(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0033760
(43) 공개일자 2020년03월30일

(51) 국제특허분류(Int. Cl.)
H04N 19/103 (2014.01) *H04N 19/117* (2014.01)
H04N 19/119 (2014.01) *H04N 19/124* (2014.01)
H04N 19/13 (2014.01)
(52) CPC특허분류
H04N 19/103 (2015.01)
H04N 19/117 (2015.01)
(21) 출원번호 10-2019-0115767
(22) 출원일자 2019년09월20일
심사청구일자 없음
(30) 우선권주장
1020180113148 2018년09월20일 대한민국(KR)

(71) 출원인
한국전자통신연구원
대전광역시 유성구 가정로 218 (가정동)
한밭대학교 산학협력단
대전광역시 유성구 동서대로 125 (덕명동)
(72) 발명자
이하현
서울특별시 중랑구 동일로102길 34-8, 402호(면목동, 우암하우스 B동)
강정원
대전광역시 유성구 지족로 362, 303동 303호(지족동, 반석마을3단지아파트)
(뒷면에 계속)
(74) 대리인
성병기

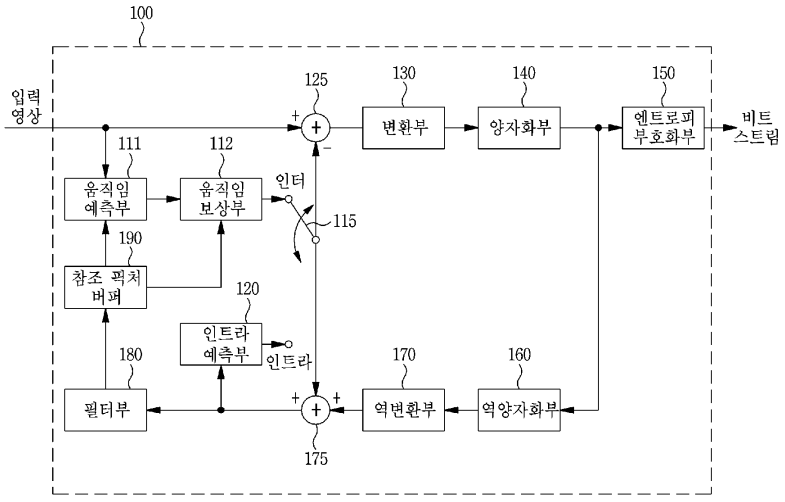
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 **영상 부호화/복호화 방법, 장치 및 비트스트림을 저장한 기록 매체**

(57) 요 약

영상 부호화/복호화 방법 및 장치가 제공된다. 본 발명에 따른 영상 복호화 방법은, 영상을 분할하여 region을 식별하는 단계 및 상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 상기 region의 예측 모드를 결정하는 단계를 포함할 수 있다. 상기 결정된 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정될 수 있다.

대 표 도 - 도1



(52) CPC특허분류
H04N 19/119 (2015.01)
H04N 19/124 (2015.01)
H04N 19/13 (2015.01)
(72) 발명자
임성창
대전광역시 유성구 도안동로 523, 202동 1801호(봉명동, 베르디움)
이진호
대전광역시 유성구 지족동로 124, 102동 1904호(지족동, 노은리슈빌3)
김휘용
대전광역시 유성구 은구비남로 34, 810동 201호(노은동, 열매마을8단지)
최해철
대전광역시 유성구 반석서로 109, 705동 904호(반석동, 반석마을7단지아파트)
권대혁
대전광역시 서구 대덕대로127번길 14, 301호(갈마동)

이 발명을 지원한 국가연구개발사업
과제고유번호 2016-0-00572
부처명 과학기술정보통신부
연구관리전문기관 정보통신기획평가원
연구사업명 ETRI연구개발지원사업
연구과제명 초고실감 미디어 서비스 실현을 위해 HEVC/3DA 대비 2배 압축을 제공하는 5세대 비디오/오디오 표준 핵심 기술 개발 및 표준화
기 여 율 1/1
주관기관 ETRI
연구기간 2019.01.01 ~ 2019.12.31

명 세 서

청구범위

청구항 1

영상을 분할하여 region을 식별하는 단계; 및

상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 상기 region의 예측 모드를 결정하는 단계를 포함하고,

상기 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정되는 영상 복호화 방법.

청구항 2

제1항에 있어서,

상기 region은 CTU(Coding Tree Unit)의 분할 트리 구조의 하나의 노드 및 그 하위 노드에 해당하는 영상 복호화 방법.

청구항 3

제1항에 있어서,

상기 region의 크기는 상기 region의 가로, 세로, 면적 및 픽셀수 중 적어도 하나를 포함하는 영상 복호화 방법.

청구항 4

제1항에 있어서,

상기 region의 분할 형태는,

상기 region의 분할 여부, 분할 트리 및 분할 방향 중 적어도 하나를 포함하는 영상 복호화 방법.

청구항 5

제1항에 있어서,

상기 region의 부호화 파라미터는,

상기 region을 포함하는 슬라이스 타입 및 상기 region의 예측 모드에 관한 신택스 요소 중 적어도 하나를 포함하는 영상 복호화 방법.

청구항 6

제1항에 있어서,

상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 화면내 예측으로 결정되는 영상 복호화 방법.

청구항 7

제6항에 있어서,

상기 소정의 크기는 64 픽셀이고, 상기 분할 형태는 쿼드트리 분할 또는 삼분할트리 분할인 영상 복호화 방법.

청구항 8

제6항에 있어서,

상기 소정의 크기는 32 픽셀이고, 상기 분할 형태는 이진트리 분할인 영상 복호화 방법.

청구항 9

제1항에 있어서,

상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 상기 region을 포함하는 슬라이스 타입에 기초하여 결정되는 영상 복호화 방법.

청구항 10

제9항에 있어서,

상기 슬라이스 타입이 I 슬라이스가 아닌 경우,

상기 region의 예측 모드는 상기 region의 부호화 파라미터에 기초하여 결정되는 영상 복호화 방법.

청구항 11

영상을 분할하여 region을 식별하는 단계; 및

상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 상기 region의 예측 모드를 결정하는 단계를 포함하고,

상기 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정되는 영상 부호화 방법.

청구항 12

제11항에 있어서,

상기 region은 CTU(Coding Tree Unit)의 분할 트리 구조의 하나의 노드 및 그 하위 노드에 해당하는 영상 부호화 방법.

청구항 13

제11항에 있어서,

상기 region의 크기는 상기 region의 가로, 세로, 면적 및 픽셀수 중 적어도 하나를 포함하는 영상 부호화 방법.

청구항 14

제11항에 있어서,

상기 region의 분할 형태는,

상기 region의 분할 여부, 분할 트리 및 분할 방향 중 적어도 하나를 포함하는 영상 부호화 방법.

청구항 15

제11항에 있어서,

상기 region의 부호화 파라미터는,

상기 region을 포함하는 슬라이스 타입 및 상기 region의 예측 모드에 관한 신택스 요소 중 적어도 하나를 포함하는 영상 부호화 방법.

청구항 16

제11항에 있어서,

상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 화면내 예측으로 결정되는 영상 부호화 방법.

청구항 17

제16항에 있어서,

상기 소정의 크기는 64 픽셀이고, 상기 분할 형태는 쿼드트리 분할 또는 삼분할트리 분할인 영상 부호화 방법.

청구항 18

제16항에 있어서,

상기 소정의 크기는 32 픽셀이고, 상기 분할 형태는 이진트리 분할인 영상 복호화 방법.

청구항 19

제11항에 있어서,

상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 상기 region을 포함하는 슬라이스 타입에 기초하여 결정되는 영상 부호화 방법.

청구항 20

영상 복호화 장치에 의해 수신되고 복호화되어 영상을 복원하는데 이용되는 비트스트림을 저장한 컴퓨터 판독 가능한 기록매체로서, 상기 비트스트림은 region의 분할 및 예측에 관한 정보를 포함하고,

상기 region의 분할에 관한 정보는 상기 영상을 분할하여 상기 region을 식별하기 위해 이용되고,

상기 region의 예측에 관한 정보는 상기 region의 예측 모드를 결정하기 위해 이용되고,

상기 region의 예측 모드는, 상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 결정되고,

상기 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정되는 기록매체.

발명의 설명

기 술 분 야

[0001] 본 발명은 영상의 부호화/복호화 방법 및 장치에 관한 것으로, 보다 상세하게는 블록의 분할 구조 및/또는 예측 모드를 시그널링하는 영상의 부호화/복호화 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] 최근 HD(High Definition) 영상 및 UHD(Ultra High Definition) 영상과 같은 고해상도, 고품질의 영상에 대한 수요가 다양한 응용 분야에서 증가하고 있다. 영상 데이터가 고해상도, 고품질이 될수록 기존의 영상 데이터에 비해 상대적으로 데이터량이 증가하기 때문에 기존의 유무선 광대역 회선과 같은 매체를 이용하여 영상 데이터를 전송하거나 기존의 저장 매체를 이용해 저장하는 경우, 전송 비용과 저장 비용이 증가하게 된다. 영상 데이터가 고해상도, 고품질화 됨에 따라 발생하는 이러한 문제들을 해결하기 위해서는 더 높은 해상도 및 화질을 갖는 영상에 대한 고효율 영상 부호화(encoding)/복호화(decoding) 기술이 요구된다.

[0003] 영상 압축 기술로 현재 픽처의 이전 또는 이후 픽처로부터 현재 픽처에 포함된 화소값을 예측하는 화면간 예측 기술, 현재 픽처 내의 화소 정보를 이용하여 현재 픽처에 포함된 화소값을 예측하는 화면내 예측 기술, 잔여 신호의 에너지를 압축하기 위한 변환 및 양자화 기술, 출현 빈도가 높은 값에 짧은 부호를 할당하고 출현 빈도가 낮은 값에 긴 부호를 할당하는 엔트로피 부호화 기술 등 다양한 기술이 존재하고 이러한 영상 압축 기술을 이용해 영상 데이터를 효과적으로 압축하여 전송 또는 저장할 수 있다.

발명의 내용

해결하려는 과제

[0004] 본 발명은 부호화/복호화 효율이 향상된 영상 부호화/복호화 방법 및 장치를 제공하는 것을 목적으로 한다.

[0005] 또한, 본 발명은 블록의 분할 구조 및/또는 예측 모드의 시그널링을 효과적으로 수행하는 영상 부호화/복호화 방법 및 장치를 제공하는 것을 목적으로 한다.

[0006] 또한, 본 발명은 본 발명에 따른 영상 부호화 방법 또는 장치에 의해 생성된 비트스트림을 저장한 기록 매체를

제공하는 것을 목적으로 한다.

[0007] 또한, 본 발명은 본 발명에 따른 영상 복호화 장치에 의해 수신되고 복호화되어 영상의 복원에 이용되는 비트스트림을 저장한 기록 매체를 제공하는 것을 목적으로 한다.

과제의 해결 수단

[0008] 본 발명의 일 실시예에 따른 영상 복호화 방법은, 영상을 분할하여 region을 식별하는 단계 및 상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 상기 region의 예측 모드를 결정하는 단계를 포함하고, 상기 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정될 수 있다.

[0009] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 region은 CTU(Coding Tree Unit)의 분할 트리 구조의 하나의 노드 및 그 하위 노드에 해당할 수 있다.

[0010] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 region의 크기는 상기 region의 가로, 세로, 면적 및 픽셀수 중 적어도 하나를 포함할 수 있다.

[0011] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 region의 분할 형태는, 상기 region의 분할 여부, 분할 트리 및 분할 방향 중 적어도 하나를 포함할 수 있다.

[0012] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 region의 부호화 파라미터는, 상기 region을 포함하는 슬라이스 타입 및 상기 region의 예측 모드에 관한 신택스 요소 중 적어도 하나를 포함할 수 있다.

[0013] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 화면내 예측으로 결정될 수 있다.

[0014] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 소정의 크기는 64 픽셀이고, 상기 분할 형태는 쿼드트리 분할 또는 삼분할트리 분할일 수 있다.

[0015] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 소정의 크기는 32 픽셀이고, 상기 분할 형태는 이진트리 분할일 수 있다.

[0016] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 상기 region을 포함하는 슬라이스 타입에 기초하여 결정될 수 있다.

[0017] 본 발명에 따른 영상 복호화 방법에 있어서, 상기 슬라이스 타입이 I 슬라이스가 아닌 경우, 상기 region의 예측 모드는 상기 region의 부호화 파라미터에 기초하여 결정될 수 있다.

[0018] 본 발명의 다른 실시예에 따른 영상 부호화 방법은, 영상을 분할하여 region을 식별하는 단계 및 상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 상기 region의 예측 모드를 결정하는 단계를 포함하고, 상기 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정될 수 있다.

[0019] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 region은 CTU(Coding Tree Unit)의 분할 트리 구조의 하나의 노드 및 그 하위 노드에 해당할 수 있다.

[0020] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 region의 크기는 상기 region의 가로, 세로, 면적 및 픽셀수 중 적어도 하나를 포함할 수 있다.

[0021] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 region의 분할 형태는, 상기 region의 분할 여부, 분할 트리 및 분할 방향 중 적어도 하나를 포함할 수 있다.

[0022] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 region의 부호화 파라미터는, 상기 region을 포함하는 슬라이스 타입 및 상기 region의 예측 모드에 관한 신택스 요소 중 적어도 하나를 포함할 수 있다.

[0023] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 화면내 예측으로 결정될 수 있다.

[0024] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 소정의 크기는 64 픽셀이고, 상기 분할 형태는 쿼드트리 분할

또는 삼분할트리 분할일 수 있다.

[0025] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 소정의 크기는 32 픽셀이고, 상기 분할 형태는 이진트리 분할일 수 있다.

[0026] 본 발명에 따른 영상 부호화 방법에 있어서, 상기 region의 크기가 소정의 크기이고, 상기 분할 형태가 소정의 분할 형태인 경우, 상기 region의 예측 모드는 상기 region을 포함하는 슬라이스 타입에 기초하여 결정될 수 있다.

[0027] 본 발명의 또 다른 실시예에 따른 컴퓨터 판독 가능한 기록 매체는, 영상 복호화 장치에 의해 수신되고 복호화되어 영상을 복원하는데 이용되는 비트스트림을 저장한 컴퓨터 판독 가능한 기록 매체로서, 상기 비트스트림은 region의 분할 및 예측에 관한 정보를 포함하고, 상기 region의 분할에 관한 정보는 상기 영상을 분할하여 상기 region을 식별하기 위해 이용되고, 상기 region의 예측에 관한 정보는 상기 region의 예측 모드를 결정하기 위해 이용되고, 상기 region의 예측 모드는, 상기 region의 크기, 분할 형태 및 region의 부호화 파라미터 중 적어도 하나에 기초하여 결정되고, 상기 region의 예측 모드는 상기 region에 포함된 모든 부호화 블록의 예측 모드로 결정될 수 있다.

[0028] 본 발명의 또 다른 실시예에 따른 컴퓨터 판독 가능한 기록 매체는, 본 발명에 따른 영상 부호화 방법 및/또는 장치에 의해 생성된 비트스트림을 저장할 수 있다.

발명의 효과

[0029] 본 발명에 따르면, 부호화/복호화 효율이 향상된 영상 부호화/복호화 방법 및 장치가 제공될 수 있다.

[0030] 또한, 본 발명에 따르면, 블록의 분할 구조 및/또는 예측 모드의 시그널링을 효과적으로 수행하는 영상 부호화/복호화 방법 및 장치가 제공될 수 있다.

[0031] 또한, 본 발명에 따르면, 본 발명에 따른 영상 부호화 방법 또는 장치에 의해 생성된 비트스트림을 저장한 기록 매체가 제공될 수 있다.

[0032] 또한, 본 발명에 따르면, 본 발명에 따른 영상 복호화 장치에 의해 수신되고 복호화되어 영상의 복원에 이용되는 비트스트림을 저장한 기록 매체가 제공될 수 있다.

도면의 간단한 설명

[0033] 도 1은 본 발명이 적용되는 부호화 장치의 일 실시예에 따른 구성을 나타내는 블록도이다.

도 2는 본 발명이 적용되는 복호화 장치의 일 실시예에 따른 구성을 나타내는 블록도이다.

도 3은 영상을 부호화 및 복호화할 때의 영상의 분할 구조를 개략적으로 나타내는 도면이다.

도 4는 화면 내 예측 과정의 실시예를 설명하기 위한 도면이다.

도 5는 화면 간 예측 과정의 실시예를 설명하기 위한 도면이다.

도 6은 변환 및 양자화의 과정을 설명하기 위한 도면이다.

도 7은 화면 내 예측에 이용 가능한 참조 샘플들을 설명하기 위한 도면이다.

