입력들로서 (numDepthValues = cMaxPrefix) 와 동일한 suffixVal 및 cMax 로 특정된다.

[0412]

[0413] 하나의 대안에서, 선택스 엘리먼트 depth_dc_flag 는, 이하에서 도시된 바와 같이, 파티션들의 수에 따라 빈만이 특정한 컨텍스트를 가지는 방법으로 이진화될 수도 있다.

[0414] **H.9.3.4.2 ctxTable, ctxIdx 및 bypassFlag 에 대한 유도 프로세스**

[0415] **H.9.3.4.2.1 일반**

[0416] 하위조항 9.3.4.2.1 에서의 사양들은 다음의 수정들과 함께 적용된다:

[0417] - 표 H-31 은 표 9-37 의 중반부에 첨부된다.

[0418] **표 H-31 : 컨텍스트 코딩된 빈들을 갖는 선택스 엘리먼트들에 대한 ctxInc 의 배정**

선택스 엘리먼트	binIdx					
	0	1	2	3	4	>=5
wedge_full_tab_idx	0	0	0	0	0	0
dmmdepth_dc_flag	0 dcNumSeg-1	na	na	na	na	na
dmmdepth_dc_abs	0	0	0	0	0	0
dmmdepth_dc_sign_flag	우회	0	0	0	0	0
edge_dc_flag	0	na	na	na	na	na
edge_dc_abs	0	0	0	0	0	0
edge_dc_sign_flag	우회	0	0	0	0	0
edge_code	0	0	0	0	0	0
res_pred_flag	0	na	na	na	na	na
ic_flag	0	na	na	na	na	na
sdc_residual_flag	우회	na	na	na	na	na
sdc_residual_abs_minus1	프리픽스: 0 서픽스: 우회	프리픽스: 0 서픽스: 우회	프리픽스: 0 서픽스: 우회	프리픽스: 0 서픽스: 우회	프리픽스: 0 서픽스: 우회	프리픽스: 0 서픽스: 우회
inter_sdc_flag	0	0	0	0	0	0
inter_sdc_resi_abs_minus1	0	0	0	0	0	0
inter_sdc_resi_sign_flag	0	0	0	0	0	0

[0419]

[0420] 대안적으로, 선택스 엘리먼트 depth_dc_flag 는, 상이한 파티션들 (제 1 또는 제 2) 이 이하에서 도시된 동일한 컨텍스트를 공유하는 방법으로 이진화된다.

[0421] 표 H-31 : 컨텍스트 코딩된 bin들을 갖는 신택스 엘리먼트들에 대한 ctxInc 의 배정

신택스 엘리먼트	binIdx					
	0	1	2	3	4	>=5
wedge_full_tab_idx	0	0	0	0	0	0
dmmdepth_dc_flag	0	na	na	na	na	na

[0422] 이 개시물의 기법들의 제 4 예는 위에서 설명된 제 3 예와 유사하다. 그러나, 제 4 예에서는, 인트라 예측 샘플 생성 동안에 이용된 이웃하는 참조 샘플들의 필터링이 DMM, CCM, 및 SDC 모드 of 적어도 서브-모드 SDCM_DMM_1 에 대해 제거된다. 위에서 설명된 바와 같이, 필터링은 서브-모드 SDCM_DC 를 포함하는 DC 인트라 예측 모드들에 대해 이미 디스에이블된다. 일부의 예들에서, 필터링은 또한, SDC 모드의 서브-모드 SDCM_Planar 에 대해 제거될 수도 있다.

[0424] 이 예에서, 비디오 인코더 (20) 는 이웃하는 참조 샘플들을 필터링하지 않으면서, 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 값들을 생성한다. 유사하게, 비디오 디코더 (30) 는 이웃하는 참조 샘플들을 필터링하지 않으면서, 심도 데이터의 수신된 잔차 값들 및 이웃하는 참조 샘플들에 기초하여 비디오 데이터의 심도 값들을 재구성한다.

[0425] 3D-HEVC 표준에 대한 추가들은 이하에서 회색으로 강조표시되고, 3D-HEVC 표준으로부터의 삭제들은 지움표시를 이용하여 표시된다.

[0426] **H.8.4.4.2.3 이웃하는 샘플들의 필터링 프로세스**

~~하위조항 8.4.4.2.3 에서의 사양들이 적용된다.~~

[0427] 이 프로세스에 대한 입력들은 다음과 같다.
 [0428]

[0429] - $x = -1, y = -1..nTbS * 2 - 1$, 그리고 $x = 0..nTbS * 2 - 1, y = -1$ 인 이웃하는 샘플들 $p[x][y]$

[0430] - 변환 블록 사이즈를 특정하는 변수 $nTbS$

[0431] 이 프로세스의 출력들은 $x = -1, y = -1..nTbS * 2 - 1$, 그리고 $x = 0..nTbS * 2 - 1, y = -1$ 인 필터링된 샘플들 $pF[x][y]$ 이다.

[0432] 변수 $filterFlag$ 는 다음과 같이 유도된다.

[0433] - 다음의 조건들 중의 하나 이상이 참일 경우, $filterFlag$ 는 0 과 동일하게 설정된다:

[0434] - $predModeIntra$ 는 INTRA_DC 와 동일하다.

~~- $predModeIntra$ 는 INTRA_ANGULAR34 보다 더 크다.~~

[0435] - $nTbS$ 는 4 와 동일하다.
 [0436]

[0437] - 이와 다를 경우, 다음이 적용된다:

[0438] - 변수 $minDistVerHor$ 는 $Min(Abs(predModeIntra - 26), Abs(predModeIntra - 10))$ 과 동일하게 설정된다.

[0439] - 변수 $intraHorVerDistThres[nTbS]$ 는 표 8-3 에서 특정된다.

[0440] - 변수 $filterFlag$ 는 다음과 같이 유도된다:

[0441] - $minDistVerHor$ 이 $intraHorVerDistThres[nTbS]$ 보다 더 클 경우, $filterFlag$ 는 1 과 동일하게 설정된다.

[0442] - 이와 다를 경우, $filterFlag$ 는 0 과 동일하게 설정된다.

표 8-3: 다양한 변환 블록 사이즈들에 대한 intraHorVerDistThres[nTbS] 의 사양

	nTbS = 8	nTbS = 16	nTbS = 32
intraHorVerDistThres[nTbS]	7	1	0

[0443]

[0444]

[0445]

[0446]

[0447]

[0448]

[0449]

[0450]

[0451]

[0452]

[0453]

[0454]

[0455]

[0456]

[0457]

[0458]

[0459]

[0460]

[0461]

[0462]

[0463]

[0464]

[0465]

[0466]

신택스 엘리먼트 filterFlag 가 1 과 동일할 때, 다음이 적용된다.

- 변수 biIntFlag 는 다음과 같이 유도된다:

- 다음의 조건들 중의 전부가 참일 경우, biIntFlag 는 1 과 동일하게 설정된다:

- strong_intra_smoothing_enabled_flag 는 1 과 동일하다.

- nTbS 는 32 와 동일하다

$$2 * p[nTbS - 1][-1] < (1 << (BitDepth_Y - 5))$$

$$- Abs(p[-1][-1] + p[-1][nTbS * 2 - 1] - 2 * p[-1][nTbS - 1]) < (1 << (BitDepth_Y - 5))$$

- 이와 다를 경우, biIntFlag 는 0 과 동일하게 설정된다.