도 8은 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 9는 luma 신호와 chroma 신호가 별도의 두 개의 트리 구조로 분할되는 예를 도시한 도면이다.

도 10은 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 11은 luma 신호와 chroma 신호가 소정 크기의 블록부터 별도의 두 개의 트리 구조로 분할되는 예를 도시한 도면이다.

도 12는 현재 CTU의 luma 신호와 chroma 신호가 소정 크기의 블록까지 공통의 트리 구조로 분할되고, 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되는 예를 도시한 도면이다.

도 13은 현재 CTU의 luma 신호와 chroma 신호가 별도의 두 개 트리 구조로 분할되는 예를 도시한 도면이다.

도 14는 현재 CTU의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

도 15는 현재 CTU의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

도 16은 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측되는 예를 도시한 도면이다.

도 17은 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되고, CTU의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

도 18은 CTU의 luma 신호와 chroma 신호가 별도의 두 개 트리 구조로 분할되고, CTU의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

도 19는 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, CTU의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

도 20은 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, CTU의 모든 leaf node(혹은 CU)는 화면내 예측되는 예를 도시한 도면이다.

도 21은 CTU의 luma 신호와 chroma 신호가 별도의 두 개의 트리 구조로 분할되고, CTU의 모든 leaf node(혹은 CU)는 화면내 예측되는 예를 도시한 도면이다.

도 22는 ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 23은 ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 24는 ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 25는 ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 26은 ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 27은 ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 28은 ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 29는 ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 30은 ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 31은 ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 32는 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 33은 현재 region의 luma 신호와 chroma 신호가 별도의 두 개 트리 구조로 분할되는 예를 도시한 도면이다.

도 34는 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

도 35는 현재 region의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

도 36은 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

도 37은 region_pred_mode_and_separate_coding_trees_flag가 제1값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 38은 region_pred_mode_and_separate_coding_trees_flag가 제2값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 39는 region_pred_mode_and_separate_coding_trees_flag가 제3값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 40은 region_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 41은 region_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

도 42는 ctu_pred_mode_flag가 제1값을 가질 때, 현재 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 43은 ctu_pred_mode_flag가 제3값을 가질 때, 현재 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 44는 region_pred_mode_flag가 제1값을 가질 때, 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 45는 region_pred_mode_flag가 제3값을 가질 때, 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

도 46은 본 개시에 따른 region을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0034] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 도면에서 유사한 참조부호는 여러 측면에 걸쳐서 동일하거나 유사한 기능을 지칭한다. 도면에서의 요소들의 형상 및 크기 등은 보다 명확한 설명을 위해 과장될 수 있다. 후술하는 예시적 실시예들에 대한 상세한 설명은, 특정 실시예를 예시로서 도시하는 첨부 도면을 참조한다. 이들 실시예는 당업자가 실시예를 실시할 수 있기에 충분하도록 상세히 설명된다. 다양한 실시예들은 서로 다르지만 상호 배타적일 필요는 없음이 이해되어야 한다. 예를 들어, 여기에 기재되어 있는 특정 형상, 구조 및 특성은 일 실시예에 관련하여 본 발명의 정신 및 범위를 벗어나지 않으면서 다른 실시예로 구현될 수 있다. 또한, 각각의 개시된 실시예 내의 개별 구성요소의 위치 또는 배치는 실시예의 정신 및 범위를 벗어나지 않으면서 변경될 수 있음이 이해되어야 한다. 따라서, 후술하는 상세한 설명은 한정적인 의미로서 취하려는 것이 아니며, 예시적 실시예들의 범위는, 적절하게 설명된다면, 그 청구항들이 주장하는 것과 균등한 모든 범위와 더불어 첨부된 청구항에 의해서만 한정된다.

[0035] 본 발명에서 제1, 제2 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.

[0036] 본 발명의 어떤 구성 요소가 다른 구성 요소에 “연결되어” 있다거나 “접속되어” 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있으나, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어"있다거나 "직접 접속되어"있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

[0037] 본 발명의 실시예에 나타나는 구성부들은 서로 다른 특징적인 기능들을 나타내기 위해 독립적으로 도시되는 것으로, 각 구성부들이 분리된 하드웨어나 하나의 소프트웨어 구성단위로 이루어짐을 의미하지 않는다. 즉, 각 구성부는 설명의 편의상 각각의 구성부로 나열하여 포함한 것으로 각 구성부 중 적어도 두 개의 구성부가 합쳐져 하나의 구성부로 이루어지거나, 하나의 구성부가 복수 개의 구성부로 나뉘어져 기능을 수행할 수 있고 이러한 각 구성부의 통합된 실시예 및 분리된 실시예도 본 발명의 본질에서 벗어나지 않는 한 본 발명의 권리범위에 포함된다.

[0038] 본 발명에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 발명에서, "포함

하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다. 즉, 본 발명에서 특정 구성을 “포함”한다고 기술하는 내용은 해당 구성 이외의 구성을 배제하는 것이 아니며, 추가적인 구성이 본 발명의 실시 또는 본 발명의 기술적 사상의 범위에 포함될 수 있음을 의미한다.

[0039] 본 발명의 일부의 구성 요소는 본 발명에서 본질적인 기능을 수행하는 필수적인 구성 요소는 아니고 단지 성능을 향상시키기 위한 선택적 구성 요소일 수 있다. 본 발명은 단지 성능 향상을 위해 사용되는 구성 요소를 제외한 본 발명의 본질을 구현하는데 필수적인 구성부만을 포함하여 구현될 수 있고, 단지 성능 향상을 위해 사용되는 선택적 구성 요소를 제외한 필수 구성 요소만을 포함한 구조도 본 발명의 권리범위에 포함된다.

[0040] 이하, 도면을 참조하여 본 발명의 실시 형태에 대하여 구체적으로 설명한다. 본 명세서의 실시예를 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 명세서의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략하고, 도면상의 동일한 구성요소에 대해서는 동일한 참조부호를 사용하고 동일한 구성요소에 대해서 중복된 설명은 생략한다.

[0041] 이하에서 영상은 동영상(video)을 구성하는 하나의 픽처(picture)를 의미할 수 있으며, 동영상 자체를 나타낼 수도 있다. 예를 들면, "영상의 부호화 및/또는 복호화"는 "동영상의 부호화 및/또는 복호화"를 의미할 수 있으며, "동영상을 구성하는 영상들 중 하나의 영상의 부호화 및/또는 복호화"를 의미할 수도 있다.

[0042] 이하에서, 용어들 "동영상" 및 "비디오"는 동일한 의미로 사용될 수 있으며, 서로 교체되어 사용될 수 있다.

[0043] 이하에서, 대상 영상은 부호화의 대상인 부호화 대상 영상 및/또는 복호화의 대상인 복호화 대상 영상일 수 있다. 또한, 대상 영상은 부호화 장치로 입력된 입력 영상일 수 있고, 복호화 장치로 입력된 입력 영상일 수 있다. 여기서, 대상 영상은 현재 영상과 동일한 의미를 가질 수 있다.

[0044] 이하에서, 용어들 "영상", "픽처", "프레임(frame)" 및 "스크린(screen)"은 동일한 의미로 사용될 수 있으며, 서로 교체되어 사용될 수 있다.

[0045] 이하에서, 대상 블록은 부호화의 대상인 부호화 대상 블록 및/또는 복호화의 대상인 복호화 대상 블록일 수 있다. 또한, 대상 블록은 현재 부호화 및/또는 복호화의 대상인 현재 블록일 수 있다. 예를 들면, 용어들 "대상 블록" 및 "현재 블록"은 동일한 의미로 사용될 수 있으며, 서로 교체되어 사용될 수 있다.

[0046] 이하에서, 용어들 "블록" 및 "유닛"은 동일한 의미로 사용될 수 있으며, 서로 교체되어 사용될 수 있다. 또는 "블록"은 특정한 유닛을 나타낼 수 있다.

[0047] 이하에서, 용어들 "영역(region)" 및 "세그먼트(segment)"는 서로 교체되어 사용될 수 있다.

[0048] 이하에서, 특정한 신호는 특정한 블록을 나타내는 신호일 수 있다. 예를 들면, 원(original) 신호는 대상 블록을 나타내는 신호일 수 있다. 예측(prediction) 신호는 예측 블록을 나타내는 신호일 수 있다. 잔여(residual) 신호는 잔여 블록(residual block)을 나타내는 신호일 수 있다.

[0049] 실시예들에서, 특정된 정보, 데이터, 플래그(flag), 색인(index) 및 요소(element), 속성(attribute) 등의 각각은 값을 가질 수 있다. 정보, 데이터, 플래그, 색인 및 요소, 속성 등의 값 "0"은 논리 거짓(logical false) 또는 제1 기정의된(predefined) 값을 나타낼 수 있다. 말하자면, 값 "0", 거짓, 논리 거짓 및 제1 기정의된 값은 서로 대체되어 사용될 수 있다. 정보, 데이터, 플래그, 색인 및 요소, 속성 등의 값 "1"은 논리 참(logical true) 또는 제2 기정의된 값을 나타낼 수 있다. 말하자면, 값 "1", 참, 논리 참 및 제2 기정의된 값은 서로 대체되어 사용될 수 있다.

[0050] 행, 열 또는 색인(index)을 나타내기 위해 i 또는 j 등의 변수가 사용될 때, i의 값은 0 이상의 정수일 수 있으며, 1 이상의 정수일 수도 있다. 말하자면, 실시예들에서 행, 열 및 색인 등은 0에서부터 카운트될 수 있으며, 1에서부터 카운트될 수 있다.

[0051] 용어 설명

[0052] 부호화기(Encoder): 부호화(Encoding)를 수행하는 장치를 의미한다. 즉, 부호화 장치를 의미할 수 있다.

[0053] 복호화기(Decoder): 복호화(Decoding)를 수행하는 장치를 의미한다. 즉, 복호화 장치를 의미할 수 있다.

[0054] 블록(Block): 샘플(Sample)의 MxN 배열이다. 여기서 M과 N은 양의 정수 값을 의미할 수 있으며, 블록은 흔히 2

차원 형태의 샘플 배열을 의미할 수 있다. 블록은 유닛을 의미할 수 있다. 현재 블록은 부호화 시 부호화의 대상이 되는 부호화 대상 블록, 복호화 시 복호화의 대상이 되는 복호화 대상 블록을 의미할 수 있다. 또한, 현재 블록은 부호화 블록, 예측 블록, 잔여 블록, 변환 블록 중 적어도 하나일 수 있다.

[0055] 샘플(Sample): 블록을 구성하는 기본 단위이다. 비트 깊이 (bit depth, B_d)에 따라 0부터 2^{Bd} - 1까지의 값으로 표현될 수 있다. 본 발명에서 샘플은 화소 또는 픽셀과 같은 의미로 사용될 수 있다. 즉, 샘플, 화소, 픽셀은 서로 같은 의미를 가질 수 있다.

[0056] 유닛(Unit): 영상 부호화 및 복호화의 단위를 의미할 수 있다. 영상의 부호화 및 복호화에 있어서, 유닛은 하나의 영상을 분할한 영역일 수 있다. 또한, 유닛은 하나의 영상을 세분화된 유닛으로 분할하여 부호화 혹은 복호화 할 때 그 분할된 단위를 의미할 수 있다. 즉, 하나의 영상은 복수의 유닛들로 분할될 수 있다. 영상의 부호화 및 복호화에 있어서, 유닛 별로 기정의된 처리가 수행될 수 있다. 하나의 유닛은 유닛에 비해 더 작은 크기를 갖는 하위 유닛으로 더 분할될 수 있다. 기능에 따라서, 유닛은 블록(Block), 매크로블록(Macroblock), 부호화 트리 유닛(Coding Tree Unit), 부호화 트리 블록(Coding Tree Block), 부호화 유닛(Coding Unit), 부호화 블록(Coding Block), 예측 유닛(Prediction Unit), 예측 블록(Prediction Block), 잔여 유닛(Residual Unit), 잔여 블록(Residual Block), 변환 유닛(Transform Unit), 변환 블록(Transform Block) 등을 의미할 수 있다. 또한, 유닛은 블록과 구분하여 지칭하기 위해 휘도(Luma) 성분 블록과 그에 대응하는 색차(Chroma) 성분 블록 그리고 각 블록에 대한 구문 요소를 포함한 것을 의미할 수 있다. 유닛은 다양한 크기와 형태를 가질 수 있으며, 특히 유닛의 형태는 정사각형뿐만 아니라 직사각형, 사다리꼴, 삼각형, 오각형 등 2차원으로 표현될 수 있는 기하학적 도형을 포함할 수 있다. 또한, 유닛 정보는 부호화 유닛, 예측 유닛, 잔여 유닛, 변환 유닛 등을 가리키는 유닛의 타입, 유닛의 크기, 유닛의 깊이, 유닛의 부호화 및 복호화 순서 등 중 적어도 하나 이상을 포함할 수 있다.

[0057] 부호화 트리 유닛(Coding Tree Unit): 하나의 휘도 성분(Y) 부호화 트리 블록과 관련된 두 색차 성분(Cb, Cr) 부호화 트리 블록들로 구성된다. 또한, 상기 블록들과 각 블록에 대한 구문 요소를 포함한 것을 의미할 수도 있다. 각 부호화 트리 유닛은 부호화 유닛, 예측 유닛, 변환 유닛 등의 하위 유닛을 구성하기 위하여 쿼드트리(quad tree), 이진트리(binary tree), 3분할트리(ternary tree) 등 하나 이상의 분할 방식을 이용하여 분할될 수 있다. 입력 영상의 분할처럼 영상의 복/부호화 과정에서 처리 단위가 되는 샘플 블록을 지칭하기 위한 용어로 사용될 수 있다. 여기서, 쿼드트리는 4분할트리(quarternary tree)를 의미할 수 있다.

[0058] 부호화 블록의 크기가 소정의 범위 내에 속하는 경우에는 쿼드트리로만 분할이 가능할 수 있다. 여기서, 소정의 범위는 쿼드트리만으로 분할이 가능한 부호화 블록의 최대 크기 및 최소 크기 중 적어도 하나로 정의될 수 있다. 쿼드트리 형태의 분할이 허용되는 부호화 블록의 최대/최소 크기를 나타내는 정보는 비트스트림을 통해 시그널링될 수 있고, 해당 정보는 시퀀스, 픽처 파라미터, 타일 그룹, 또는 슬라이스(세그먼트) 중 적어도 하나의 단위로 시그널링될 수 있다. 또는, 부호화 블록의 최대/최소 크기는 부호화기/복호화기에 기-설정된 고정된 크기일 수도 있다. 예를 들어, 부호화 블록의 크기가 256x256 내지 64x64에 해당하는 경우에는 쿼드트리로만 분할이 가능할 수 있다. 또는 부호화 블록의 크기가 최대 변환 블록의 크기 보다 큰 경우에는 쿼드트리로만 분할이 가능할 수 있다. 이때, 상기 분할되는 블록은 부호화 블록 또는 변환 블록 중 적어도 하나일 수 있다. 이러한 경우에 부호화 블록의 분할을 나타내는 정보(예컨대, split_flag)는 쿼드트리 분할 여부를 나타내는 플래그일 수 있다. 부호화 블록의 크기가 소정의 범위 내에 속하는 경우에는 이진트리 또는 3분할트리로만 분할이 가능할 수 있다. 이 경우, 쿼드트리에 관한 상기 설명은 이진트리 또는 3분할트리에 대해서도 동일하게 적용될 수 있다.

[0059] 부호화 트리 블록(Coding Tree Block): Y 부호화 트리 블록, Cb 부호화 트리 블록, Cr 부호화 트리 블록 중 어느 하나를 지칭하기 위한 용어로 사용될 수 있다.

[0060] 주변 블록(Neighbor block): 현재 블록에 인접한 블록을 의미할 수 있다. 현재 블록에 인접한 블록은 현재 블록에 경계가 맞닿은 블록 또는 현재 블록으로부터 소정의 거리 내에 위치한 블록을 의미할 수 있다. 주변 블록은 현재 블록의 꼭지점에 인접한 블록을 의미할 수 있다. 여기에서, 현재 블록의 꼭지점에 인접한 블록이란, 현재 블록에 가로로 인접한 이웃 블록에 세로로 인접한 블록 또는 현재 블록에 세로로 인접한 이웃 블록에 가로로 인접한 블록일 수 있다. 주변 블록은 복원된 주변 블록을 의미할 수도 있다.

[0061] 복원된 주변 블록(Reconstructed Neighbor Block): 현재 블록 주변에 공간적(Spatial)/시간적(Temporal)으로 이미 부호화 혹은 복호화된 주변 블록을 의미할 수 있다. 이때, 복원된 주변 블록은 복원된 주변 유닛을 의미할

수 있다. 복원된 공간적 주변 블록은 현재 픽처 내의 블록이면서 부호화 및/또는 복호화를 통해 이미 복원된 블록일 수 있다. 복원된 시간적 주변 블록은 참조 영상 내에서 현재 픽처의 현재 블록과 대응하는 위치의 복원된 블록 또는 그 주변 블록일 수 있다.

[0062] 유닛 깊이(Depth): 유닛이 분할된 정도를 의미할 수 있다. 트리 구조(Tree Structure)에서 가장 상위 노드(Root Node)는 분할되지 않은 최초의 유닛에 대응할 수 있다. 가장 상위 노드는 루트 노드로 칭해질 수 있다. 또한, 가장 상위 노드는 최소의 깊이 값을 가질 수 있다. 이 때, 가장 상위 노드는 레벨(Level) 0의 깊이를 가질 수 있다. 레벨 1의 깊이를 갖는 노드는 최초의 유닛이 한 번 분할됨에 따라 생성된 유닛을 나타낼 수 있다. 레벨 2의 깊이를 갖는 노드는 최초의 유닛이 두 번 분할됨에 따라 생성된 유닛을 나타낼 수 있다. 레벨 n의 깊이를 갖는 노드는 최초의 유닛이 n번 분할됨에 따라 생성된 유닛을 나타낼 수 있다. 리프 노드(Leaf Node)는 가장 하위의 노드일 수 있으며, 더 분할될 수 없는 노드일 수 있다. 리프 노드의 깊이는 최대 레벨일 수 있다. 예를 들면, 최대 레벨의 기정의된 값은 3일 수 있다. 루트 노드는 깊이가 가장 얕고, 리프 노드는 깊이가 가장 깊다고 할 수 있다. 또한, 유닛을 트리 구조로 표현했을 때 유닛이 존재하는 레벨이 유닛 깊이를 의미할 수 있다.

[0063] 비트스트림(Bitstream): 부호화된 영상 정보를 포함하는 비트의 열을 의미할 수 있다.

[0064] 파라미터 세트(Parameter Set): 비트스트림 내의 구조 중 헤더(header) 정보에 해당한다. 비디오 파라미터 세트(video parameter set), 시퀀스 파라미터 세트(sequence parameter set), 픽처 파라미터 세트(picture parameter set), 적응 파라미터 세트(adaptation parameter set) 중 적어도 하나가 파라미터 세트에 포함될 수 있다. 또한, 파라미터 세트는 타일 그룹, 슬라이스(slice) 헤더 및 타일(tile) 헤더 정보를 포함할 수도 있다. 또한, 상기 타일 그룹은 여러 타일을 포함하는 그룹을 의미할 수 있으며, 슬라이스와 동일한 의미일 수 있다.

[0065] 적응 파라미터 세트는 서로 다른 픽처, 서브픽처, 슬라이스, 타일 그룹, 타일, 혹은 브릭에서 참조하여 공유될 수 있는 파라미터 세트를 의미할 수 있다. 또한, 픽처 내 서브픽처, 슬라이스, 타일 그룹, 타일, 혹은 브릭에서는 서로 다른 적응 파라미터 세트를 참조하여, 적응 파라미터 세트 내 정보를 사용할 수 있다.

[0066] 또한, 적응 파라미터 세트는 픽처 내 서브픽처, 슬라이스, 타일 그룹, 타일, 혹은 브릭에서는 서로 다른 적응 파라미터 세트의 식별자를 사용하여 서로 다른 적응 파라미터 세트를 참조할 수 있다.

[0067] 또한, 적응 파라미터 세트는 서브픽처 내 슬라이스, 타일 그룹, 타일, 혹은 브릭에서는 서로 다른 적응 파라미터 세트의 식별자를 사용하여 서로 다른 적응 파라미터 세트를 참조할 수 있다.

[0068] 또한, 적응 파라미터 세트는 슬라이스 내 타일, 혹은 브릭에서는 서로 다른 적응 파라미터 세트의 식별자를 사용하여 서로 다른 적응 파라미터 세트를 참조할 수 있다.

[0069] 또한, 적응 파라미터 세트는 타일 내 브릭에서는 서로 다른 적응 파라미터 세트의 식별자를 사용하여 서로 다른 적응 파라미터 세트를 참조할 수 있다.

[0070] 상기 서브픽처의 파라미터 세트 혹은 헤더에 적응 파라미터 세트 식별자에 대한 정보를 포함하여, 해당 적응 파라미터 세트 식별자에 대응하는 적응 파라미터 세트를 서브픽처에서 사용할 수 있다.

[0071] 상기 타일의 파라미터 세트 혹은 헤더에 적응 파라미터 세트 식별자에 대한 정보를 포함하여, 해당 적응 파라미터 세트 식별자에 대응하는 적응 파라미터 세트를 타일에서 사용할 수 있다.

[0072] 상기 브릭의 헤더에 적응 파라미터 세트 식별자에 대한 정보를 포함하여, 해당 적응 파라미터 세트 식별자에 대응하는 적응 파라미터 세트를 브릭에서 사용할 수 있다.

[0073] 상기 픽처는 하나 이상의 타일 행과 하나 이상의 타일 열로 분할될 수 있다.

[0074] 상기 서브픽처는 픽처 내에서 하나 이상의 타일 행과 하나 이상의 타일 열로 분할될 수 있다. 상기 서브픽처는 픽처 내에서 직사각형/정사각형 형태를 가지는 영역이며, 하나 이상의 CTU를 포함할 수 있다. 또한, 하나의 서브픽처 내에는 적어도 하나 이상의 타일/브릭/슬라이스가 포함될 수 있다.

[0075] 상기 타일은 픽처 내에서 직사각형/정사각형 형태를 가지는 영역이며, 하나 이상의 CTU를 포함할 수 있다. 또한, 타일은 하나 이상의 브릭으로 분할될 수 있다.

[0076] 상기 브릭은 타일 내에서 하나 이상의 CTU 행을 의미할 수 있다. 타일은 하나 이상의 브릭으로 분할될 수 있고, 각 브릭은 적어도 하나 이상의 CTU 행을 가질 수 있다. 2개 이상으로 분할되지 않는 타일도 브릭을 의미할 수 있다.

[0077] 상기 슬라이스는 픽처 내에서 하나 이상의 타일을 포함할 수 있고, 타일 내 하나 이상의 브릭을 포함할 수 있다.

[0078] 파싱(Parsing): 비트스트림을 엔트로피 복호화하여 구문 요소(Syntax Element)의 값을 결정하는 것을 의미하거나, 엔트로피 복호화 자체를 의미할 수 있다.

[0079] 심볼(Symbol): 부호화/복호화 대상 유닛의 구문 요소, 부호화 파라미터(coding parameter), 변환 계수(Transform Coefficient)의 값 등 중 적어도 하나를 의미할 수 있다. 또한, 심볼은 엔트로피 부호화의 대상 혹은 엔트로피 복호화의 결과를 의미할 수 있다.

[0080] 예측 모드(Prediction Mode): 화면 내 예측으로 부호화/복호화되는 모드 또는 화면 간 예측으로 부호화/복호화되는 모드를 지시하는 정보일 수 있다.

[0081] 예측 유닛(Prediction Unit): 화면 간 예측, 화면 내 예측, 화면 간 보상, 화면 내 보상, 움직임 보상 등 예측을 수행할 때의 기본 단위를 의미할 수 있다. 하나의 예측 유닛은 더 작은 크기를 가지는 복수의 파티션(Partition) 또는 복수의 하위 예측 유닛들로 분할 될 수도 있다. 복수의 파티션들 또한 예측 또는 보상의 수행에 있어서의 기본 단위일 수 있다. 예측 유닛의 분할에 의해 생성된 파티션 또한 예측 유닛일 수 있다.

[0082] 예측 유닛 파티션(Prediction Unit Partition): 예측 유닛이 분할된 형태를 의미할 수 있다.

[0083] 참조 영상 리스트(Reference Picture List): 화면 간 예측 혹은 움직임 보상에 사용되는 하나 이상의 참조 영상들을 포함하는 리스트를 의미할 수 있다. 참조 영상 리스트의 종류는 LC (List Combined), L0 (List 0), L1 (List 1), L2 (List 2), L3 (List 3) 등이 있을 수 있으며, 화면 간 예측에는 1개 이상의 참조 영상 리스트들이 사용될 수 있다.

[0084] 화면 간 예측 지시자(Inter Prediction Indicator): 현재 블록의 화면 간 예측 방향(단방향 예측, 쌍방향 예측 등)을 의미할 수 있다. 또는, 현재 블록의 예측 블록을 생성할 때 사용되는 참조 영상의 개수를 의미할 수 있다. 또는, 현재 블록에 대해 화면 간 예측 혹은 움직임 보상을 수행할 때 사용되는 예측 블록의 개수를 의미할 수 있다.

[0085] 예측 리스트 활용 플래그(prediction list utilization flag): 특정 참조 영상 리스트 내 적어도 하나의 참조 영상을 이용하여 예측 블록을 생성하는지 여부를 나타낸다. 예측 리스트 활용 플래그를 이용하여 화면 간 예측 지시자를 도출할 수 있고, 반대로 화면 간 예측 지시자를 이용하여 예측 리스트 활용 플래그를 도출할 수 있다. 예를 들어, 예측 리스트 활용 플래그가 제1 값인 0을 지시하는 경우, 해당 참조 영상 리스트 내 참조 영상을 이용하여 예측 블록을 생성하지 않는 것을 나타낼 수 있고, 제2 값인 1을 지시하는 경우, 해당 참조 영상 리스트를 이용하여 예측 블록을 생성할 수 있는 것을 나타낼 수 있다.

[0086] 참조 영상 색인(Reference Picture Index): 참조 영상 리스트에서 특정 참조 영상을 지시하는 색인을 의미할 수 있다.

[0087] 참조 영상(Reference Picture): 화면 간 예측 혹은 움직임 보상을 위해서 특정 블록이 참조하는 영상을 의미할 수 있다. 또는, 참조 영상은 화면 간 예측 또는 움직임 보상을 위해 현재 블록이 참조하는 참조 블록을 포함하는 영상일 수 있다. 이하, 용어 "참조 픽처" 및 "참조 영상"은 동일한 의미로 사용될 수 있으며, 서로 교체되어 사용될 수 있다.

[0088] 움직임 벡터(Motion Vector): 화면 간 예측 혹은 움직임 보상에 사용되는 2차원 벡터일 수 있다. 움직임 벡터는 부호화/복호화 대상 블록과 참조 블록 사이의 오프셋을 의미할 수 있다. 예를 들어, (mvX, mvY)는 움직임 벡터를 나타낼 수 있다. mvX는 수평(horizontal) 성분, mvY는 수직(vertical) 성분을 나타낼 수 있다.

[0089] 탐색 영역(Search Range): 탐색 영역은 화면 간 예측 중 움직임 벡터에 대한 탐색이 이루어지는 2차원의 영역일 수 있다. 예를 들면, 탐색 영역의 크기는 MxN일 수 있다. M 및 N은 각각 양의 정수일 수 있다.

[0090] 움직임 벡터 후보(Motion Vector Candidate): 움직임 벡터를 예측할 때 예측 후보가 되는 블록 혹은 그 블록의 움직임 벡터를 의미할 수 있다. 또한, 움직임 벡터 후보는 움직임 벡터 후보 리스트에 포함될 수 있다.

[0091] 움직임 벡터 후보 리스트(Motion Vector Candidate List): 하나 이상의 움직임 벡터 후보들을 이용하여 구성된 리스트를 의미할 수 있다.

[0092] 움직임 벡터 후보 색인(Motion Vector Candidate Index): 움직임 벡터 후보 리스트 내의 움직임 벡터 후보를 가

리키는 지시자를 의미할 수 있다. 움직임 벡터 예측기(Motion Vector Predictor)의 색인(index)일 수 있다.

[0093] 움직임 정보(Motion Information): 움직임 벡터, 참조 영상 색인, 화면 간 예측 지시자 뿐만 아니라 예측 리스트 활용 플래그, 참조 영상 리스트 정보, 참조 영상, 움직임 벡터 후보, 움직임 벡터 후보 색인, 머지 후보, 머지 색인 등 중 적어도 하나를 포함하는 정보를 의미할 수 있다.

[0094] 머지 후보 리스트(Merge Candidate List): 하나 이상의 머지 후보들을 이용하여 구성된 리스트를 의미할 수 있다.

[0095] 머지 후보(Merge Candidate): 공간적 머지 후보, 시간적 머지 후보, 조합된 머지 후보, 조합 양예측 머지 후보, 제로 머지 후보 등을 의미할 수 있다. 머지 후보는 화면 간 예측 지시자, 각 리스트에 대한 참조 영상 색인, 움직임 벡터, 예측 리스트 활용 플래그, 화면 간 예측 지시자 등의 움직임 정보를 포함할 수 있다.

[0096] 머지 색인(Merge Index): 머지 후보 리스트 내 머지 후보를 가리키는 지시자를 의미할 수 있다. 또한, 머지 색인은 공간적/시간적으로 현재 블록과 인접하게 복원된 블록들 중 머지 후보를 유도한 블록을 지시할 수 있다. 또한, 머지 색인은 머지 후보가 가지는 움직임 정보 중 적어도 하나를 지시할 수 있다.

[0097] 변환 유닛(Transform Unit): 변환, 역변환, 양자화, 역양자화, 변환 계수 부호화/복호화와 같이 잔여 신호(residual signal) 부호화/복호화를 수행할 때의 기본 단위를 의미할 수 있다. 하나의 변환 유닛은 분할되어 더 작은 크기를 가지는 복수의 하위 변환 유닛들로 분할될 수 있다. 여기서, 변환/역변환은 1차 변환/역변환 및 2차 변환/역변환 중 적어도 하나 이상을 포함할 수 있다.

[0098] 스케일링(Scaling): 양자화된 레벨에 인수를 곱하는 과정을 의미할 수 있다. 양자화된 레벨에 대한 스케일링의 결과로 변환 계수를 생성할 수 있다. 스케일링을 역양자화(dequantization)라고도 부를 수 있다.

[0099] 양자화 매개변수(Quantization Parameter): 양자화에서 변환 계수를 이용하여 양자화된 레벨(quantized level)을 생성할 때 사용하는 값을 의미할 수 있다. 또는, 역양자화에서 양자화된 레벨을 스케일링하여 변환 계수를 생성할 때 사용하는 값을 의미할 수도 있다. 양자화 매개변수는 양자화 스텝 크기(step size)에 매핑된 값일 수 있다.

[0100] 잔여 양자화 매개변수(Delta Quantization Parameter): 예측된 양자화 매개변수와 부호화/복호화 대상 유닛의 양자화 매개변수의 차분(difference) 값을 의미할 수 있다.

[0101] 스캔(Scan): 유닛, 블록 혹은 행렬 내 계수의 순서를 정렬하는 방법을 의미할 수 있다. 예를 들어, 2차원 배열을 1차원 배열 형태로 정렬하는 것을 스캔이라고 한다. 또는, 1차원 배열을 2차원 배열 형태로 정렬하는 것도 스캔 혹은 역 스캔(Inverse Scan)이라고 부를 수 있다.

[0102] 변환 계수(Transform Coefficient): 부호화기에서 변환을 수행하고 나서 생성된 계수 값을 의미할 수 있다. 또는, 복호화기에서 엔트로피 복호화 및 역양자화 중 적어도 하나를 수행하고 나서 생성된 계수 값을 의미할 수도 있다. 변환 계수 또는 잔여 신호에 양자화를 적용한 양자화된 레벨 또는 양자화된 변환 계수 레벨도 변환 계수의 의미에 포함될 수 있다.

[0103] 양자화된 레벨(Quantized Level): 부호화기에서 변환 계수 또는 잔여 신호에 양자화를 수행하여 생성된 값을 의미할 수 있다. 또는, 복호화기에서 역양자화를 수행하기 전 역양자화의 대상이 되는 값을 의미할 수도 있다. 유사하게, 변환 및 양자화의 결과인 양자화된 변환 계수 레벨도 양자화된 레벨의 의미에 포함될 수 있다.

[0104] 넌제로 변환 계수(Non-zero Transform Coefficient): 값의 크기가 0이 아닌 변환 계수 혹은 값의 크기가 0이 아닌 변환 계수 레벨 혹은 양자화된 레벨을 의미할 수 있다.

[0105] 양자화 행렬(Quantization Matrix): 영상의 주관적 화질 혹은 객관적 화질을 향상시키기 위해서 양자화 혹은 역양자화 과정에서 이용하는 행렬을 의미할 수 있다. 양자화 행렬을 스케일링 리스트(scaling list)라고도 부를 수 있다.

[0106] 양자화 행렬 계수(Quantization Matrix Coefficient): 양자화 행렬 내의 각 원소(element)를 의미할 수 있다. 양자화 행렬 계수를 행렬 계수(matrix coefficient)라고도 할 수 있다.