- 필터링은 다음과 같이 수행된다:

- biIntFlag 가 1 과 동일할 경우, $x = -1$, $y = -1.63$, 그리고 $x = 0.63$, $y = -1$ 인 필터링된 샘플 값들 $pF[x][y]$ 은 다음과 같이 유도된다:

$$[1] pF[-1][-1] = p[-1][-1] \quad (8-30)$$

$$[2] y = 0.62 \text{ 에 대하여 } pF[-1][y] = ((63 - y) * p[-1][-1] + (y + 1) * p[-1][63] + 32) >> 6 \quad (8-31)$$

$$[3] pF[-1][63] = p[-1][63] \quad (8-32)$$

$$[4] x = 0.62 \text{ 에 대하여 } pF[x][-1] = ((63 - x) * p[-1][-1] + (x + 1) * p[63][-1] + 32) >> 6 \quad (8-33)$$

$$[5] pF[63][-1] = p[63][-1] \quad (8-34)$$

- 이와 다를 경우 (biIntFlag 가 0 과 동일함), $x = -1$, $y = -1..nTbS * 2 - 1$, 그리고 $x = 0..nTbS * 2 - 1$, $y = -1$ 인 필터링된 샘플 값들 $pF[x][y]$ 은 다음과 같이 유도된다:

$$[6] pF[-1][-1] = (p[-1][0] + 2 * p[-1][-1] + p[0][-1] + 2) >> 2 \quad (8-35)$$

$$[7] y = 0..nTbS * 2 - 2 \text{ 에 대하여 } pF[-1][y] = (p[-1][y + 1] + 2 * p[-1][y] + p[-1][y - 1] + 2) >> 2 \quad (8-36)$$

$$[8] pF[-1][nTbS * 2 - 1] = p[-1][nTbS * 2 - 1] \quad (8-37)$$

$$[9] x = 0..nTbS * 2 - 2 \text{ 에 대하여 } pF[x][-1] = (p[x - 1][-1] + 2 * p[x][-1] + p[x + 1][-1] + 2) >> 2 \quad (8-38)$$

$$[10] pF[nTbS * 2 - 1][-1] = p[nTbS * 2 - 1][-1] \quad (8-39)$$

도 10 은 심도 모델링 테이블에서의 3D 비디오 데이터에 대한 심도 데이터를 인코딩하기 위해 이용된 심도 인트라 모드들을 표시하는 비디오 인코더의 일 예의 동작을 예시하는 플로우차트이다. 일 예의 동작은 도 1 및 도 2 로부터의 비디오 인코더 (20) 에 대하여 설명된다. 비디오 인코더 (20) 는 인코딩된 비디오 데이터의 적어도 부분을 나타내는 비디오 비트스트림을 생성한다. 이 예에서, 비디오 데이터는 텍스처 데이터 및 대응하는 심도 데이터가 3D-HEVC 프로세스에 따라 코딩되는 멀티뷰 플러스 심도 포맷으로 표현된 3D 비디오 데이터를 포함한다. 비디오 데이터의 심도 데이터는 비디오 데이터의 대응하는 텍스처 데이터에 대한 심도 데이

터를 나타내는 심도 맵일 수도 있다.

- [0467] 이 개시물의 기법들에 따르면, 비디오 인코더 (20) 는 HEVC 인트라 코딩 모드들과는 상이한 복수의 심도 인트라 코딩 모드들 중의 선택된 하나를 이용하여 비디오 데이터에 대한 심도 데이터를 인코딩한다 (300). 심도 인트라 코딩 모드들은 DMM, SDC 모드 및 CCM 중의 적어도 2 개를 포함할 수도 있다. 비디오 인코더 (20) 는 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 선택된 심도 인트라 코딩 모드에 따라 예측된 심도 데이터의 잔차 값들을 생성할 수도 있다. 다음으로, 비디오 인코더 (20) 는 심도 인트라 모드들이 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 인코딩한다 (302). 신택스 엘리먼트는, 플래그의 제 1 값이 심도 인트라 코딩 모드들 중의 적어도 하나가 인에이블되는 것을 표시하고 플래그의 제 2 값이 심도 인트라 코딩 모드들 중의 어느 것도 인에이블되지 않음을 표시하는 심도 모드 플래그일 수도 있다. 다른 예들에서, 신택스 엘리먼트는, 플래그의 제 1 값이 심도 인트라 코딩 모드들의 전부나 인에이블되는 것을 표시하고 플래그의 제 2 값이 심도 인트라 코딩 모드들 중의 어느 것도 인에이블되지 않음을 표시하는 심도 모드 플래그일 수도 있다.
- [0468] 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 신택스 엘리먼트에 기초하여, 비디오 인코더 (20) 는 HEVC 인트라 코딩 모드들에 대한 신택스 엘리먼트들로부터 분리되어 있는 심도 인트라 코딩 모드들 중의 선택된 하나를 표시하는 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 추가로 인코딩한다 (304). 예를 들어, 심도 모델링 테이블은 DMM 의 서브-모드들 중의 하나, SDC 모드의 서브-모드들 중의 하나, 또는 CCM 과 같이, 심도 데이터를 인코딩하기 위해 이용된 특정 심도 인트라 코딩 모드를 표시하는 하나 이상의 신택스 엘리먼트들을 포함할 수도 있다. 일부의 예들에서, CCM 및/또는 DMM 서브-모드들 중의 적어도 하나는 HEVC 인트라 코딩 모드들로서 정의될 수도 있고, 그러므로, 심도 모델링 테이블 내에 포함되지 않을 수도 있다.
- [0469] 게다가, 심도 모델링 테이블은 특정된 심도 인트라 코딩 모드에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 포함할 수도 있다. 신택스 엘리먼트들의 세트는 심도 데이터의 잔차 값들이 존재하는지 여부를 표시하는 신택스 엘리먼트들과, 존재할 경우, 심도 데이터의 잔차 값들의 절대값 및 부호를 포함할 수도 있다. 일부의 예들에서, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 인트라 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시할 수도 있다.
- [0470] 심도 데이터가 심도 인트라 코딩 모드들과는 상이한 HEVC 인트라 코딩 모드들 중의 하나를 이용하여 인코딩되고, 신택스 엘리먼트가 심도 인트라 코드들 중의 어느 것도 인에이블되지 않음을 표시할 경우, 비디오 인코더 (20) 는 심도 모델링 테이블을 인코딩하거나 송신하지 않는다. 그 대신, 그 경우에는, 비디오 인코더 (20) 가 HEVC 인트라 코딩 모드들 중의 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 인코딩한다. 예를 들어, 표시된 HEVC 인트라 코딩 모드는 HEVC 인트라 코딩 모드 또는 HEVC I_PCM 모드일 수도 있다. 일부의 예들에서, 표시된 HEVC 인트라 코딩 모드는 HEVC 인트라 코딩 모드로서 정의된 DDM 의 서브-모드 또는 CCM 일 수도 있다. CCM 이 HEVC 인트라 코딩 모드로서 정의될 경우, CCM 에 대한 신택스 엘리먼트들은 심도 모델링 테이블 내에 포함되지 않지만, 그 대신에, 별도의 체인 코딩 테이블 내에 포함될 수도 있다.
- [0471] 도 11 은 심도 모델링 테이블로부터의 3D 비디오 데이터에 대한 심도 데이터를 디코딩하기 위하여 심도 인트라 코딩 모드를 선택하는 비디오 디코더의 일 예의 동작을 예시하는 플로우차트이다. 일 예의 동작은 도 1 및 도 3 으로부터의 비디오 디코더 (30) 에 대하여 설명된다. 비디오 디코더 (30) 는 인코딩된 비디오 데이터의 적어도 부분을 나타내는 비디오 비트스트림을 수신한다. 이 예에서, 비디오 데이터는 텍스처 데이터 및 대응하는 심도 데이터가 3D-HEVC 프로세스에 따라 코딩되는 멀티뷰 플러스 심도 포맷으로 표현된 3D 비디오 데이터를 포함한다. 비디오 데이터의 심도 데이터는 비디오 데이터의 대응하는 텍스처 데이터에 대한 심도 데이터를 나타내는 심도 맵일 수도 있다.
- [0472] 이 개시물의 기법들에 따르면, 비디오 디코더 (30) 는 복수의 심도 인트라 코딩 모드들 중의 임의의 것이 비디오 데이터에 대해 인에이블되는지 여부를 표시하는 비트스트림으로부터의 신택스 엘리먼트를 디코딩하고, 여기서, 심도 인트라 코딩 모드들은 HEVC 인트라 코딩 모드들과 상이하다 (310). 심도 인트라 코딩 모드들은 DMM, SDC 모드 및 CCM 중의 적어도 2 개를 포함할 수도 있다. 신택스 엘리먼트는, 플래그의 제 1 값이 심도 인트라 코딩 모드들 중의 적어도 하나가 인에이블되는 것을 표시하고 플래그의 제 2 값이 심도 인트라 코딩 모드들 중의 어느 것도 인에이블되지 않음을 표시하는 심도 모드 플래그일 수도 있다. 다른 예들에서, 신택스 엘리먼트는, 플래그의 제 1 값이 심도 인트라 코딩 모드들의 전부나 인에이블되는 것을 표시하고 플래그의 제 2 값이 심도 인트라 코딩 모드들 중의 어느 것도 인에이블되지 않음을 표시하는 심도 모드 플래그일 수도 있다. 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩한 후 (312 의 예 브랜

치), 비디오 디코더 (30) 는 또한, HEVC 인트라 코딩 모드들에 대한 선택스 엘리먼트들로부터 분리되어 있는 심도 인트라 코딩 모드들에 대한 선택스 엘리먼트들을 포함하는 심도 모델링 테이블을 디코딩한다 (316).

[0473] 선택스 엘리먼트가 심도 인트라 모드들 중의 어느 것도 인에이블되지 않음을 표시할 경우 (312 의 아니오 브랜치), 비디오 디코더 (30) 는 심도 모델링 테이블을 디코딩하지 않는다. 그 대신, 그 경우에는, 비디오 디코더 (30) 가 심도 인트라 코딩 모드들과는 상이한 HEVC 인트라 코딩 모드들 중의 하나를 선택한다 (314). 선택된 HEVC 인트라 코딩 모드는 HEVC 인트라 코딩 모드 또는 HEVC I_PCM 모드일 수도 있다. 일부의 예들에서, 선택된 HEVC 인트라 코딩 모드는 HEVC 인트라 코딩 모드로서 정의된 DDM 의 서브-모드 또는 CCM 일 수도 있다. HEVC 인트라 코딩 모드로서 정의된 CCM 의 경우, CCM 에 대한 선택스 엘리먼트들은 심도 모델링 테이블 내에 포함되지 않지만, 그 대신에, 별도의 체인 코딩 테이블 내에 포함될 수도 있다.

[0474] 심도 인트라 코딩 모드들이 비디오 데이터에 대해 인에이블되고 비디오 디코더 (30) 가 심도 모델링 테이블을 디코딩할 때 (312, 316 의 예 브랜치), 비디오 디코더 (30) 는 심도 모델링 테이블에서의 선택스 엘리먼트들에 기초하여 심도 인트라 코딩 모드들 중의 하나를 선택한다 (318). 예를 들어, 심도 모델링 테이블은 DMM 의 서브-모드들 중의 하나, SDC 모드와 서브-모드들 중의 하나, 또는 CCM 과 같이, 심도 데이터를 인코딩하기 위한 특정 심도 인트라 코딩 모드를 표시하는 하나 이상의 선택스 엘리먼트들을 포함할 수도 있다. 일부의 예들에서, CCM 및/또는 DMM 서브-모드들 중의 적어도 하나는 HEVC 인트라 코딩 모드들로서 정의될 수도 있고, 그러므로, 심도 모델링 테이블 내에 포함되지 않을 수도 있다.

[0475] 게다가, 심도 모델링 테이블은 특정된 심도 인트라 코딩 모드에 대한 심도 데이터의 잔차 정보를 표시하는 선택스 엘리먼트들의 세트를 포함할 수도 있다. 선택스 엘리먼트들의 세트는 심도 데이터의 잔차 값들이 존재하는지 여부를 표시하는 선택스 엘리먼트들과, 존재할 경우, 심도 데이터의 잔차 값들의 절대값 및 부호를 포함할 수도 있다. 일부의 예들에서, 선택스 엘리먼트들의 동일한 세트는 복수의 심도 인트라 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시할 수도 있다.

[0476] 다음으로, 비디오 디코더 (30) 는 선택된 심도 인트라 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩한다 (320). 예를 들어, 비디오 디코더 (30) 는 심도 데이터의 잔차 값들 및 이웃하는 참조 샘플들에 기초하여 선택된 심도 인트라 코딩 모드에 따라 비디오 데이터의 심도 값들을 재구성할 수도 있다. 다음으로, 비디오 데이터의 재구성된 심도 값들은 디스플레이 디바이스 상에서 3D 비디오 데이터를 제시하기 위하여 저장될 수도 있거나 이용될 수도 있다.

[0477] 도 12 는 2 개 이상의 심도 코딩 모드들에 대해 통합된 메커니즘들을 이용하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 비디오 인코더의 일 예의 동작을 예시하는 플로우차트이다. 일 예의 동작은 도 1 및 도 2 로부터의 비디오 인코더 (20) 에 대하여 설명된다. 비디오 인코더 (20) 는 인코딩된 비디오 데이터의 적어도 부분을 나타내는 비디오 비트스트림을 생성한다. 이 예에서, 비디오 데이터는 텍스처 데이터 및 대응하는 심도 데이터가 3D-HEVC 프로세스에 따라 코딩되는 멀티뷰 플러스 심도 포맷으로 표현된 3D 비디오 데이터를 포함한다. 비디오 데이터의 심도 데이터는 비디오 데이터의 대응하는 텍스처 데이터에 대한 심도 데이터를 나타내는 심도 맵일 수도 있다.

[0478] 비디오 인코더 (20) 는 비디오 데이터에 대한 심도 데이터를 인코딩하기 위한 복수의 심도 코딩 모드들 중의 하나를 선택한다 (330). 심도 코딩 모드들은 DMM, SDC 모드 및 CCM 중의 적어도 2 개를 포함하는 심도 인트라 코딩 모드들일 수도 있다. 다른 예들에서, 심도 코딩 모드들은 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드의 양자를 포함할 수도 있다. 그 경우, 심도 코딩 모드들은 DMM, SDC 모드, 인터 SDC 모드 및 CCM 중의 적어도 2 개를 포함할 수도 있다.

[0479] 이 개시물의 기법들에 따르면, 비디오 디코더 (30) 는 선택된 심도 코딩 모드에 따라 심도 데이터를 인코딩하기 전에 이웃하는 참조 샘플들을 필터링하지 않을 수도 있다 (332). 일부의 경우에는, 이웃하는 참조 샘플들의 필터링이 심도 인트라 코딩 모드들 중의 임의의 것에 대해 디스에이블될 수도 있다. 다른 경우에는, 이웃하는 참조 샘플들의 필터링이 DDM, CCM, 및 SDC 모드와 서브-모드들 SDCM_DC 및 SDCM_DMM_1 에 대해 디스에이블될 수도 있다. 다음으로, 비디오 인코더 (20) 는 비디오 데이터의 원래의 심도 값들 및 이웃하는 참조 샘플들에 기초하여 예측된 심도 데이터의 잔차 정보를 생성하기 위하여, 선택된 심도 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 인코딩한다 (334).

[0480] 비디오 인코더 (20) 는 비디오 데이터에 대한 선택된 심도 코딩 모드를 표시하는 하나 이상의 선택스 엘리먼트들을 인코딩한다 (336). 일부의 예들에서, 비디오 인코딩 (20) 은 상이한 심도 코딩 모드들의 각각을 표시

하기 위하여 상이한 신택스 엘리먼트들을 인코딩할 수도 있다. 다른 예들에서, 비디오 인코더 (20) 는 심도 인트라 모드들이 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 인코딩할 수도 있다.