[0107] 기본 행렬(Default Matrix): 부호화기와 복호화기에서 미리 정의되어 있는 소정의 양자화 행렬을 의미할 수 있다.

[0108] 비 기본 행렬(Non-default Matrix): 부호화기와 복호화기에서 미리 정의되지 않고, 사용자에 의해서 시그널링되

는 양자화 행렬을 의미할 수 있다.

[0109] 통계값(statistic value): 연산 가능한 특정 값들을 가지는 변수, 부호화 파라미터, 상수 등 적어도 하나에 대한 통계값은 해당 특정 값들의 평균값, 가중평균값, 가중합값, 최소값, 최대값, 최빈값, 중간값, 보간값 중 적어도 하나 이상일 수 있다.

[0110] 도 1은 본 발명이 적용되는 부호화 장치의 일 실시예에 따른 구성을 나타내는 블록도이다.

[0111] 부호화 장치(100)는 인코더, 비디오 부호화 장치 또는 영상 부호화 장치일 수 있다. 비디오는 하나 이상의 영상들을 포함할 수 있다. 부호화 장치(100)는 하나 이상의 영상들을 순차적으로 부호화할 수 있다.

[0112] 도 1을 참조하면, 부호화 장치(100)는 움직임 예측부(111), 움직임 보상부(112), 인트라 예측부(120), 스위치(115), 감산기(125), 변환부(130), 양자화부(140), 엔트로피 부호화부(150), 역양자화부(160), 역변환부(170), 가산기(175), 필터부(180) 및 참조 픽처 버퍼(190)를 포함할 수 있다.

[0113] 부호화 장치(100)는 입력 영상에 대해 인트라 모드 및/또는 인터 모드로 부호화를 수행할 수 있다. 또한, 부호화 장치(100)는 입력 영상에 대한 부호화를 통해 부호화된 정보를 포함하는 비트스트림을 생성할 수 있고, 생성된 비트스트림을 출력할 수 있다. 생성된 비트스트림은 컴퓨터 판독가능한 기록 매체에 저장될 수 있거나, 유/무선 전송 매체를 통해 스트리밍될 수 있다. 예측 모드로 인트라 모드가 사용되는 경우 스위치(115)는 인트라로 전환될 수 있고, 예측 모드로 인터 모드가 사용되는 경우 스위치(115)는 인터로 전환될 수 있다. 여기서 인트라 모드는 화면 내 예측 모드를 의미할 수 있으며, 인터 모드는 화면 간 예측 모드를 의미할 수 있다. 부호화 장치(100)는 입력 영상의 입력 블록에 대한 예측 블록을 생성할 수 있다. 또한, 부호화 장치(100)는 예측 블록이 생성된 후, 입력 블록 및 예측 블록의 차분(residual)을 사용하여 잔여 블록을 부호화할 수 있다. 입력 영상은 현재 부호화의 대상인 현재 영상으로 칭해질 수 있다. 입력 블록은 현재 부호화의 대상인 현재 블록 혹은 부호화 대상 블록으로 칭해질 수 있다.

[0114] 예측 모드가 인트라 모드인 경우, 인트라 예측부(120)는 현재 블록의 주변에 이미 부호화/복호화된 블록의 샘플을 참조 샘플로서 이용할 수 있다. 인트라 예측부(120)는 참조 샘플을 이용하여 현재 블록에 대한 공간적 예측을 수행할 수 있고, 공간적 예측을 통해 입력 블록에 대한 예측 샘플들을 생성할 수 있다. 여기서 인트라 예측은 화면 내 예측을 의미할 수 있다.

[0115] 예측 모드가 인터 모드인 경우, 움직임 예측부(111)는, 움직임 예측 과정에서 참조 영상으로부터 입력 블록과 가장 매치가 잘 되는 영역을 검색할 수 있고, 검색된 영역을 이용하여 움직임 벡터를 도출할 수 있다. 이때, 상기 영역으로 탐색 영역을 사용할 수 있다. 참조 영상은 참조 픽처 버퍼(190)에 저장될 수 있다. 여기서, 참조 영상에 대한 부호화/복호화가 처리되었을 때 참조 픽처 버퍼(190)에 저장될 수 있다.

[0116] 움직임 보상부(112)는 움직임 벡터를 이용하는 움직임 보상을 수행함으로써 현재 블록에 대한 예측 블록을 생성할 수 있다. 여기서 인터 예측은 화면 간 예측 혹은 움직임 보상을 의미할 수 있다.

[0117] 상기 움직임 예측부(111)과 움직임 보상부(112)는 움직임 벡터의 값이 정수 값을 가지지 않을 경우에 참조 영상 내의 일부 영역에 대해 보간 필터(Interpolation Filter)를 적용하여 예측 블록을 생성할 수 있다. 화면 간 예측 혹은 움직임 보상을 수행하기 위해 부호화 유닛을 기준으로 해당 부호화 유닛에 포함된 예측 유닛의 움직임 예측 및 움직임 보상 방법이 스킵 모드(Skip Mode), 머지 모드(Merge Mode), 향상된 움직임 벡터 예측(Advanced Motion Vector Prediction; AMVP) 모드, 현재 픽처 참조 모드 중 어떠한 방법인지 여부를 판단할 수 있고, 각 모드에 따라 화면 간 예측 혹은 움직임 보상을 수행할 수 있다.

[0118] 감산기(125)는 입력 블록 및 예측 블록의 차분을 사용하여 잔여 블록을 생성할 수 있다. 잔여 블록은 잔여 신호로 칭해질 수도 있다. 잔여 신호는 원 신호 및 예측 신호 간의 차이(difference)를 의미할 수 있다. 또는, 잔여 신호는 원신호 및 예측 신호 간의 차이를 변환(transform)하거나, 양자화하거나, 또는 변환 및 양자화함으로써 생성된 신호일 수 있다. 잔여 블록은 블록 단위의 잔여 신호일 수 있다.

[0119] 변환부(130)는 잔여 블록에 대해 변환(transform)을 수행하여 변환 계수(transform coefficient)를 생성할 수 있고, 생성된 변환 계수를 출력할 수 있다. 여기서, 변환 계수는 잔여 블록에 대한 변환을 수행함으로써 생성된 계수 값일 수 있다. 변환 생략(transform skip) 모드가 적용되는 경우, 변환부(130)는 잔여 블록에 대한 변환을 생략할 수도 있다.

[0120] 변환 계수 또는 잔여 신호에 양자화를 적용함으로써 양자화된 레벨(quantized level)이 생성될 수 있다. 이하,

실시예들에서는 양자화된 레벨도 변환 계수로 칭해질 수 있다.

[0121] 양자화부(140)는 변환 계수 또는 잔여 신호를 양자화 매개변수에 따라 양자화함으로써 양자화된 레벨을 생성할 수 있고, 생성된 양자화된 레벨을 출력할 수 있다. 이때, 양자화부(140)에서는 양자화 행렬을 사용하여 변환 계수를 양자화할 수 있다.

[0122] 엔트로피 부호화부(150)는, 양자화부(140)에서 산출된 값들 또는 부호화 과정에서 산출된 부호화 파라미터(Coding Parameter) 값들 등에 대하여 확률 분포에 따른 엔트로피 부호화를 수행함으로써 비트스트림(bitstream)을 생성할 수 있고, 비트스트림을 출력할 수 있다. 엔트로피 부호화부(150)는 영상의 샘플에 관한 정보 및 영상의 복호화를 위한 정보에 대한 엔트로피 부호화를 수행할 수 있다. 예를 들면, 영상의 복호화를 위한 정보는 구문 요소(syntax element) 등을 포함할 수 있다.

[0123] 엔트로피 부호화가 적용되는 경우, 높은 발생 확률을 갖는 심볼(symbol)에 적은 수의 비트가 할당되고 낮은 발생 확률을 갖는 심볼에 많은 수의 비트가 할당되어 심볼이 표현됨으로써, 부호화 대상 심볼들에 대한 비트열의 크기가 감소될 수 있다. 엔트로피 부호화부(150)는 엔트로피 부호화를 위해 지수 골롬(exponential Golomb), CAVLC(Context-Adaptive Variable Length Coding), CABAC(Context-Adaptive Binary Arithmetic Coding)과 같은 부호화 방법을 사용할 수 있다. 예를 들면, 엔트로피 부호화부(150)는 가변 길이 부호화(Variable Length Coding/Code; VLC) 테이블을 이용하여 엔트로피 부호화를 수행할 수 있다. 또한 엔트로피 부호화부(150)는 대상 심볼의 이진화(binarization) 방법 및 대상 심볼/빈(bin)의 확률 모델(probability model)을 도출한 후, 도출된 이진화 방법, 확률 모델, 문맥 모델(Context Model)을 사용하여 산술 부호화를 수행할 수도 있다.

[0124] 엔트로피 부호화부(150)는 변환 계수 레벨(양자화된 레벨)을 부호화하기 위해 변환 계수 스캐닝(Transform Coefficient Scanning) 방법을 통해 2차원의 블록 형태(form) 계수를 1차원의 벡터 형태로 변경할 수 있다.

[0125] 부호화 파라미터(Coding Parameter)는 구문 요소와 같이 부호화기에서 부호화되어 복호화기로 시그널링되는 정보(플래그, 색인 등)뿐만 아니라, 부호화 과정 혹은 복호화 과정에서 유도되는 정보를 포함할 수 있으며, 영상을 부호화하거나 복호화할 때 필요한 정보를 의미할 수 있다. 예를 들어, 유닛/블록 크기, 유닛/블록 깊이, 유닛/블록 분할 정보, 유닛/블록 형태, 유닛/블록 분할 구조, 쿼드트리 형태의 분할 여부, 이진트리 형태의 분할 여부, 이진트리 형태의 분할 방향(가로 방향 혹은 세로 방향), 이진트리 형태의 분할 형태(대칭 분할 혹은 비대칭 분할), 3분할트리 형태의 분할 여부, 3분할트리 형태의 분할 방향(가로 방향 혹은 세로 방향), 3분할트리 형태의 분할 형태(대칭 분할 혹은 비대칭 분할), 복합형트리 형태의 분할 여부, 복합형트리 형태의 분할 방향(가로 방향 혹은 세로 방향), 복합형트리 형태의 분할 형태(대칭 분할 혹은 비대칭 분할), 복합형트리 형태의 분할 트리(이진트리 혹은 3분할 트리), 예측 모드(화면 내 예측 또는 화면 간 예측), 화면 내 휘도 예측 모드/방향, 화면 내 색차 예측 모드/방향, 화면 내 분할 정보, 화면 간 분할 정보, 부호화 블록 분할 플래그, 예측 블록 분할 플래그, 변환 블록 분할 플래그, 참조 샘플 필터링 방법, 참조 샘플 필터 탭, 참조 샘플 필터 계수, 예측 블록 필터링 방법, 예측 블록 필터 탭, 예측 블록 필터 계수, 예측 블록 경계 필터링 방법, 예측 블록 경계 필터 탭, 예측 블록 경계 필터 계수, 화면 내 예측 모드, 화면 간 예측 모드, 움직임 정보, 움직임 벡터, 움직임 벡터 차분, 참조 영상 색인, 화면 간 예측 방향, 화면 간 예측 지시자, 예측 리스트 활용 플래그, 참조 영상 리스트, 참조 영상, 움직임 벡터 예측 색인, 움직임 벡터 예측 후보, 움직임 벡터 후보 리스트, 머지 모드 사용 여부, 머지 색인, 머지 후보, 머지 후보 리스트, 스킵(skip) 모드 사용 여부, 보간 필터 종류, 보간 필터 탭, 보간 필터 계수, 움직임 벡터 크기, 움직임 벡터 표현 정확도, 변환 종류, 변환 크기, 1차 변환 사용 여부 정보, 2차 변환 사용 여부 정보, 1차 변환 색인, 2차 변환 색인, 잔여 신호 유무 정보, 부호화 블록 패턴(Coded Block Pattern), 부호화 블록 플래그(Coded Block Flag), 양자화 매개변수, 잔여 양자화 매개변수, 양자화 행렬, 화면 내 루프 필터 적용 여부, 화면 내 루프 필터 계수, 화면 내 루프 필터 탭, 화면 내 루프 필터 모양/형태, 디블록킹 필터 적용 여부, 디블록킹 필터 계수, 디블록킹 필터 탭, 디블록킹 필터 강도, 디블록킹 필터 모양/형태, 적응적 샘플 오프셋 적용 여부, 적응적 샘플 오프셋 값, 적응적 샘플 오프셋 카테고리, 적응적 샘플 오프셋 종류, 적응적 루프 필터 적용 여부, 적응적 루프 필터 계수, 적응적 루프 필터 탭, 적응적 루프 필터 모양/형태, 이진화/역이진화 방법, 문맥 모델 결정 방법, 문맥 모델 업데이트 방법, 레귤러 모드 수행 여부, 바이패스 모드 수행 여부, 문맥 빈, 바이패스 빈, 중요 계수 플래그, 마지막 중요 계수 플래그, 계수 그룹 단위 부호화 플래그, 마지막 중요 계수 위치, 계수 값이 1보다 큰지에 대한 플래그, 계수 값이 2보다 큰지에 대한 플래그, 계수 값이 3보다 큰지에 대한 플래그, 나머지 계수 값 정보, 부호(sign) 정보, 복원된 휘도 샘플, 복원된 색차 샘플, 잔여 휘도 샘플, 잔여 색차 샘플, 휘도 변환 계수, 색차 변환 계수, 휘도 양자화된 레벨, 색차 양자화된 레벨, 변환 계수 레벨 스캐닝 방법, 복호화기 측면 움직임 벡터 탐색 영역의 크기, 복호화기 측면 움직임 벡터 탐색 영역의 형태, 복호화기 측면 움직임 벡터 탐색 횟수, CTU 크기 정보, 최소 블록 크기 정보, 최대 블록 크

기 정보, 최대 블록 깊이 정보, 최소 블록 깊이 정보, 영상 디스플레이/출력 순서, 슬라이스 식별 정보, 슬라이스 타입, 슬라이스 분할 정보, 타일 그룹 식별 정보, 타일 그룹 타입, 타일 그룹 분할 정보, 타일 식별 정보, 타일 타입, 타일 분할 정보, 픽처 타입, 입력 샘플 비트 심도, 복원 샘플 비트 심도, 잔여 샘플 비트 심도, 변환 계수 비트 심도, 양자화된 레벨 비트 심도, 휘도 신호에 대한 정보, 색차 신호에 대한 정보 중 적어도 하나의 값 또는 조합된 형태가 부호화 파라미터에 포함될 수 있다.

[0126] 여기서, 플래그 혹은 색인을 시그널링(signaling)한다는 것은 인코더에서는 해당 플래그 혹은 색인을 엔트로피 부호화(Entropy Encoding)하여 비트스트림(Bitstream)에 포함하는 것을 의미할 수 있고, 디코더에서는 비트스트림으로부터 해당 플래그 혹은 색인을 엔트로피 복호화(Entropy Decoding)하는 것을 의미할 수 있다.

[0127] 부호화 장치(100)가 인터 예측을 통한 부호화를 수행할 경우, 부호화된 현재 영상은 이후에 처리되는 다른 영상에 대한 참조 영상으로서 사용될 수 있다. 따라서, 부호화 장치(100)는 부호화된 현재 영상을 다시 복원 또는 복호화할 수 있고, 복원 또는 복호화된 영상을 참조 영상으로 참조 픽처 버퍼(190)에 저장할 수 있다.

[0128] 양자화된 레벨은 역양자화부(160)에서 역양자화(dequantization)될 수 있고. 역변환부(170)에서 역변환(inverse transform)될 수 있다. 역양자화 및/또는 역변환된 계수는 가산기(175)를 통해 예측 블록과 합해질 수 있다, 역양자화 및/또는 역변환된 계수와 예측 블록을 합함으로써 복원 블록(reconstructed block)이 생성될 수 있다. 여기서, 역양자화 및/또는 역변환된 계수는 역양자화 및 역변환 중 적어도 하나 이상이 수행된 계수를 의미하며, 복원된 잔여 블록을 의미할 수 있다.

[0129] 복원 블록은 필터부(180)를 거칠 수 있다. 필터부(180)는 디블록킹 필터(deblocking filter), 샘플 적응적 오프셋(Sample Adaptive Offset; SAO), 적응적 루프 필터(Adaptive Loop Filter; ALF) 등 적어도 하나를 복원 샘플, 복원 블록 또는 복원 영상에 적용할 수 있다. 필터부(180)는 루프내 필터(in-loop filter)로 칭해질 수도 있다.

[0130] 디블록킹 필터는 블록들 간의 경계에서 발생한 블록 왜곡을 제거할 수 있다. 디블록킹 필터를 수행할지 여부를 판단하기 위해 블록에 포함된 몇 개의 열 또는 행에 포함된 샘플을 기초로 현재 블록에 디블록킹 필터 적용할지 여부를 판단할 수 있다. 블록에 디블록킹 필터를 적용하는 경우 필요한 디블록킹 필터링 강도에 따라 서로 다른 필터를 적용할 수 있다.

[0131] 샘플 적응적 오프셋을 이용하여 부호화 에러를 보상하기 위해 샘플 값에 적정 오프셋(offset) 값을 더할 수 있다. 샘플 적응적 오프셋은 디블록킹을 수행한 영상에 대해 샘플 단위로 원본 영상과의 오프셋을 보정할 수 있다. 영상에 포함된 샘플을 일정한 수의 영역으로 구분한 후 오프셋을 수행할 영역을 결정하고 해당 영역에 오프셋을 적용하는 방법 또는 각 샘플의 에지 정보를 고려하여 오프셋을 적용하는 방법을 사용할 수 있다.

[0132] 적응적 루프 필터는 복원 영상 및 원래의 영상을 비교한 값에 기반하여 필터링을 수행할 수 있다. 영상에 포함된 샘플을 소정의 그룹으로 나눈 후 해당 그룹에 적용될 필터를 결정하여 그룹마다 차별적으로 필터링을 수행할 수 있다. 적응적 루프 필터를 적용할지 여부에 관련된 정보는 부호화 유닛(Coding Unit, CU) 별로 시그널링될 수 있고, 각각의 블록에 따라 적용될 적응적 루프 필터의 모양 및 필터 계수는 달라질 수 있다.

[0133] 필터부(180)를 거친 복원 블록 또는 복원 영상은 참조 픽처 버퍼(190)에 저장될 수 있다. 필터부(180)를 거친 복원 블록은 참조 영상의 일부일 수 있다. 말하자면, 참조 영상은 필터부(180)를 거친 복원 블록들로 구성된 복원 영상일 수 있다. 저장된 참조 영상은 이후 화면 간 예측 혹은 움직임 보상에 사용될 수 있다.

[0134] 도 2는 본 발명이 적용되는 복호화 장치의 일 실시예에 따른 구성을 나타내는 블록도이다.

[0135] 복호화 장치(200)는 디코더, 비디오 복호화 장치 또는 영상 복호화 장치일 수 있다.

[0136] 도 2를 참조하면, 복호화 장치(200)는 엔트로피 복호화부(210), 역양자화부(220), 역변환부(230), 인트라 예측부(240), 움직임 보상부(250), 가산기(255), 필터부(260) 및 참조 픽처 버퍼(270)를 포함할 수 있다.

[0137] 복호화 장치(200)는 부호화 장치(100)에서 출력된 비트스트림을 수신할 수 있다. 복호화 장치(200)는 컴퓨터 판독가능한 기록 매체에 저장된 비트스트림을 수신하거나, 유/무선 전송 매체를 통해 스트리밍되는 비트스트림을 수신할 수 있다. 복호화 장치(200)는 비트스트림에 대하여 인트라 모드 또는 인터 모드로 복호화를 수행할 수 있다. 또한, 복호화 장치(200)는 복호화를 통해 복원된 영상 또는 복호화된 영상을 생성할 수 있고, 복원된 영상 또는 복호화된 영상을 출력할 수 있다.

[0138] 복호화에 사용되는 예측 모드가 인트라 모드인 경우 스위치가 인트라로 전환될 수 있다. 복호화에 사용되는 예

측 모드가 인터 모드인 경우 스위치가 인터로 전환될 수 있다.

[0139] 복호화 장치(200)는 입력된 비트스트림을 복호화하여 복원된 잔여 블록(reconstructed residual block)을 획득할 수 있고, 예측 블록을 생성할 수 있다. 복원된 잔여 블록 및 예측 블록이 획득되면, 복호화 장치(200)는 복원된 잔여 블록과 및 예측 블록을 더함으로써 복호화 대상이 되는 복원 블록을 생성할 수 있다. 복호화 대상 블록은 현재 블록으로 칭해질 수 있다.

[0140] 엔트로피 복호화부(210)는 비트스트림에 대한 확률 분포에 따른 엔트로피 복호화를 수행함으로써 심볼들을 생성할 수 있다. 생성된 심볼들은 양자화된 레벨 형태의 심볼을 포함할 수 있다. 여기에서, 엔트로피 복호화 방법은 상술된 엔트로피 부호화 방법의 역과정일 수 있다.

[0141] 엔트로피 복호화부(210)는 변환 계수 레벨(양자화된 레벨)을 복호화하기 위해 변환 계수 스캐닝 방법을 통해 1차원의 벡터 형태 계수를 2차원의 블록 형태로 변경할 수 있다.

[0142] 양자화된 레벨은 역양자화부(220)에서 역양자화될 수 있고, 역변환부(230)에서 역변환될 수 있다. 양자화된 레벨은 역양자화 및/또는 역변환이 수행된 결과로서, 복원된 잔여 블록으로 생성될 수 있다. 이때, 역양자화부(220)는 양자화된 레벨에 양자화 행렬을 적용할 수 있다.

[0143] 인트라 모드가 사용되는 경우, 인트라 예측부(240)는 복호화 대상 블록 주변의 이미 복호화된 블록의 샘플 값을 이용하는 공간적 예측을 현재 블록에 대해 수행함으로써 예측 블록을 생성할 수 있다.

[0144] 인터 모드가 사용되는 경우, 움직임 보상부(250)는 움직임 벡터 및 참조 픽처 버퍼(270)에 저장되어 있는 참조 영상을 이용하는 움직임 보상을 현재 블록에 대해 수행함으로써 예측 블록을 생성할 수 있다. 상기 움직임 보상부(250)는 움직임 벡터의 값이 정수 값을 가지지 않을 경우에 참조 영상 내의 일부 영역에 대해 보간 필터를 적용하여 예측 블록을 생성할 수 있다. 움직임 보상을 수행하기 위해 부호화 유닛을 기준으로 해당 부호화 유닛에 포함된 예측 유닛의 움직임 보상 방법이 스킵 모드, 머지 모드, AMVP 모드, 현재 픽처 참조 모드 중 어떠한 방법인지 여부를 판단할 수 있고, 각 모드에 따라 움직임 보상을 수행할 수 있다.

[0145] 가산기(255)는 복원된 잔여 블록 및 예측 블록을 가산하여 복원 블록을 생성할 수 있다. 필터부(260)는 디블록킹 필터, 샘플 적응적 오프셋 및 적응적 루프 필터 등 적어도 하나를 복원 블록 또는 복원 영상에 적용할 수 있다. 필터부(260)는 복원 영상을 출력할 수 있다. 복원 블록 또는 복원 영상은 참조 픽처 버퍼(270)에 저장되어 인터 예측에 사용될 수 있다. 필터부(260)를 거친 복원 블록은 참조 영상의 일부일 수 있다. 말하자면, 참조 영상은 필터부(260)를 거친 복원 블록들로 구성된 복원 영상일 수 있다. 저장된 참조 영상은 이후 화면 간 예측 혹은 움직임 보상에 사용될 수 있다.

[0146] 도 3은 영상을 부호화 및 복호화할 때의 영상의 분할 구조를 개략적으로 나타내는 도면이다. 도 3은 하나의 유닛이 복수의 하위 유닛으로 분할되는 실시예를 개략적으로 나타낸다.

[0147] 영상을 효율적으로 분할하기 위해, 부호화 및 복호화에 있어서, 부호화 유닛(Coding Unit; CU)이 사용될 수 있다. 영상 부호화/복호화의 기본 단위로서 부호화 유닛이 사용될 수 있다. 또한, 영상 부호화/복호화 시 화면 내 예측 모드 및 화면 간 예측 모드가 구분되는 단위로 부호화 유닛을 사용할 수 있다. 부호화 유닛은 예측, 변환, 양자화, 역변환, 역양자화, 또는 변환 계수의 부호화/복호화의 과정을 위해 사용되는 기본 단위일 수 있다.

[0148] 도 3을 참조하면, 영상(300)은 최대 부호화 유닛(Largest Coding Unit; LCU) 단위로 순차적으로 분할되고, LCU 단위로 분할 구조가 결정된다. 여기서, LCU는 부호화 트리 유닛(Coding Tree Unit; CTU)과 동일한 의미로 사용될 수 있다. 유닛의 분할은 유닛에 해당하는 블록의 분할을 의미할 수 있다. 블록 분할 정보에는 유닛의 깊이(depth)에 관한 정보가 포함될 수 있다. 깊이 정보는 유닛이 분할되는 회수 및/또는 정도를 나타낼 수 있다. 하나의 유닛은 트리 구조(tree structure)를 기초로 깊이 정보를 가지고 계층적으로 복수의 하위 유닛들로 분할될 수 있다. 말하자면, 유닛 및 상기의 유닛의 분할에 의해 생성된 하위 유닛은 노드 및 상기의 노드의 자식 노드에 각각 대응할 수 있다. 각각의 분할된 하위 유닛은 깊이 정보를 가질 수 있다. 깊이 정보는 CU의 크기를 나타내는 정보일 수 있고, 각 CU마다 저장될 수 있다. 유닛 깊이는 유닛이 분할된 회수 및/또는 정도를 나타내므로, 하위 유닛의 분할 정보는 하위 유닛의 크기에 관한 정보를 포함할 수도 있다.

[0149] 분할 구조는 CTU(310) 내에서의 부호화 유닛(Coding Unit; CU)의 분포를 의미할 수 있다. 이러한 분포는 하나의 CU를 복수(2, 4, 8, 16 등을 포함하는 2 이상의 양의 정수)의 CU들로 분할할지 여부에 따라 결정할 수 있다. 분할에 의해 생성된 CU의 가로 크기 및 세로 크기는 각각 분할 전의 CU의 가로 크기의 절반 및 세로 크기의 절반이거나, 분할된 개수에 따라 분할 전의 CU의 가로 크기보다 작은 크기 및 세로 크기보다 작은 크기를 가질 수

있다. CU는 복수의 CU로 재귀적으로 분할될 수 있다. 재귀적 분할에 의해, 분할된 CU의 가로 크기 및 세로 크기 중 적어도 하나의 크기가 분할 전의 CU의 가로 크기 및 세로 크기 중 적어도 하나에 비해 감소될 수 있다. CU의 분할은 기정의된 깊이 또는 기정의된 크기까지 재귀적으로 이루어질 수 있다. 예컨대, CTU의 깊이는 0일 수 있고, 최소 부호화 유닛(Smallest Coding Unit; SCU)의 깊이는 기정의된 최대 깊이일 수 있다. 여기서, CTU는 상술된 것과 같이 최대의 부호화 유닛 크기를 가지는 부호화 유닛일 수 있고, SCU는 최소의 부호화 유닛 크기를 가지는 부호화 유닛일 수 있다. CTU(310)로부터 분할이 시작되고, 분할에 의해 CU의 가로 크기 및/또는 세로 크기가 줄어들 때마다 CU의 깊이는 1씩 증가한다. 예를 들면, 각각의 깊이 별로, 분할되지 않는 CU는 2Nx2N 크기를 가질 수 있다. 또한, 분할되는 CU의 경우, 2Nx2N 크기의 CU가 NxN 크기를 가지는 4개의 CU들로 분할될 수 있다. N의 크기는 깊이가 1씩 증가할 때마다 절반으로 감소할 수 있다.

[0150] 또한, CU가 분할되는지 여부에 대한 정보는 CU의 분할 정보를 통해 표현될 수 있다. 분할 정보는 1비트의 정보일 수 있다. SCU를 제외한 모든 CU는 분할 정보를 포함할 수 있다. 예를 들면, 분할 정보의 값이 제1 값이면, CU가 분할되지 않을 수 있고, 분할 정보의 값이 제2 값이면, CU가 분할될 수 있다.

[0151] 도 3을 참조하면, 깊이가 0인 CTU는 64x64 블록일 수 있다. 0은 최소 깊이일 수 있다. 깊이가 3인 SCU는 8x8 블록일 수 있다. 3은 최대 깊이일 수 있다. 32x32 블록 및 16x16 블록의 CU는 각각 깊이 1 및 깊이 2로 표현될 수 있다.

[0152] 예를 들어, 하나의 부호화 유닛이 4개의 부호화 유닛으로 분할 될 경우, 분할된 4개의 부호화 유닛의 가로 및 세로 크기는 분할되기 전 부호화 유닛의 가로 및 세로 크기와 비교하여 각각 절반의 크기를 가질 수 있다. 일 예로, 32x32 크기의 부호화 유닛이 4개의 부호화 유닛으로 분할 될 경우, 분할된 4개의 부호화 유닛은 각각 16x16의 크기를 가질 수 있다. 하나의 부호화 유닛이 4개의 부호화 유닛으로 분할 될 경우, 부호화 유닛은 쿼드트리(quad-tree) 형태로 분할(쿼드트리 분할, quad-tree partition)되었다고 할 수 있다.

[0153] 예를 들어, 하나의 부호화 유닛이 2개의 부호화 유닛으로 분할 될 경우, 분할된 2개의 부호화 유닛의 가로 혹은 세로 크기는 분할되기 전 부호화 유닛의 가로 혹은 세로 크기와 비교하여 절반의 크기를 가질 수 있다. 일 예로, 32x32 크기의 부호화 유닛이 2개의 부호화 유닛으로 세로로 분할 될 경우, 분할된 2개의 부호화 유닛은 각각 16x32의 크기를 가질 수 있다. 일 예로, 8x32 크기의 부호화 유닛이 2개의 부호화 유닛으로 가로로 분할 될 경우, 분할된 2개의 부호화 유닛은 각각 8x16의 크기를 가질 수 있다. 하나의 부호화 유닛이 2개의 부호화 유닛으로 분할 될 경우, 부호화 유닛은 이진트리(binary-tree) 형태로 분할(이진트리 분할, binary-tree partition)되었다고 할 수 있다.

[0154] 예를 들어, 하나의 부호화 유닛이 3개의 부호화 유닛으로 분할 될 경우, 분할되기 전 부호화 유닛의 가로 혹은 세로 크기를 1:2:1의 비율로 분할함으로써, 3개의 부호화 유닛으로 분할할 수 있다. 일 예로, 16x32 크기의 부호화 유닛이 3개의 부호화 유닛으로 가로로 분할 될 경우, 분할된 3개의 부호화 유닛은 상측부터 각각 16x8, 16x16 및 16x8의 크기를 가질 수 있다. 일 예로, 32x32 크기의 부호화 유닛이 3개의 부호화 유닛으로 세로로 분할 될 경우, 분할된 3개의 부호화 유닛은 좌측부터 각각 8x32, 16x32 및 8x32의 크기를 가질 수 있다. 하나의 부호화 유닛이 3개의 부호화 유닛으로 분할 될 경우, 부호화 유닛은 3분할트리(ternary-tree) 형태로 분할(3분할트리 분할, ternary-tree partition)되었다고 할 수 있다.

[0155] 도 3의 CTU(320)는 쿼드트리 분할, 이진트리 분할 및 3분할트리 분할이 모두 적용된 CTU의 일 예이다.

[0156] 전술한 바와 같이, CTU를 분할하기 위해, 쿼드트리 분할, 이진트리 분할 및 3분할트리 분할 중 적어도 하나가 적용될 수 있다. 각각의 분할은 소정의 우선 순위에 기초하여 적용될 수 있다. 예컨대, CTU에 대해 쿼드트리 분할이 우선적으로 적용될 수 있다. 더 이상 쿼드트리 분할될 수 없는 부호화 유닛은 쿼드트리의 리프 노드에 해당될 수 있다. 쿼드트리의 리프 노드에 해당하는 부호화 유닛은 이진트리 및/또는 3분할트리의 루트 노드가 될 수 있다. 즉, 쿼드트리의 리프 노드에 해당하는 부호화 유닛은 이진트리 분할되거나 3분할트리 분할되거나 또는 더 이상 분할되지 않을 수 있다. 이 때, 쿼드트리의 리프 노드에 해당하는 부호화 유닛을 이진트리 분할하거나 3분할트리 분할하여 생성된 부호화 유닛에 대해서는 다시 쿼드트리 분할이 수행되지 않도록 함으로써, 블록의 분할 및/또는 분할 정보의 시그널링을 효과적으로 수행할 수 있다.

[0157] 쿼드트리의 각 노드에 해당하는 부호화 유닛의 분할은 쿼드 분할 정보를 이용하여 시그널링될 수 있다. 제1값(예컨대, '1')을 갖는 쿼드 분할 정보는 해당 부호화 유닛이 쿼드트리 분할됨을 지시할 수 있다. 제2값(예컨대, '0')을 갖는 쿼드 분할 정보는 해당 부호화 유닛이 쿼드트리 분할되지 않음을 지시할 수 있다. 쿼드 분할 정보는 소정의 길이(예컨대, 1비트)를 갖는 플래그일 수 있다.