신택스 엘리먼트가 심도 인트라 코딩 모드들이 인에이블되는 것을 표시할 때, 비디오 인코더 (20) 는 HEVC 인트라 코딩 모드들에 대한 신택스로부터 분리되어 있는 심도 인트라 코딩 모드들 중의 선택된 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 추가로 인코딩할 수도 있다.

[0481] 이 개시물의 기법들에 따르면, 다음으로, 비디오 인코더 (20) 는 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트를 인코딩하고, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시한다 (338). 예를 들어, 신택스 엘리먼트들의 세트는 심도 코딩 모드들 중의 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시할 수도 있다. 신택스 엘리먼트들의 세트는 DMM, SDC 모드, 인터 SDC 모드 및 CCM 중의 적어도 2 개 사이에서 공유될 수도 있다.

[0482] 비디오 인코더 (20) 는 단일 컨텍스트 모델 선택 방법을 이용하여 컨텍스트 모델들의 단일 세트로부터, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는 신택스 엘리먼트들의 세트에 대한 컨텍스트 모델들을 선택할 수도 있다. 예를 들어, 심도 코딩 모드들 중의 2 개 이상에 걸쳐 공유되는 신택스 엘리먼트들의 각각에 대하여 하나의 컨텍스트 모델만이 요구될 수도 있다. 이러한 방법으로, 컨텍스트 모델들의 동일한 세트, 예컨대, 공유된 신택스 엘리먼트들의 각각에 대한 하나가 심도 코딩 모드들 중의 임의의 것에 적용될 수도 있다.

[0483] 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트는 심도 데이터의 잔차 값들이 심도 코딩 모드들 중의 임의의 것에 대해 존재하는지 여부를 표시하는 신택스 엘리먼트를 포함할 수도 있다. 예를 들어, 신택스 엘리먼트들의 세트가 심도 인트라 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는 경우에는, 비디오 인코더 (20) 는 심도 데이터의 현재의 PU 에 대해 적용된 코딩 파티션들의 수에 기초하여 심도 데이터의 잔차 값들이 존재하는지 여부를 표시하는 신택스 엘리먼트에 대한 컨텍스트 모델을 선택할 수도 있다. 신택스 엘리먼트들의 세트가 인터 SDC 모드에 대한 잔차 정보를 표시하는 다른 경우에는, 비디오 인코더 (20) 는 심도 데이터의 현재의 CU 에 대한 코딩 파티션들의 수에 기초하여 심도 데이터의 잔차 값들이 존재하는지 여부를 표시하는 신택스 엘리먼트에 대한 컨텍스트 모델을 선택할 수도 있다.

[0484] 게다가, 신택스 엘리먼트들의 세트는 심도 코딩 모드들 중의 임의의 것에 대한 심도 데이터의 잔차 값들의 절대값 및 부호를 표시하는 신택스 엘리먼트들을 포함할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드들을 포함하는 심도 코딩 모드들 중의 임의의 것에 대한 단일 이진화 방법을 이용하여 심도 데이터의 잔차 값들의 절대값 및 부호를 표시하는 신택스 엘리먼트들을 이진화할 수도 있다. 단일 이진화 방법은 DMM, SDC 모드 또는 CCM 에 대해 기존에 이용된 이진화 방법들 중의 하나일 수도 있다.

[0485] 도 13 은 2 개 이상의 심도 코딩 모드들에 대해 통합된 메커니즘들을 이용하여 3D 비디오 데이터에 대한 심도 데이터의 잔차 정보를 결정하는 비디오 디코더의 일 예의 동작을 예시하는 플로우차트이다. 일 예의 동작은 도 1 및 도 3 으로부터의 비디오 디코더 (30) 에 대하여 설명된다. 비디오 디코더 (30) 는 인코딩된 비디오 데이터의 적어도 부분을 나타내는 비디오 비트스트림을 수신한다. 이 예에서, 비디오 데이터는 텍스처 데이터 및 대응하는 심도 데이터가 3D-HEVC 프로세스에 따라 코딩되는 멀티뷰 플러스 심도 포맷으로 표현된 3D 비디오 데이터를 포함한다. 비디오 데이터의 심도 데이터는 비디오 데이터의 대응하는 텍스처 데이터에 대한 심도 데이터를 나타내는 심도 맵일 수도 있다.

[0486] 이 개시물의 기법들에 따르면, 비디오 디코더 (30) 는 비디오 데이터에 대한 심도 데이터의 잔차 정보를 표시하는 비트스트림으로부터의 신택스 엘리먼트들의 세트를 디코딩하고, 여기서, 신택스 엘리먼트들의 동일한 세트는 복수의 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시한다 (350). 예를 들어, 신택스 엘리먼트들의 세트는 심도 코딩 모드들 중의 임의의 것에 대한 심도 데이터의 각각의 코딩 파티션에 대한 적어도 하나의 잔차 값을 표시할 수도 있다. 심도 코딩 모드들은 DMM, SDC 모드 및 CCM 중의 적어도 2 개를 포함하는 심도 인트라 코딩 모드들일 수도 있다. 다른 예들에서, 심도 코딩 모드들은 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드의 양자를 포함할 수도 있다. 그 경우, 심도 코딩 모드들은 DMM, SDC 모드, 인터 SDC 모드 및 CCM 중의 적어도 2 개를 포함할 수도 있다.

[0487] 비디오 디코더 (30) 는 단일 컨텍스트 모델 선택 방법을 이용하여 컨텍스트 모델들의 단일 세트로부터, 심도 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는 신택스 엘리먼트들의 세트에 대한 컨텍스트 모델들을

선택할 수도 있다. 예를 들어, 심도 코딩 모드들 중의 2 개 이상에 걸쳐 공유되는 신택스 엘리먼트들의 각각에 대하여 하나의 컨텍스트 모델만이 요구될 수도 있다. 이러한 방법으로, 컨텍스트 모델들의 동일한 세트, 예컨대, 공유된 신택스 엘리먼트들의 각각에 대한 하나가 심도 코딩 모드들 중의 임의의 것에 적용될 수도 있다.

[0488] 심도 데이터의 잔차 정보를 표시하는 신택스 엘리먼트들의 세트는 심도 데이터의 잔차 값들이 심도 코딩 모드들 중의 임의의 것에 대해 존재하는지 여부를 표시하는 신택스 엘리먼트를 포함할 수도 있다. 예를 들어, 신택스 엘리먼트들의 세트가 심도 인트라 코딩 모드들 중의 임의의 것에 대한 잔차 정보를 표시하는 경우에는, 비디오 디코더 (30) 는 심도 데이터의 현재의 PU 에 대해 적용된 코딩 파티션들의 수에 기초하여 심도 데이터의 잔차 값들이 존재하는지 여부를 표시하는 신택스 엘리먼트에 대한 컨텍스트 모델을 선택할 수도 있다. 신택스 엘리먼트들의 세트가 인터 SDC 모드에 대한 잔차 정보를 표시하는 다른 경우들에는, 비디오 디코더 (30) 는 심도 데이터의 현재의 CU 에 대한 코딩 파티션들의 수에 기초하여 심도 데이터의 잔차 값들이 존재하는지 여부를 표시하는 신택스 엘리먼트에 대한 컨텍스트 모델을 선택할 수도 있다.

[0489] 게다가, 신택스 엘리먼트들의 세트는 심도 코딩 모드들 중의 임의의 것에 대한 심도 데이터의 잔차 값들의 절대값 및 부호를 표시하는 신택스 엘리먼트들을 포함할 수도 있다. 예를 들어, 비디오 디코더 (30) 는 심도 인트라 코딩 모드들 및 심도 인터 코딩 모드들을 포함하는 심도 코딩 모드들 중의 임의의 것에 대한 단일 이진화 방법을 이용하여 심도 데이터의 잔차 값들의 절대값 및 부호를 표시하는 신택스 엘리먼트들을 이진화할 수도 있다. 단일 이진화 방법은 DMM, SDC 모드 또는 CCM 에 대해 기존에 이용된 이진화 방법들 중의 하나일 수도 있다.