[0158] 이진트리 분할과 3분할트리 분할 사이에는 우선순위가 존재하지 않을 수 있다. 즉, 쿼드트리의 리프 노드에 해당하는 부호화 유닛은 이진트리 분할되거나 3분할트리 분할될 수 있다. 또한, 이진트리 분할 또는 3분할트리 분할에 의해 생성된 부호화 유닛은 다시 이진트리 분할 또는 3분할트리 분할되거나 또는 더 이상 분할되지 않을 수 있다.

[0159] 이진트리 분할과 3분할트리 분할 사이에 우선순위가 존재하지 않는 경우의 분할은 복합형트리 분할(multi-type tree partition)이라고 호칭할 수 있다. 즉, 쿼드트리의 리프 노드에 해당하는 부호화 유닛은 복합형트리(multi-type tree)의 루트 노드가 될 수 있다. 복합형트리의 각 노드에 해당하는 부호화 유닛의 분할은 복합형트리의 분할 여부 정보, 분할 방향 정보 및 분할 트리 정보 중 적어도 하나를 이용하여 시그널링될 수 있다. 상기 복합형트리의 각 노드에 해당하는 부호화 유닛의 분할을 위해 순차적으로 분할 여부 정보, 분할 방향 정보 및 분할 트리 정보가 시그널링될 수도 있다.

[0160] 제1값(예컨대, '1')을 갖는 복합형트리의 분할 여부 정보는 해당 부호화 유닛이 복합형트리 분할됨을 지시할 수 있다. 제2값(예컨대, '0')을 갖는 복합형트리의 분할 여부 정보는 해당 부호화 유닛이 복합형트리 분할되지 않음을 지시할 수 있다.

[0161] 복합형트리의 각 노드에 해당하는 부호화 유닛이 복합형트리 분할되는 경우, 해당 부호화 유닛은 분할 방향 정보를 더 포함할 수 있다. 분할 방향 정보는 복합형트리 분할의 분할 방향을 지시할 수 있다. 제1값(예컨대, '1')을 갖는 분할 방향 정보는 해당 부호화 유닛이 세로 방향으로 분할됨을 지시할 수 있다. 제2값(예컨대, '0')을 갖는 분할 방향 정보는 해당 부호화 유닛이 가로 방향으로 분할됨을 지시할 수 있다.

[0162] 복합형트리의 각 노드에 해당하는 부호화 유닛이 복합형트리 분할되는 경우, 해당 부호화 유닛은 분할 트리 정보를 더 포함할 수 있다. 분할 트리 정보는 복합형트리 분할을 위해 사용된 트리를 지시할 수 있다. 제1값(예컨대, '1')을 갖는 분할 트리 정보는 해당 부호화 유닛이 이진트리 분할됨을 지시할 수 있다. 제2값(예컨대, '0')을 갖는 분할 트리 정보는 해당 부호화 유닛이 3분할트리 분할됨을 지시할 수 있다.

[0163] 분할 여부 정보, 분할 트리 정보 및 분할 방향 정보는 각각 소정의 길이(예컨대, 1비트)를 갖는 플래그일 수 있다.

[0164] 쿼드 분할 정보, 복합형트리의 분할 여부 정보, 분할 방향 정보 및 분할 트리 정보 중 적어도 하나는 엔트로피 부호화/복호화될 수 있다. 상기 정보들의 엔트로피 부호화/복호화를 위해, 현재 부호화 유닛에 인접한 주변 부호화 유닛의 정보가 이용될 수 있다. 예컨대, 좌측 부호화 유닛 및/또는 상측 부호화 유닛의 분할 형태(분할 여부, 분할 트리 및/또는 분할 방향)는 현재 부호화 유닛의 분할 형태와 유사할 확률이 높다. 따라서, 주변 부호화 유닛의 정보에 기초하여, 현재 부호화 유닛의 정보의 엔트로피 부호화/복호화를 위한 컨텍스트 정보를 유도할 수 있다. 이때, 주변 부호화 유닛의 정보에는 해당 부호화 유닛의 쿼드 분할 정보, 복합형트리의 분할 여부 정보, 분할 방향 정보 및 분할 트리 정보 중 적어도 하나가 포함될 수 있다.

[0165] 다른 실시예로서, 이진트리 분할과 3분할트리 분할 중, 이진트리 분할이 우선적으로 수행될 수 있다. 즉, 이진트리 분할이 먼저 적용되고, 이진트리의 리프 노드에 해당하는 부호화 유닛을 3분할트리의 루트 노드로 설정할 수도 있다. 이 경우, 3분할트리의 노드에 해당하는 부호화 유닛에 대해서는 쿼드트리 분할 및 이진트리 분할이 수행되지 않을 수 있다.

[0166] 쿼드트리 분할, 이진트리 분할 및/또는 3분할트리 분할에 의해 더 이상 분할되지 않는 부호화 유닛은 부호화, 예측 및/또는 변환의 단위가 될 수 있다. 즉, 예측 및/또는 변환을 위해 부호화 유닛이 더 이상 분할되지 않을 수 있다. 따라서, 부호화 유닛을 예측 유닛 및/또는 변환 유닛으로 분할하기 위한 분할 구조, 분할 정보 등이 비트스트림에 존재하지 않을 수 있다.

[0167] 다만, 분할의 단위가 되는 부호화 유닛의 크기가 최대 변환 블록의 크기보다 큰 경우, 해당 부호화 유닛은 최대 변환 블록의 크기와 같거나 또는 작은 크기가 될 때까지 재귀적으로 분할될 수 있다. 예컨대, 부호화 유닛의 크기가 64x64이고, 최대 변환 블록의 크기가 32x32인 경우, 상기 부호화 유닛은 변환을 위해, 4개의 32x32 블록으로 분할될 수 있다. 예컨대, 부호화 유닛의 크기가 32x64이고, 최대 변환 블록의 크기가 32x32인 경우, 상기 부호화 유닛은 변환을 위해, 2개의 32x32 블록으로 분할될 수 있다. 이 경우, 변환을 위한 부호화 유닛의 분할 여부는 별도로 시그널링되지 않고, 상기 부호화 유닛의 가로 또는 세로와 최대 변환 블록의 가로 또는 세로의 비교에 의해 결정될 수 있다. 예컨대, 부호화 유닛의 가로가 최대 변환 블록의 가로보다 큰 경우, 부호화 유닛은 세로로 2등분될 수 있다. 또한, 부호화 유닛의 세로가 최대 변환 블록의 세로보다 큰 경우, 부호화 유닛은 가로로 2등분될 수 있다.

[0168] 부호화 유닛의 최대 및/또는 최소 크기에 관한 정보, 변환 블록의 최대 및/또는 최소 크기에 관한 정보는 부호화 유닛의 상위 레벨에서 시그널링되거나 결정될 수 있다. 상기 상위 레벨은 예컨대, 시퀀스 레벨, 픽처 레벨, 타일 레벨, 타일 그룹 레벨, 슬라이스 레벨 등일 수 있다. 예컨대, 부호화 유닛의 최소 크기는 4x4로 결정될 수 있다. 예컨대, 변환 블록의 최대 크기는 64x64로 결정될 수 있다. 예컨대, 변환 블록의 최소 크기는 4x4로 결정될 수 있다.

[0169] 쿼드트리의 리프 노드에 해당하는 부호화 유닛의 최소 크기(쿼드트리 최소 크기)에 관한 정보 및/또는 복합형트리의 루트 노드에서 리프 노드에 이르는 최대 깊이(복합형트리 최대 깊이)에 관한 정보는 부호화 유닛의 상위 레벨에서 시그널링되거나 결정될 수 있다. 상기 상위 레벨은 예컨대, 시퀀스 레벨, 픽처 레벨, 슬라이스 레벨, 타일 그룹 레벨, 타일 레벨 등일 수 있다. 상기 쿼드트리 최소 크기에 관한 정보 및/또는 상기 복합형트리 최대 깊이에 관한 정보는 화면 내 슬라이스와 화면 간 슬라이스의 각각에 대해 시그널링되거나 결정될 수 있다.

[0170] CTU의 크기와 변환 블록의 최대 크기에 대한 차분 정보는 부호화 유닛의 상위 레벨에서 시그널링되거나 결정될 수 있다. 상기 상위 레벨은 예컨대, 시퀀스 레벨, 픽처 레벨, 슬라이스 레벨, 타일 그룹 레벨, 타일 레벨 등일 수 있다. 이진트리의 각 노드에 해당하는 부호화 유닛의 최대 크기(이진트리 최대 크기)에 관한 정보는 부호화 트리 유닛의 크기와 상기 차분 정보를 기반으로 결정될 수 있다. 3분할트리의 각 노드에 해당하는 부호화 유닛의 최대 크기(3분할트리 최대 크기)는 슬라이스의 타입에 따라 다른 값을 가질 수 있다. 예컨대, 화면 내 슬라이스인 경우, 3분할트리 최대 크기는 32x32일 수 있다. 또한, 예컨대, 화면 간 슬라이스인 경우, 3분할 트리 최대 크기는 128x128일 수 있다. 예컨대, 이진트리의 각 노드에 해당하는 부호화 유닛의 최소 크기(이진트리 최소 크기) 및/또는 3분할트리의 각 노드에 해당하는 부호화 유닛의 최소 크기(3분할트리 최소 크기)는 부호화 블록의 최소 크기로 설정될 수 있다.

[0171] 또 다른 예로, 이진트리 최대 크기 및/또는 3분할트리 최대 크기는 슬라이스 레벨에서 시그널링되거나 결정될 수 있다. 또한, 이진트리 최소 크기 및/또는 3분할트리 최소 크기는 슬라이스 레벨에서 시그널링되거나 결정될 수 있다.

[0172] 전술한 다양한 블록의 크기 및 깊이 정보에 기초하여, 쿼드 분할 정보, 복합형트리의 분할 여부 정보, 분할 트리 정보 및/또는 분할 방향 정보 등이 비트스트림에 존재하거나 존재하지 않을 수 있다.

[0173] 예컨대, 부호화 유닛의 크기가 쿼드트리 최소 크기보다 크지 않으면, 상기 부호화 유닛은 쿼드 분할 정보를 포함하지 않고, 해당 쿼드 분할 정보는 제2값으로 추론될 수 있다.

[0174] 예컨대, 복합형트리의 노드에 해당하는 부호화 유닛의 크기(가로 및 세로)가 이진트리 최대 크기(가로 및 세로) 및/또는 3분할트리 최대 크기(가로 및 세로)보다 큰 경우, 상기 부호화 유닛은 이진트리 분할 및/또는 3분할트리 분할되지 않을 수 있다. 그에 따라, 상기 복합형트리의 분할 여부 정보는 시그널링되지 않고, 제2값으로 추론될 수 있다.

[0175] 또는, 복합형트리의 노드에 해당하는 부호화 유닛의 크기(가로 및 세로)가 이진트리 최소 크기(가로 및 세로)와 동일하거나, 부호화 유닛의 크기(가로 및 세로)가 3분할트리 최소 크기(가로 및 세로)의 두 배와 동일한 경우, 상기 부호화 유닛은 이진트리 분할 및/또는 3분할트리 분할되지 않을 수 있다. 그에 따라, 상기 복합형트리의 분할 여부 정보는 시그널링되지 않고, 제2값으로 추론될 수 있다. 왜냐하면, 상기 부호화 유닛을 이진트리 분할 및/또는 3분할트리 분할할 경우, 이진트리 최소 크기 및/또는 3분할트리 최소 크기보다 작은 부호화 유닛이 생성되기 때문이다.

[0176] 또는, 이진트리 분할 또는 3분할트리 분할은 가상의 파이프라인 데이터 유닛의 크기(이하, 파이프라인 버퍼 크기)에 기초하여 제한될 수 있다. 예컨대, 이진트리 분할 또는 3분할트리 분할에 의해, 부호화 유닛이 파이프라인 버퍼 크기에 적합하지 않은 서브 부호화 유닛으로 분할될 경우, 해당 이진트리 분할 또는 3분할트리 분할은 제한될 수 있다. 파이프라인 버퍼 크기는 최대 변환 블록의 크기(예컨대, 64X64)일 수 있다. 예컨대, 파이프라인 버퍼 크기가 64X64일 때, 아래의 분할은 제한될 수 있다.

[0177] - NxM(N 및/또는 M은 128) 부호화 유닛에 대한 3분할트리 분할

[0178] - 128xN(N <= 64) 부호화 유닛에 대한 수평 방향 이진트리 분할

[0179] - Nx128(N <= 64) 부호화 유닛에 대한 수직 방향 이진트리 분할

[0180] 또는, 복합형트리의 노드에 해당하는 부호화 유닛의 복합형트리 내의 깊이가 복합형트리 최대 깊이와 동일한 경우, 상기 부호화 유닛은 이진트리 분할 및/또는 3분할트리 분할되지 않을 수 있다. 그에 따라, 상기 복합형트리

의 분할 여부 정보는 시그널링되지 않고, 제2값으로 추론될 수 있다.

[0181] 또는, 복합형트리의 노드에 해당하는 부호화 유닛에 대해 수직 방향 이진트리 분할, 수평 방향 이진트리 분할, 수직 방향 3분할트리 분할 및 수평 방향 3분할트리 분할 중 적어도 하나가 가능한 경우에만, 상기 복합형트리의 분할 여부 정보를 시그널링할 수 있다. 그렇지 않은 경우, 상기 부호화 유닛은 이진트리 분할 및/또는 3분할트리 분할되지 않을 수 있다. 그에 따라, 상기 복합형트리의 분할 여부 정보는 시그널링되지 않고, 제2값으로 추론될 수 있다.

[0182] 또는, 복합형트리의 노드에 해당하는 부호화 유닛에 대해 수직 방향 이진트리 분할과 수평 방향 이진트리 분할이 모두 가능하거나, 수직 방향 3분할트리 분할과 수평 방향 3분할트리 분할이 모두 가능한 경우에만, 상기 분할 방향 정보를 시그널링할 수 있다. 그렇지 않은 경우, 상기 분할 방향 정보는 시그널링되지 않고, 분할이 가능한 방향을 지시하는 값으로 추론될 수 있다.

[0183] 또는, 복합형트리의 노드에 해당하는 부호화 유닛에 대해 수직 방향 이진트리 분할과 수직 방향 3분할트리 분할이 모두 가능하거나, 수평 방향 이진트리 분할과 수평 방향 3분할트리 분할이 모두 가능한 경우에만, 상기 분할 트리 정보를 시그널링할 수 있다. 그렇지 않은 경우, 상기 분할 트리 정보는 시그널링되지 않고, 분할이 가능한 트리를 지시하는 값으로 추론될 수 있다.

[0184] 도 4는 화면 내 예측 과정의 실시예를 설명하기 위한 도면이다.

[0185] 도 4의 중심으로부터 외곽으로의 화살표들은 화면 내 예측 모드들의 예측 방향들을 나타낼 수 있다.

[0186] 화면 내 부호화 및/또는 복호화는 현재 블록의 주변 블록의 참조 샘플을 이용하여 수행될 수 있다. 주변 블록은 복원된 주변 블록일 수 있다. 예를 들면, 화면 내 부호화 및/또는 복호화는 복원된 주변 블록이 포함하는 참조 샘플의 값 또는 부호화 파라미터를 이용하여 수행될 수 있다.

[0187] 예측 블록은 화면 내 예측의 수행의 결과로 생성된 블록을 의미할 수 있다. 예측 블록은 CU, PU 및 TU 중 적어도 하나에 해당할 수 있다. 예측 블록의 단위는 CU, PU 및 TU 중 적어도 하나의 크기일 수 있다. 예측 블록은 2x2, 4x4, 16x16, 32x32 또는 64x64 등의 크기를 갖는 정사각형의 형태의 블록일 수 있고, 2x8, 4x8, 2x16, 4x16 및 8x16 등의 크기를 갖는 직사각형 모양의 블록일 수도 있다.

[0188] 화면 내 예측은 현재 블록에 대한 화면 내 예측 모드에 따라 수행될 수 있다. 현재 블록이 가질 수 있는 화면 내 예측 모드의 개수는 기정의된 고정된 값일 수 있으며, 예측 블록의 속성에 따라 다르게 결정된 값일 수 있다. 예를 들면, 예측 블록의 속성은 예측 블록의 크기 및 예측 블록의 형태 등을 포함할 수 있다.

[0189] 화면 내 예측 모드의 개수는 블록의 크기에 관계없이 N개로 고정될 수 있다. 또는, 예를 들면, 화면 내 예측 모드의 개수는 3, 5, 9, 17, 34, 35, 36, 65, 또는 67 등일 수 있다. 또는, 화면 내 예측 모드의 개수는 블록의 크기 및/또는 색 성분(color component)의 타입에 따라 상이할 수 있다. 예를 들면, 색 성분이 휘도(luma) 신호인지 아니면 색차(chroma) 신호인지에 따라 화면 내 예측 모드의 개수가 다를 수 있다. 예컨대, 블록의 크기가 커질수록 화면 내 예측 모드의 개수는 많아질 수 있다. 또는 휘도 성분 블록의 화면 내 예측 모드의 개수는 색차 성분 블록의 화면 내 예측 모드의 개수보다 많을 수 있다.

[0190] 화면 내 예측 모드는 비방향성 모드 또는 방향성 모드일 수 있다. 비방향성 모드는 DC 모드 또는 플래너(Planar) 모드일 수 있으며, 방향성 모드(angular mode)는 특정한 방향 또는 각도를 가지는 예측 모드일 수 있다. 상기 화면 내 예측 모드는 모드 번호, 모드 값, 모드 숫자, 모드 각도, 모드 방향 중 적어도 하나로 표현될 수 있다. 화면 내 예측 모드의 개수는 상기 비방향성 및 방향성 모드를 포함하는 하나 이상의 M개 일 수 있다. 현재 블록을 화면 내 예측하기 위해 복원된 주변 블록에 포함되는 샘플들이 현재 블록의 참조 샘플로 이용 가능한지 여부를 검사하는 단계가 수행될 수 있다. 현재 블록의 참조 샘플로 이용할 수 없는 샘플이 존재할 경우, 복원된 주변 블록에 포함된 샘플들 중 적어도 하나의 샘플 값을 복사 및/또는 보간한 값을 이용하여 참조 샘플로 이용할 수 없는 샘플의 샘플 값으로 대체한 후, 현재 블록의 참조 샘플로 이용할 수 있다.

[0191] 도 7은 화면 내 예측에 이용 가능한 참조 샘플들을 설명하기 위한 도면이다.

[0192] 도 7에 도시된 바와 같이, 현재 블록의 화면 내 예측을 위해, 참조 샘플 라인 0 내지 참조 샘플 라인 3 중 적어도 하나가 이용될 수 있다. 도 7에 있어서, 세그먼트 A와 세그먼트 F의 샘플들은 복원된 이웃 블록으로부터 가져오는 대신 각각 세그먼트 B와 세그먼트 E의 가장 가까운 샘플들로 패딩될 수 있다. 현재 블록의 화면 내 예측을 위해 이용될 참조 샘플 라인을 지시하는 인덱스 정보가 시그널링될 수 있다. 현재 블록의 상단 경계가 CTU의 경계인 경우, 참조 샘플 라인 0만 이용가능할 수 있다. 따라서 이 경우, 상기 인덱스 정보는 시그널링되지 않을

수 있다. 참조 샘플 라인 0 이외에 다른 참조 샘플 라인이 이용되는 경우, 후술하는 예측 블록에 대한 필터링은 수행되지 않을 수 있다.

[0193] 화면 내 예측 시 화면 내 예측 모드 및 현재 블록의 크기 중 적어도 하나에 기반하여 참조 샘플 또는 예측 샘플 중 적어도 하나에 필터를 적용할 수 있다.

[0194] 플래너 모드의 경우, 현재 블록의 예측 블록을 생성할 때, 예측 대상 샘플의 예측 블록 내 위치에 따라, 현재 샘플의 상단 및 좌측 참조 샘플, 현재 블록의 우상단 및 좌하단 참조 샘플의 가중합을 이용하여 예측 대상 샘플의 샘플값을 생성할 수 있다. 또한, DC 모드의 경우, 현재 블록의 예측 블록을 생성할 때, 현재 블록의 상단 및 좌측 참조 샘플들의 평균 값을 이용할 수 있다. 또한, 방향성 모드의 경우 현재 블록의 상단, 좌측, 우상단 및/또는 좌하단 참조 샘플을 이용하여 예측 블록을 생성할 수 있다. 예측 샘플 값 생성을 위해 실수 단위의 보간을 수행할 수도 있다.

[0195] 색 성분간 화면내 예측의 경우, 제1 색 성분의 대응 복원 블록에 기초하여 제2 색 성분의 현재 블록에 대한 예측 블록을 생성할 수 있다. 예컨대, 제1 색 성분은 휘도 성분, 제2 색 성분은 색차 성분일 수 있다. 색 성분간 화면내 예측을 위해, 제1 색 성분과 제2 색 성분 간의 선형 모델의 파라미터가 템플릿에 기초하여 유도될 수 있다. 템플릿은 현재 블록의 상단 및/또는 좌측 주변 샘플 및 이에 대응하는 제1 색 성분의 복원 블록의 상단 및/또는 좌측 주변 샘플을 포함할 수 있다. 예컨대, 선형 모델의 파라미터는 템플릿내의 샘플들 중 최대값을 갖는 제1 색 성분의 샘플값과 이에 대응하는 제2 색 성분의 샘플값, 템플릿내의 샘플들 중 최소값을 갖는 제1 색 성분의 샘플값과 이에 대응하는 제2 색 성분의 샘플값을 이용하여 유도될 수 있다. 선형 모델의 파라미터가 유도되면, 대응 복원 블록을 선형 모델에 적용하여 현재 블록에 대한 예측 블록을 생성할 수 있다. 영상 포맷에 따라, 제1 색 성분의 복원 블록의 주변 샘플 및 대응 복원 블록에 대해 서브 샘플링이 수행될 수 있다. 예컨대, 제2 색 성분의 1개의 샘플이 제1 색 성분의 4개의 샘플들에 대응되는 경우, 제1 색 성분의 4개의 샘플들을 서브 샘플링하여, 1개의 대응 샘플을 계산할 수 있다. 이 경우, 선형 모델의 파라미터 유도 및 색 성분간 화면 내 예측은 서브 샘플링된 대응 샘플에 기초하여 수행될 수 있다. 색 성분간 화면내 예측의 수행 여부 및/또는 템플릿의 범위는 화면내 예측 모드로서 시그널링될 수 있다.

[0196] 현재 블록은 가로 또는 세로 방향으로 2개 또는 4개의 서브 블록들로 분할될 수 있다. 분할된 서브 블록들은 순차적으로 복원될 수 있다. 즉, 서브 블록에 대해 화면 내 예측이 수행되어 서브 예측 블록이 생성될 수 있다. 또한, 서브 블록에 대해 역양자화 및/또는 역변환이 수행되어 서브 잔차 블록이 생성될 수 있다. 서브 예측 블록을 서브 잔차 블록에 더해서 복원된 서브 블록이 생성될 수 있다. 복원된 서브 블록은 후순위 서브 블록의 화면 내 예측을 위한 참조 샘플로서 이용될 수 있다. 서브 블록은 소정 개수(예컨대, 16개) 이상의 샘플들을 포함하는 블록일 수 있다. 따라서, 예컨대, 현재 블록이 8x4 블록 또는 4x8 블록의 경우, 현재 블록은 2개의 서브 블록들로 분할될 수 있다. 또한, 현재 블록이 4x4 블록인 경우, 현재 블록은 서브 블록들로 분할될 수 없다. 현재 블록이 그 외의 크기를 갖는 경우, 현재 블록은 4개의 서브 블록들로 분할될 수 있다. 상기 서브 블록 기반의 화면 내 예측의 수행 여부 및/또는 분할 방향(가로 또는 세로)에 관한 정보가 시그널링될 수 있다. 상기 서브 블록 기반의 화면 내 예측은 참조 샘플 라인 0을 이용하는 경우에만 수행되도록 제한될 수 있다. 상기 서브 블록 기반의 화면 내 예측이 수행되는 경우, 후술하는 예측 블록에 대한 필터링은 수행되지 않을 수 있다.

[0197] 화면내 예측된 예측 블록에 필터링을 수행하여 최종 예측 블록을 생성할 수 있다. 상기 필터링은 필터링 대상 샘플, 좌측 참조 샘플, 상단 참조 샘플 및/또는 좌상단 참조 샘플에 소정의 가중치를 적용함으로써 수행될 수 있다. 상기 필터링에 이용되는 가중치 및/또는 참조 샘플(범위, 위치 등)은 블록 크기, 화면내 예측 모드 및 필터링 대상 샘플의 예측 블록 내 위치 중 적어도 하나에 기초하여 결정될 수 있다. 상기 필터링은 소정의 화면내 예측 모드(예컨대, DC, planar, 수직, 수평, 대각 및/또는 인접 대각 모드)의 경우에만 수행될 수 있다. 인접 대각 모드는 대각 모드에 k를 가감한 모드일 수 있다. 예컨대, k는 8 이하의 양의 정수일 수 있다.

[0198] 현재 블록의 화면 내 예측 모드는 현재 블록의 주변에 존재하는 블록의 화면 내 예측 모드로부터 예측하여 엔트로피 부호화/복호화할 수 있다. 현재 블록과 주변 블록의 화면 내 예측 모드가 동일하면 소정의 플래그 정보를 이용하여 현재 블록과 주변 블록의 화면 내 예측 모드가 동일하다는 정보를 시그널링할 수 있다. 또한, 복수 개의 주변 블록의 화면 내 예측 모드 중 현재 블록의 화면 내 예측 모드와 동일한 화면 내 예측 모드에 대한 지시자 정보를 시그널링 할 수 있다. 현재 블록과 주변 블록의 화면 내 예측 모드가 상이하면 주변 블록의 화면 내 예측 모드를 기초로 엔트로피 부호화/복호화를 수행하여 현재 블록의 화면 내 예측 모드 정보를 엔트로피 부호화/복호화할 수 있다.

[0199] 도 5는 화면 간 예측 과정의 실시예를 설명하기 위한 도면이다.

[0200] 도 5에 도시된 사각형은 영상을 나타낼 수 있다. 또한, 도 5에서 화살표는 예측 방향을 나타낼 수 있다. 각 영상은 부호화 타입에 따라 I 픽처(Intra Picture), P 픽처(Predictive Picture), B 픽처(Bi-predictive Picture) 등으로 분류될 수 있다.

[0201] I 픽처는 화면 간 예측 없이 화면 내 예측을 통해 부호화/복호화될 수 있다. P 픽처는 단방향(예컨대, 순방향 또는 역방향)에 존재하는 참조 영상만을 이용하는 화면 간 예측을 통해 부호화/복호화될 수 있다. B 픽처는 쌍방향(예컨대, 순방향 및 역방향)에 존재하는 참조 영상들을 이용하는 화면 간 예측을 통해 부호화/복호화 될 수 있다. 또한, B 픽처인 경우, 쌍방향에 존재하는 참조 영상들을 이용하는 화면 간 예측 또는 순방향 및 역방향 중 일 방향에 존재하는 참조 영상을 이용하는 화면 간 예측을 통해 부호화/복호화될 수 있다. 여기에서, 쌍방향은 순방향 및 역방향일 수 있다. 여기서, 화면 간 예측이 사용되는 경우, 부호화기에서는 화면 간 예측 혹은 움직임 보상을 수행할 수 있고, 복호화기에서는 그에 대응하는 움직임 보상을 수행할 수 있다.

[0202] 아래에서, 실시예에 따른 화면 간 예측에 대해 구체적으로 설명된다.

[0203] 화면 간 예측 혹은 움직임 보상은 참조 영상 및 움직임 정보를 이용하여 수행될 수 있다.

[0204] 현재 블록에 대한 움직임 정보는 부호화 장치(100) 및 복호화 장치(200)의 각각에 의해 화면 간 예측 중 도출될 수 있다. 움직임 정보는 복원된 주변 블록의 움직임 정보, 콜 블록(collocated block; col block)의 움직임 정보 및/또는 콜 블록에 인접한 블록을 이용하여 도출될 수 있다. 콜 블록은 이미 복원된 콜 픽처(collocated picture; col picture) 내에서 현재 블록의 공간적 위치에 대응하는 블록일 수 있다. 여기서, 콜 픽처는 참조 영상 리스트에 포함된 적어도 하나의 참조 영상 중에서 하나의 픽처일 수 있다.

[0205] 움직임 정보의 도출 방식은 현재 블록의 예측 모드에 따라 다를 수 있다. 예를 들면, 화면 간 예측을 위해 적용되는 예측 모드로서, AMVP 모드, 머지 모드, 스킵 모드, 움직임 벡터 차분을 가진 머지 모드, 서브 블록 머지 모드, 삼각 분할 모드, 인터 인트라 결합 예측 모드, 어파인 인터 모드 등이 있을 수 있다. 여기서 머지 모드를 움직임 병합 모드(motion merge mode)라고 지칭할 수 있다.

[0206] 예를 들면, 예측 모드로서, AMVP가 적용되는 경우, 복원된 주변 블록의 움직임 벡터, 콜 블록의 움직임 벡터, 콜 블록에 인접한 블록의 움직임 벡터, (0, 0) 움직임 벡터 중 적어도 하나를 움직임 벡터 후보로 결정하여 움직임 벡터 후보 리스트(motion vector candidate list)를 생성할 수 있다. 생성된 움직임 벡터 후보 리스트를 이용하여 움직임 벡터 후보를 유도할 수 있다. 유도된 움직임 벡터 후보를 기반으로 현재 블록의 움직임 정보를 결정할 수 있다. 여기서, 콜 블록의 움직임 벡터 또는 콜 블록에 인접한 블록의 움직임 벡터를 시간적 움직임 벡터 후보(temporal motion vector candidate)라 지칭할 수 있고, 복원된 주변 블록의 움직임 벡터를 공간적 움직임 벡터 후보(spatial motion vector candidate)라 지칭할 수 있다.

[0207] 부호화 장치(100)는 현재 블록의 움직임 벡터 및 움직임 벡터 후보 간의 움직임 벡터 차분(MVD: Motion Vector Difference)을 계산할 수 있고, MVD를 엔트로피 부호화할 수 있다. 또한, 부호화 장치(100)는 움직임 벡터 후보 색인을 엔트로피 부호화하여 비트스트림을 생성할 수 있다. 움직임 벡터 후보 색인은 움직임 벡터 후보 리스트에 포함된 움직임 벡터 후보 중에서 선택된 최적의 움직임 벡터 후보를 지시할 수 있다. 복호화 장치(200)는 움직임 벡터 후보 색인을 비트스트림으로부터 엔트로피 복호화하고, 엔트로피 복호화된 움직임 벡터 후보 색인을 이용하여 움직임 벡터 후보 리스트에 포함된 움직임 벡터 후보 중에서 복호화 대상 블록의 움직임 벡터 후보를 선택할 수 있다. 또한, 복호화 장치(200)는 엔트로피 복호화된 MVD 및 움직임 벡터 후보의 합을 통해 복호화 대상 블록의 움직임 벡터를 도출할 수 있다.

[0208] 한편, 부호화 장치(100)는 계산된 MVD의 해상도 정보를 엔트로피 부호화할 수 있다. 복호화 장치(200)는 MVD 해상도 정보를 이용하여 엔트로피 복호화된 MVD의 해상도를 조정할 수 있다.

[0209] 한편, 부호화 장치(100)는 어파인 모델에 기반하여 현재 블록의 움직임 벡터 및 움직임 벡터 후보 간의 움직임 벡터 차분(MVD: Motion Vector Difference)을 계산할 수 있고, MVD를 엔트로피 부호화할 수 있다. 복호화 장치(200)는 엔트로피 복호화된 MVD 및 어파인 제어 움직임 벡터 후보의 합을 통해 복호화 대상 블록의 어파인 제어 움직임 벡터를 도출하여 서브 블록 단위로 움직임 벡터를 유도할 수 있다.

[0210] 비트스트림은 참조 영상을 지시하는 참조 영상 색인 등을 포함할 수 있다. 참조 영상 색인은 엔트로피 부호화되어 비트스트림을 통해 부호화 장치(100)로부터 복호화 장치(200)로 시그널링될 수 있다. 복호화 장치(200)는 유도된 움직임 벡터와 참조 영상 색인 정보에 기반하여 복호화 대상 블록에 대한 예측 블록을 생성할 수 있다.

[0211] 움직임 정보의 도출 방식의 다른 예로, 머지 모드가 있다. 머지 모드란 복수의 블록들에 대한 움직임의 병합을

의미할 수 있다. 머지 모드는 현재 블록의 움직임 정보를 주변 블록의 움직임 정보로부터 유도하는 모드를 의미할 수 있다. 머지 모드가 적용되는 경우, 복원된 주변 블록의 움직임 정보 및/또는 콜 블록의 움직임 정보를 이용하여 머지 후보 리스트(merge candidate list)를 생성할 수 있다. 움직임 정보는 1) 움직임 벡터, 2) 참조 영상 색인, 및 3) 화면 간 예측 지시자 중 적어도 하나를 포함할 수 있다. 예측 지시자는 단방향 (L0 예측, L1 예측) 또는 쌍방향일 수 있다.