[0490] 비디오 디코더 (30) 는 또한, 비디오 데이터에 대한 심도 코딩 모드들 중의 하나를 표시하는 비트스트림으로부터의 하나 이상의 신택스 엘리먼트들을 디코딩한다 (352). 일부의 예들에서, 비디오 디코더 (30) 는 상이한 심도 코딩 모드들의 각각을 표시하기 위하여 상이한 신택스 엘리먼트들을 디코딩할 수도 있다. 다른 예들에서, 비디오 디코더 (30) 는 심도 인트라 코딩 모드들 중의 임의의 것이 비디오 데이터에 대해 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩할 수도 있다. 심도 인트라 코딩 모드들이 인에이블되는 것을 표시하는 신택스 엘리먼트를 디코딩한 후, 비디오 디코더 (30) 는 또한, HEVC 인트라 코딩 모드들에 대한 신택스 엘리먼트들로부터 분리되어 있는 심도 인트라 코딩 모드들 중의 하나를 표시하는 하나 이상의 신택스 엘리먼트들을 포함하는 심도 모델링 테이블을 디코딩할 수도 있다.

[0491] 이 개시물의 기법들에 따르면, 비디오 디코더 (30) 는 표시된 심도 코딩 모드에 따라 심도 데이터를 인코딩하기 전에 이웃하는 참조 샘플들을 필터링하지 않을 수도 있다 (354). 일부의 경우들에는, 이웃하는 참조 샘플들의 필터링이 심도 인트라 코딩 모드들 중의 임의의 것에 대해 디스에이블될 수도 있다. 다른 경우들에는, 이웃하는 참조 샘플들의 필터링이 DDM, CCM, 및 SDC 모드의 적어도 서브-모드들 SDCM_DC 및 SDCM_DMM_1 에 대해 디스에이블될 수도 있다. 다음으로, 비디오 디코더 (30) 는 심도 데이터의 잔차 정보 및 이웃하는 참조 샘플들에 기초하여 비디오 데이터의 심도 값들을 재구성하기 위하여, 표시된 심도 코딩 모드를 이용하여 비디오 데이터에 대한 심도 데이터를 디코딩한다 (356). 다음으로, 비디오 데이터의 재구성된 심도 값들은 디스플레이 디바이스 상에서 3D 비디오 데이터를 제시하기 위하여 저장될 수도 있거나 이용될 수도 있다.

[0492] 위에서 설명된 기법들은 비디오 인코더 (20) (도 1 및 도 2) 및/또는 비디오 디코더 (30) (도 1 및 도 3) 에 의해 수행될 수도 있고, 이들 양자는 일반적으로 비디오 코더로서 지칭될 수도 있다. 게다가, 비디오 코딩은 일반적으로, 적용가능한 바와 같이 비디오 인코딩 및/또는 비디오 디코딩을 지칭할 수도 있다. 이 개시물의 기법들은 일반적으로 3D-HEVC 에 대하여 설명되지만, 기법들은 이러한 방법으로 제한되지 않는다. 위에서 설명된 기법들은 또한, 다른 현재의 표준들 또는 아직 개발되지 않은 미래의 표준들에 적용가능할 수도 있다.

[0493] 예에 따라서는, 본원에서 설명된 방법들 중의 임의의 것의 어떤 액트들 또는 이벤트들이 상이한 시퀀스에서 수행될 수 있거나, 추가될 수도 있거나, 병합될 수도 있거나, 또는 함께 배제 (예컨대, 모든 설명된 액트들 또는 이벤트들이 방법의 실시를 위해 필요한 것은 아님) 될 수도 있다는 것을 이해해야 한다. 또한, 어떤 예들에서는, 액트들 또는 이벤트들이 순차적인 것이 아니라, 예컨대, 멀티-스레딩된 (multi-threaded) 프로세싱, 인터럽트 프로세싱, 또는 다수의 프로세서들을 통해 동시에 수행될 수도 있다. 게다가, 이 개시물의 어떤 양태들은 명료함의 목적들을 위하여 단일 모듈 또는 유닛에 의해 수행되는 것으로 설명되지만, 이 개시물의 기법들은 비디오 코더와 연관된 유닛들 또는 모듈들의 조합에 의해 수행될 수도 있는 것을 이해해야 한다.

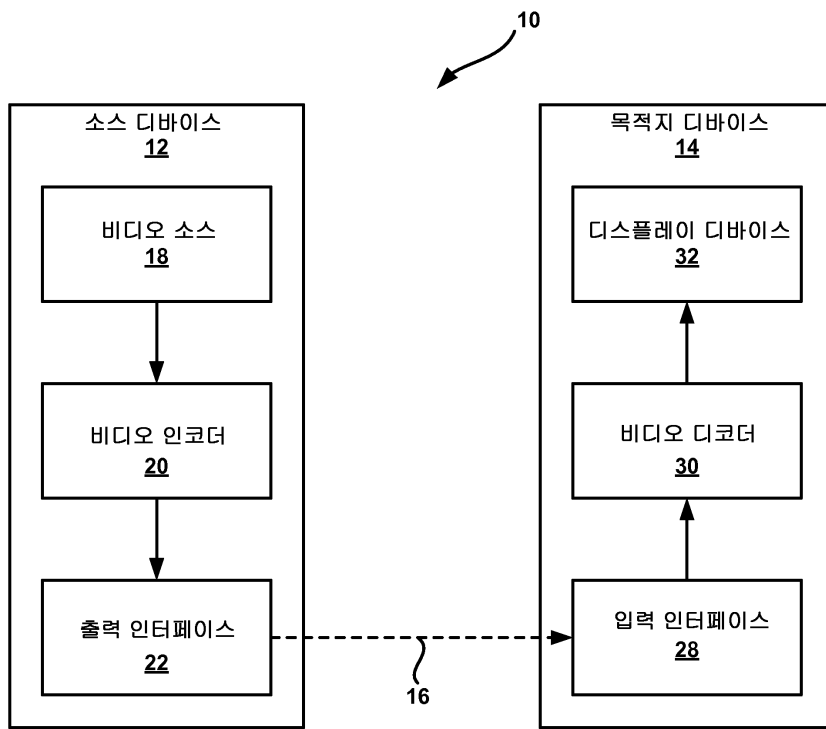
[0494] 기법들의 다양한 양태들의 특별한 조합들이 위에서 설명되지만, 이 조합들은 이 개시물에서 설명된 기법들의 예를 단지 예시하기 위하여 제공된다. 따라서, 이 개시물의 기법들은 이 예의 조합들에 제한되지 않아야 하

고, 이 개시물에서 설명된 기법들의 다양한 양태들의 임의의 상상가능한 조합을 망라할 수도 있다.

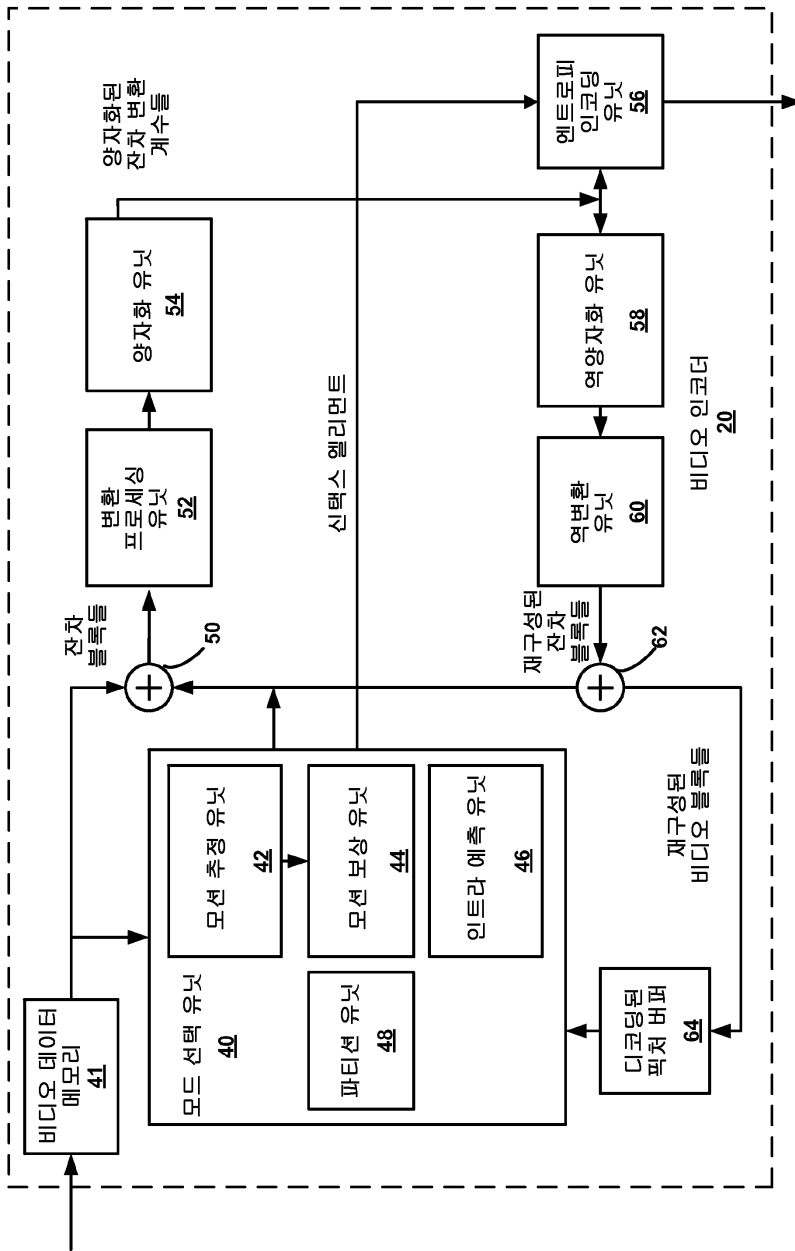
- [0495] 하나 이상의 예들에서는, 설명된 기능들이 하드웨어, 소프트웨어, 펌웨어, 또는 그 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현될 경우, 기능들은 하나 이상의 명령들 또는 코드로서, 컴퓨터-판독가능한 저장 매체 상에 저장되거나 통신 매체를 통해 송신될 수도 있고, 하드웨어-기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터-판독가능한 매체들은 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터-판독가능한 저장 매체들, 또는 예컨대, 통신 프로토콜에 따라 하나의 장소로부터 또 다른 장소로 컴퓨터 프로그램의 전송을 용이하게 하는 통신 매체들을 포함할 수도 있다.
- [0496] 이러한 방식으로, 컴퓨터-판독가능한 매체들은 일반적으로 (1) 비-일시적인 유형의 컴퓨터-판독가능한 저장 매체들, 또는 (2) 신호 또는 반송파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체들은 이 개시물에서 설명된 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 이용가능한 저장 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터-판독가능한 저장 매체 및 패키징 재료들을 포함할 수도 있다.
- [0497] 제한이 아닌 예로서, 이러한 컴퓨터-판독가능한 저장 매체들은 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장, 자기 디스크 저장, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 명령들 또는 데이터 구조들의 형태로 희망하는 프로그램 코드를 저장하기 위해 이용될 수 있으며 컴퓨터에 의해 액세스될 수 있는 임의의 다른 저장 매체를 포함할 수 있다. 또한, 임의의 접속은 통신 매체로 적절하게 칭해진다. 예를 들어, 동축 케이블, 광섬유 케이블, 트위스트 페어 (twisted pair), 디지털 가입자 회선 (digital subscriber line; DSL), 또는 무선 기술들 예컨대, 적외선, 라디오 (radio), 및 마이크로파 (microwave) 를 이용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 명령들이 송신될 경우, 동축 케이블, 광섬유 케이블, 트위스트 페어, DSL, 또는 무선 기술들 예컨대, 적외선, 라디오, 및 마이크로파는 통신 매체의 정의 내에 포함된다.
- [0498] 그러나, 컴퓨터-판독가능한 저장 매체들 및 데이터 저장 매체들은 접속들, 반송파들, 신호들, 또는 다른 일시적 매체 (transient medium) 들을 포함하는 것이 아니라, 그 대신에, 비-일시적인, 유형의 저장 매체들에 관한 것이라는 것을 이해해야 한다. 본원에서 이용된 바와 같은 디스크 (disk) 및 디스크 (disc) 는 콤팩트 디스크 (compact disc; CD), 레이저 디스크 (laser disc), 광학 디스크 (optical disc), 디지털 다기능 디스크 (digital versatile disc; DVD), 플로피 디스크 (floppy disk) 및 블루레이 디스크 (blu-ray disc) 를 포함하고, 여기서, 디스크 (disk) 들은 통상 데이터를 자기적으로 재생하는 반면, 디스크 (disc) 들은 데이터를 레이저로 광학적으로 재생한다. 상기한 것의 조합들은 컴퓨터-판독가능한 매체들의 범위 내에 또한 포함되어야 한다.
- [0499] 명령들은 하나 이상의 디지털 신호 프로세서 (digital signal processor; DSP) 들, 범용 마이크로프로세서들, 주문형 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍가능한 로직 어레이 (field programmable logic array; FPGA) 들, 또는 다른 등가의 통합된 또는 개별 로직 회로부와 같은 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 본원에서 이용된 바와 같은 용어 "프로세서" 는 상기한 구조, 또는 본원에서 설명된 기법들의 구현을 위해 적당한 임의의 다른 구조 중의 임의의 것을 지칭할 수도 있다. 게다가, 일부의 양태들에서는, 본원에서 설명된 기능성이 인코딩 및 디코딩을 위해 구성되거나 조합된 코덱 (codec) 내에 통합된 전용 하드웨어 및/또는 소프트웨어 모듈들 내에서 제공될 수도 있다. 또한, 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.
- [0500] 이 개시물의 기법들은 무선 핸드셋 (wireless handset), 집적 회로 (integrated circuit; IC) 또는 IC 들의 세트 (예컨대, 칩셋) 를 포함하는 광범위한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들, 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위하여 이 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하지는 않는다. 오히려, 위에서 설명된 바와 같이, 다양한 유닛들은 코덱 하드웨어 유닛 내에 조합될 수도 있거나, 적당한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명된 바와 같은 하나 이상의 프로세서들을 포함하는 상호동작하는 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.
- [0501] 개시물의 다양한 양태들이 설명되었다. 이러한 그리고 다른 양태들은 다음의 청구항들의 범위 내에 있다.

도면

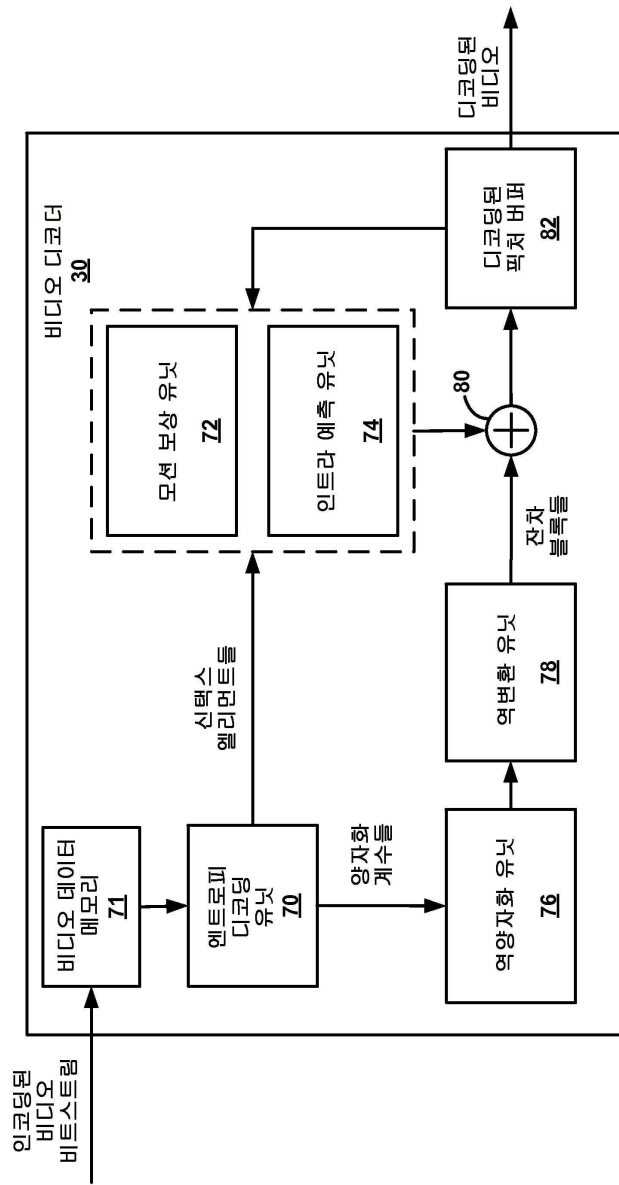
도면1



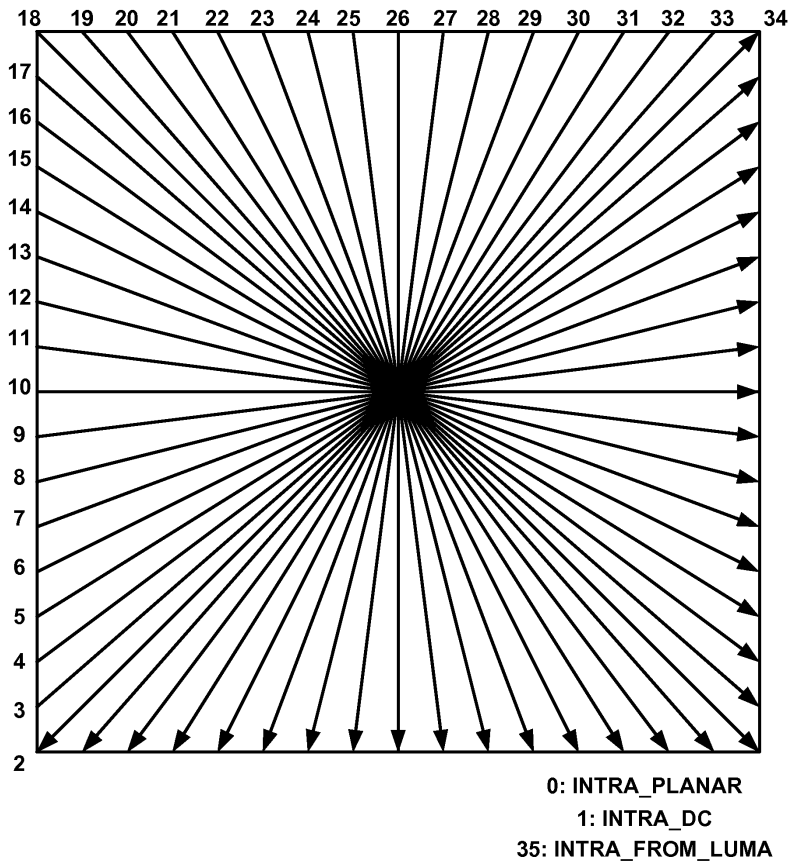
도면2



도면3



도면4

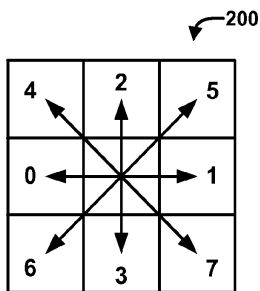


도면5

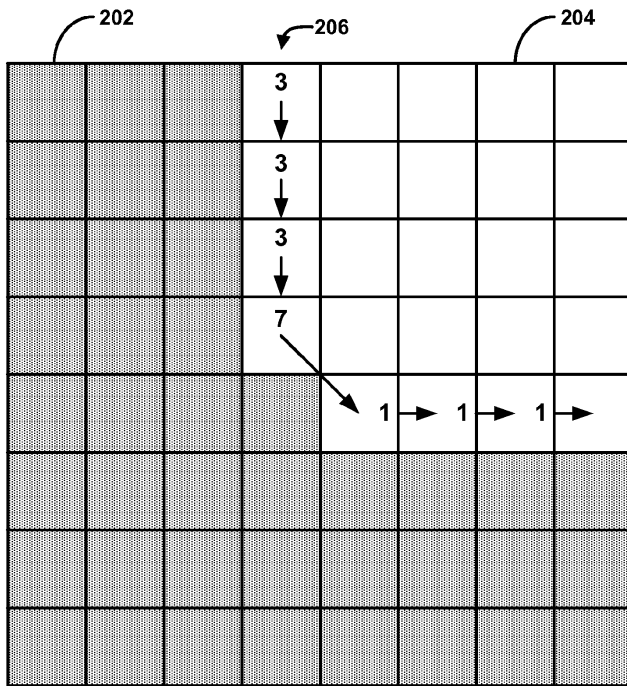
각도 인트라 예측 모드들에 대한 V 및 H의 값들

인트라 모드 인덱스	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
H	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
V	32	26	21	17	13	9	5	2	0	-2	-5	-9	-13	-17	-21	-26	-32
인트라 모드 인덱스	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	
H	26	21	17	13	9	5	2	0	2	5	9	13	17	21	26	32	
V	-32	-32	-32	-32	-32	-32	-32	-32	32	32	32	32	32	32	32	32	32

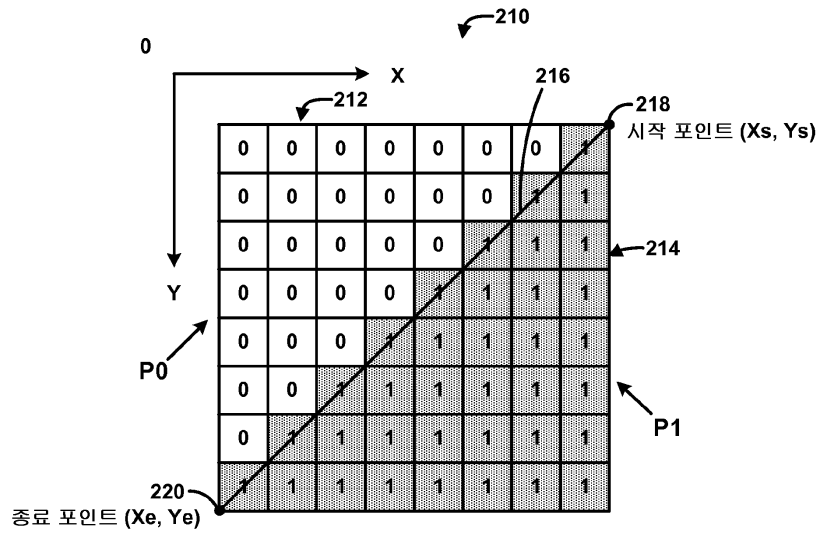
도면6a



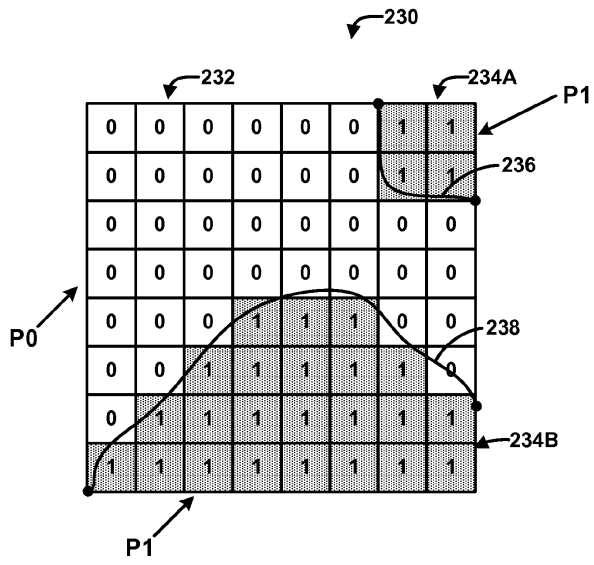
도면6b



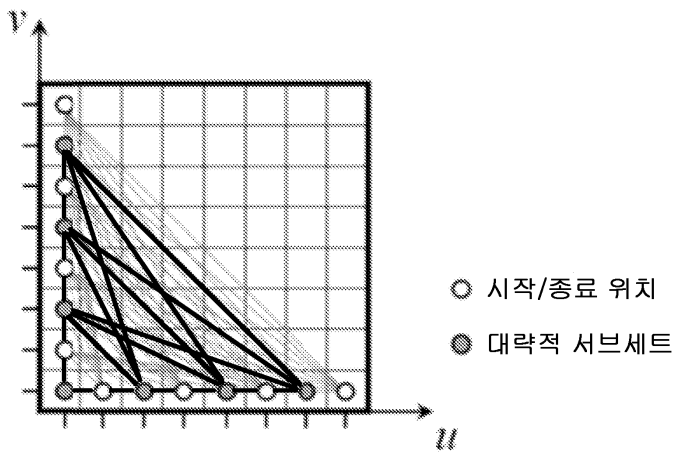
도면7a



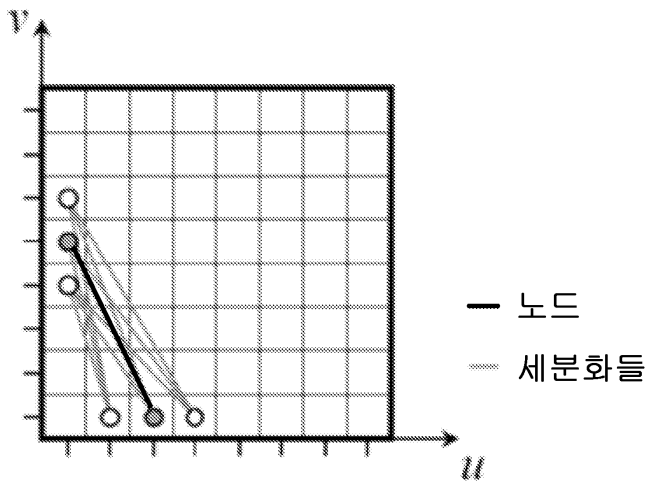
도면7b



도면8a



도면8b

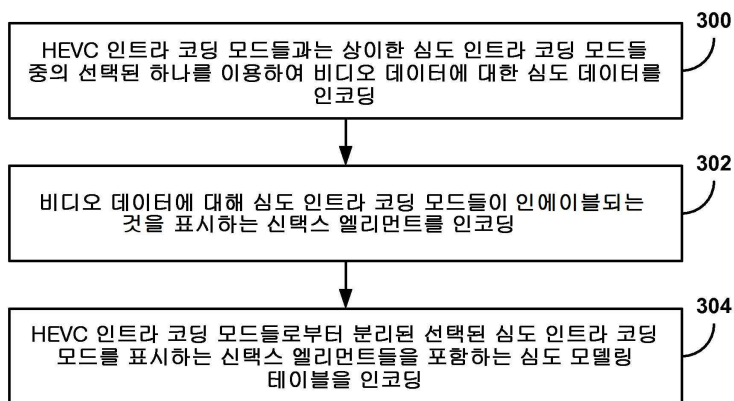


도면9

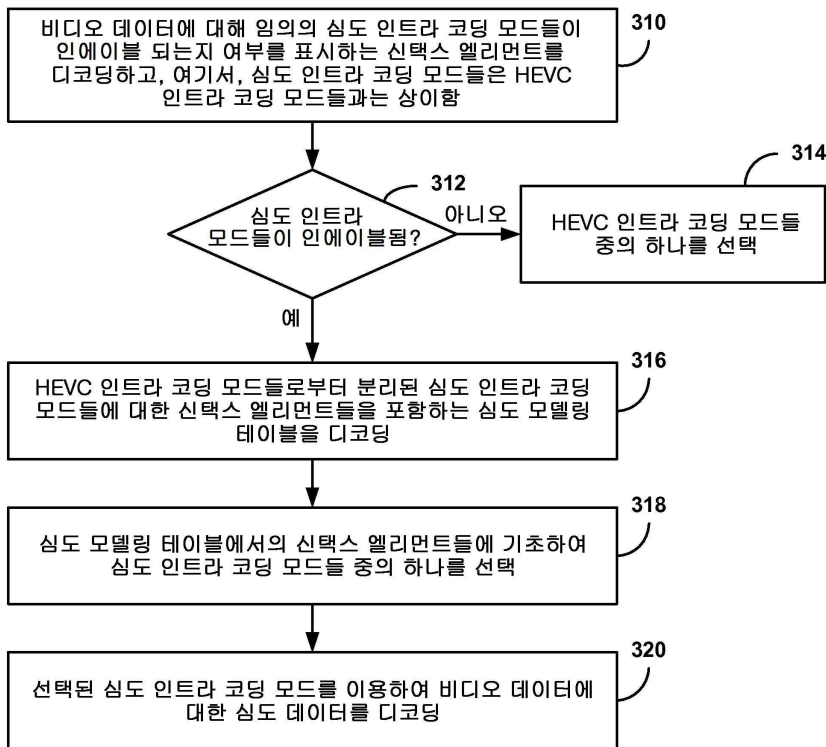
상이한 PU 사이즈들에 대한 이용가능한 DMM 모드들 및 웨지렛 패턴들의 수

PU 사이즈	이용가능한 DMM 모드들	웨지렛 패턴들의 수
4x4	DMM_mode_1 and DMM_mode_3	86
8x8	DMM_mode_1/2/3/4	782
16x16	DMM_mode_1/2/3/4	1394
32x32	DMM_mode_1/2/3/4	1503
64x64	없음	없음

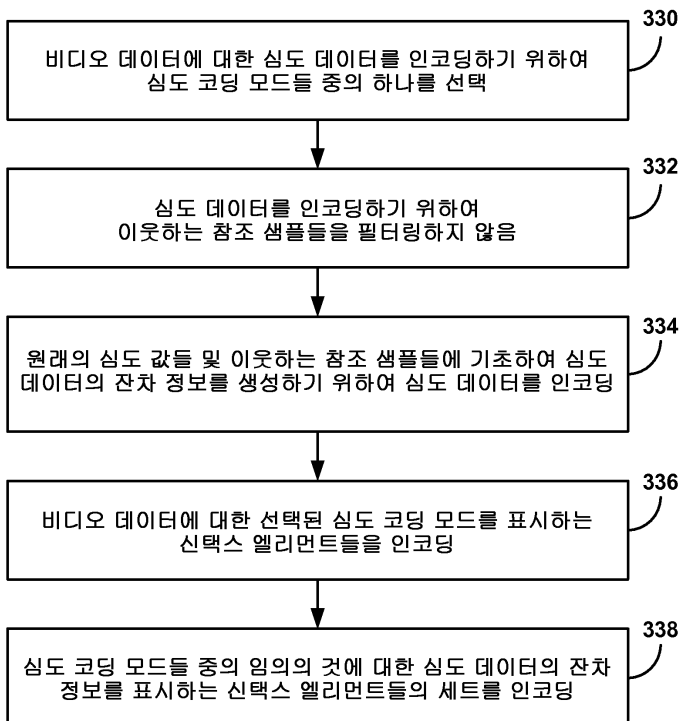
도면10



도면11



도면12



도면13