[0212] 머지 후보 리스트는 움직임 정보들이 저장된 리스트를 나타낼 수 있다. 머지 후보 리스트에 저장되는 움직임 정보는, 현재 블록에 인접한 주변 블록의 움직임 정보(공간적 머지 후보(spatial merge candidate)) 및 참조 영상에서 현재 블록에 대응되는(collocated) 블록의 움직임 정보(시간적 머지 후보(temporal merge candidate)), 이미 머지 후보 리스트에 존재하는 움직임 정보들의 조합에 의해 생성된 새로운 움직임 정보, 현재 블록 이전에 부호화/복호화된 블록의 움직임 정보(히스토리 기반 머지 후보(history-based merge candidate)) 및 제로 머지 후보 중 적어도 하나일 수 있다.

[0213] 부호화 장치(100)는 머지 플래그(merge flag) 및 머지 색인(merge index) 중 적어도 하나를 엔트로피 부호화하여 비트스트림을 생성한 후 복호화 장치(200)로 시그널링할 수 있다. 머지 플래그는 블록 별로 머지 모드를 수행할지 여부를 나타내는 정보일 수 있고, 머지 색인은 현재 블록에 인접한 주변 블록들 중 어떤 블록과 머지를 할 것인가에 대한 정보일 수 있다. 예를 들면, 현재 블록의 주변 블록들은 현재 블록의 좌측 인접 블록, 상단 인접 블록 및 시간적 인접 블록 중 적어도 하나를 포함할 수 있다.

[0214] 한편, 부호화 장치(100)는 머지 후보의 움직임 정보 중 움직임 벡터를 보정하기 위한 보정 정보를 엔트로피 부호화하여 복호화 장치(200)로 시그널링할 수 있다. 복호화 장치(200)는 머지 색인에 의해 선택된 머지 후보의 움직임 벡터를 보정 정보에 기초하여 보정할 수 있다. 여기서, 보정 정보는 보정 여부 정보, 보정 방향 정보 및 보정 크기 정보 중 적어도 하나를 포함할 수 있다. 위와 같이, 시그널링되는 보정 정보를 기초로 머지 후보의 움직임 벡터를 보정하는 예측 모드를 움직임 벡터 차분을 가진 머지 모드로 칭할 수 있다.

[0215] 스킵 모드는 주변 블록의 움직임 정보를 그대로 현재 블록에 적용하는 모드일 수 있다. 스킵 모드가 사용되는 경우, 부호화 장치(100)는 어떤 블록의 움직임 정보를 현재 블록의 움직임 정보로서 이용할 것인지에 대한 정보를 엔트로피 부호화하여 비트스트림을 통해 복호화 장치(200)에 시그널링할 수 있다. 이때, 부호화 장치(100)는 움직임 벡터 차분 정보, 부호화 블록 플래그 및 변환 계수 레벨(양자화된 레벨) 중 적어도 하나에 관한 구문 요소를 복호화 장치(200)에 시그널링하지 않을 수 있다.

[0216] 서브 블록 머지 모드(subblock merge mode)는, 부호화 블록(CU)의 서브 블록 단위로 움직임 정보를 유도하는 모드를 의미할 수 있다. 서브 블록 머지 모드가 적용되는 경우, 참조 영상에서 현재 서브 블록에 대응되는 (collocated) 서브 블록의 움직임 정보 (서브블록 기반 시간적 머지 후보(Sub-block based temporal merge candidate)) 및/또는 어파인 제어 포인트 움직임 벡터 머지 후보(affine ontrol point motion vector merge candidate)를 이용하여 서브 블록 머지 후보 리스트(subblock merge candidate list)가 생성될 수 있다.

[0217] 삼각 분할 모드(triangle partition mode)는, 현재 블록을 대각선 방향으로 분할하여 각각의 움직임 정보를 유도하고, 유도된 각각의 움직임 정보를 이용하여 각각의 예측 샘플을 유도하고, 유도된 각각의 예측 샘플을 가중합하여 현재 블록의 예측 샘플을 유도하는 모드를 의미할 수 있다.

[0218] 인터 인트라 결합 예측 모드는, 화면 간 예측으로 생성된 예측 샘플과 화면 내 예측으로 생성된 예측 샘플을 가중합하여 현재 블록의 예측 샘플을 유도하는 모드를 의미할 수 있다.

[0219] 복호화 장치(200)는 도출된 움직임 정보를 자체적으로 보정할 수 있다. 복호화 장치(200)는 도출된 움직임 정보가 지시하는 참조 블록을 기준으로 기정의된 구역 탐색하여 최소의 SAD를 갖는 움직임 정보를 보정된 움직임 정보로 유도할 수 있다.

[0220] 복호화 장치(200)는 광학적 흐름(Optical Flow)을 이용하여 화면 간 예측을 통해 유도된 예측 샘플을 보상할 수 있다.

[0221] 도 6은 변환 및 양자화의 과정을 설명하기 위한 도면이다.

[0222] 도 6에 도시된 바와 같이 잔여 신호에 변환 및/또는 양자화 과정을 수행하여 양자화된 레벨이 생성될 수 있다. 상기 잔여 신호는 원본 블록과 예측 블록(화면 내 예측 블록 혹은 화면 간 예측 블록) 간의 차분으로 생성될 수 있다. 여기에서, 예측 블록은 화면 내 예측 또는 화면 간 예측에 의해 생성된 블록일 수 있다. 여기서, 변환은 1차 변환 및 2차 변환 중 적어도 하나를 포함할 수 있다. 잔여 신호에 대해서 1차 변환을 수행하면 변환 계수가

생성될 수 있고, 변환 계수에 2차 변환을 수행하여 2차 변환 계수를 생성할 수 있다.

[0223] 1차 변환(Primary Transform)은 기-정의된 복수의 변환 방법 중 적어도 하나를 이용하여 수행될 수 있다. 일예로, 기-정의된 복수의 변환 방법은 DCT(Discrete Cosine Transform), DST(Discrete Sine Transform) 또는 KLT(Karhunen-Loève Transform) 기반 변환 등을 포함할 수 있다. 1차 변환이 수행 후 생성되는 변환 계수에 2차 변환(Secondary Transform)을 수행할 수 있다. 1차 변환 및/또는 2차 변환시에 적용되는 변환 방법은 현재 블록 및/또는 주변 블록의 부호화 파라미터 중 적어도 하나에 따라 결정될 수 있다. 또는 변환 방법을 지시하는 변환 정보가 시그널링될 수도 있다. DCT 기반 변환은 예컨대, DCT2, DCT-8 등을 포함할 수 있다. DST 기반 변환은 예컨대, DST-7을 포함할 수 있다.

[0224] 1차 변환 및/또는 2차 변환이 수행된 결과 또는 잔여 신호에 양자화를 수행하여 양자화된 레벨을 생성할 수 있다. 양자화된 레벨은 화면 내 예측 모드 또는 블록 크기/형태 중 적어도 하나를 기준으로 우상단 대각 스캔, 수직 스캔, 수평 스캔 중 적어도 하나에 따라 스캐닝(scanning) 될 수 있다. 예를 들어, 우상단(up-right) 대각 스캐닝을 이용하여 블록의 계수를 스캔함으로써 1차원 벡터 형태로 변경시킬 수 있다. 변환 블록의 크기 및/또는 화면 내 예측 모드에 따라 우상단 대각 스캔 대신 2차원의 블록 형태 계수를 열 방향으로 스캔하는 수직 스캔, 2차원의 블록 형태 계수를 행 방향으로 스캔하는 수평 스캔이 사용될 수도 있다. 스캐닝된 양자화된 레벨은 엔트로피 부호화되어 비트스트림에 포함될 수 있다.

[0225] 복호화기에서는 비트스트림을 엔트로피 복호화하여 양자화된 레벨을 생성할 수 있다. 양자화된 레벨은 역 스캐닝(Inverse Scanning)되어 2차원의 블록 형태로 정렬될 수 있다. 이때, 역 스캐닝의 방법으로 우상단 대각 스캔, 수직 스캔, 수평 스캔 중 적어도 하나가 수행될 수 있다.

[0226] 양자화된 레벨에 역양자화를 수행할 수 있고, 2차 역변환 수행 여부에 따라 2차 역변환을 수행할 수 있고, 2차 역변환이 수행된 결과에 1차 역변환 수행 여부에 따라 1차 역변환을 수행하여 복원된 잔여 신호가 생성될 수 있다.

[0227] 화면 내 예측 또는 화면 간 예측을 통해 복원된 휘도 성분에 대해 인루프 필터링 전에 동적 범위(dynamic range)의 역매핑(inverse mapping)이 수행될 수 있다. 동적 범위는 16개의 균등한 조각(piece)으로 분할될 수 있고, 각 조각에 대한 매핑 함수가 시그널링될 수 있다. 상기 매핑 함수는 슬라이스 레벨 또는 타일 그룹 레벨에서 시그널링될 수 있다. 상기 역매핑을 수행하기 위한 역매핑 함수는 상기 매핑 함수에 기초하여 유도될 수 있다. 인루프 필터링, 참조 픽처의 저장 및 움직임 보상은 역매핑된 영역에서 수행되며, 화면 간 예측을 통해 생성된 예측 블록은 상기 매핑 함수를 이용한 매핑에 의해 매핑된 영역으로 전환된 후, 복원 블록의 생성에 이용될 수 있다. 그러나, 화면 내 예측은 매핑된 영역에서 수행되므로, 화면 내 예측에 의해 생성된 예측 블록은 매핑/역매핑 없이, 복원 블록의 생성에 이용될 수 있다.

[0228] 현재 블록이 색차 성분의 잔차 블록인 경우, 매핑된 영역의 색차 성분에 대해 스케일링을 수행함으로써 상기 잔차 블록은 역매핑된 영역으로 전환될 수 있다. 상기 스케일링의 가용 여부는 슬라이스 레벨 또는 타일 그룹 레벨에서 시그널링될 수 있다. 상기 스케일링은 루마 성분에 대한 상기 매핑이 가용하고 휘도 성분의 분할과 색차 성분의 분할이 동일한 트리 구조를 따르는 경우에만 상기 스케일링이 적용될 수 있다. 상기 스케일링은 상기 색차 블록에 대응하는 휘도 예측 블록의 샘플값의 평균에 기초하여 수행될 수 있다. 이 때, 현재 블록이 화면 간 예측을 사용하는 경우, 상기 휘도 예측 블록은 매핑된 휘도 예측 블록을 의미할 수 있다. 휘도 예측 블록의 샘플값의 평균이 속하는 조각(piece)의 인덱스를 이용하여, 룩업테이블을 참조함으로써, 상기 스케일링에 필요한 값을 유도할 수 있다. 최종적으로 상기 유도된 값을 이용하여 상기 잔차 블록을 스케일링함으로써, 상기 잔차 블록은 역매핑된 영역으로 전환될 수 있다. 이 후의 색차 성분 블록의 복원, 화면 내 예측, 화면 간 예측, 인루프 필터링 및 참조 픽처의 저장은 역매핑된 영역에서 수행될 수 있다.

[0229] 상기 휘도 성분 및 색차 성분의 매핑/역매핑이 가용한지 여부를 나타내는 정보는 시퀀스 파라미터 셋을 통해 시그널링될 수 있다.

[0230] 현재 블록의 예측 블록은 현재 블록과 현재 픽처 내 참조 블록 사이의 위치 이동(displacement)을 나타내는 블록 벡터에 기초하여 생성될 수 있다. 이와 같이, 현재 픽처를 참조하여 예측 블록을 생성하는 예측 모드를 화면 내 블록 카피(Intra Block Copy, IBC) 모드라고 명명할 수 있다. IBC 모드는 MxN(M<=64, N<=64) 부호화 유닛에 적용될 수 있다. IBC 모드는 스킵 모드, 머지 모드, AMVP 모드 등을 포함할 수 있다. 스킵 모드 또는 머지 모드의 경우, 머지 후보 리스트가 구성되고, 머지 인덱스가 시그널링되어 하나의 머지 후보가 특정될 수 있다. 상기 특정된 머지 후보의 블록 벡터가 현재 블록의 블록 벡터로서 이용될 수 있다. 머지 후보 리스트는 공간적 후보,

히스토리에 기반한 후보, 두개 후보의 평균에 기반한 후보 또는 제로 머지 후보 등 적어도 하나 이상을 포함할 수 있다. AMVP 모드의 경우, 차분 블록 벡터가 시그널링될 수 있다. 또한, 예측 블록 벡터는 현재 블록의 좌측 이웃 블록 및 상단 이웃 블록으로부터 유도될 수 있다. 어느 이웃 블록을 이용할지에 관한 인덱스는 시그널링될 수 있다. IBC 모드의 예측 블록은 현재 CTU 또는 좌측 CTU에 포함되고, 기 복원된 영역내의 블록으로 한정될 수 있다. 예를 들어, 현재 블록의 예측 블록은, 현재 블록이 속한 64x64 블록보다 부호화/복호화 순서상 앞선 3개의 64x64 블록 영역내에 위치하도록 블록 벡터의 값이 제한될 수 있다. 이와 같이 블록 벡터의 값을 제한함으로써, IBC 모드 구현에 따른 메모리 소비와 장치의 복잡도를 경감할 수 있다.

[0231] 본 발명에 따르면, 현재 블록에 대한 분할 구조를 선택하거나 또는 예측 모드를 결정함에 있어, 1. CTU(Coding Tree Unit) 수준에서 유도하는 방법, 2. Region 수준에서 유도하는 방법 및 3. 예측 모드 수준에서 유도하는 방법 중 적어도 하나가 사용될 수 있다.

[0232] 1. CTU 수준에서 유도하는 방법

[0233] 현재 CTU에 포함된 휘도(luma) 신호와 색차(chroma) 신호는 공통의 트리 구조를 이용하여 분할되거나, 별도의 두 개의 트리 구조를 이용하여 분할될 수 있다.

[0234] 현재 CTU에 대한 분할 트리 구조의 모든 leaf node(혹은 부호화 블록(CU))들은 화면간 예측으로 유도될 수 있다.

[0235] 또는, 현재 CTU에 대한 분할 트리 구조의 모든 leaf node(혹은 CU)들은 화면내 예측으로 유도될 수 있다.

[0236] 또는, 현재 CTU에 대한 분할 트리 구조의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0237] 현재 CTU의 luma 신호와 chroma 신호는 소정 크기 블록까지는 공통의 트리 구조로 분할되고, 이후에는 별도의 두 개 분할 트리 구조로 분할될 수 있다.

[0238] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기 블록까지 공통의 트리 구조 또는 별도의 두 개 트리 구조로 분할될 수 있다. 이후 luma 신호와 chroma 신호는 별도의 두 개 분할 트리 구조로 분할될 수 있다.

[0239] 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)들은 화면간 예측될 수 있다.

[0240] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0241] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측 또는 화면간 예측될 수 있다.

[0242] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0243] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0244] 현재 CTU의 luma 신호와 chroma 신호는 소정의 블록 크기까지 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0245] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0246] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정의 블록 크기까지 공통의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0247] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정의 블록 크기까지 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0248] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0249] 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0250] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기 블록부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0251] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0252] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0253] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기 블록들부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0254] 현재 CTU의 트리 구조 혹은 예측 모드는 ctu_separate_coding_trees_flag, ctu_pred_mode_flag, ctu_pred_mode_and_separate_coding_trees_flag 중 적어도 하나에 기초하여 부호화되거나 복호화될 수 있다.

[0255] 예컨대, ctu_separate_coding_trees_flag 및 ctu_pred_mode_and_separate_coding_trees_flag 중 적어도 하나를 이용하여 현재 CTU의 luma 신호와 chroma 신호의 분할 트리 구조가 공통의 트리 구조인지 별도의 두 개의 트리 구조인지가 부호화/복호화될 수 있다.

[0256] 예컨대, ctu_pred_mode_flag 및 ctu_pred_mode_and_separate_coding_trees_flag 중 적어도 하나를 이용하여 현재 CTU 내 leaf node(혹은 CU)가 부호화/복호화될 수 있다. 예컨대, 현재 CTU 내 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다. 또는, 현재 CTU 내 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다. 또는, 현재 CTU 내 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0257] 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되거나, 별도의 두 개의 트리 구조로 분할될 수 있다. 다시 말해, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조(structure)의 root가 되거나, 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0258] 예컨대, ctu_separate_coding_trees_flag가 제1값(예컨대, 0)인 경우, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 즉, 현재 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할됨을 나타낼 수 있다.

[0259] 도 8은 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0260] 도 8에 도시된 바와 같이, 현재 CTU의 luma 신호와 chroma 신호는 동일한 분할 트리 구조를 가질 수 있다. 이 때, ctu_separate_coding_trees_flag는 제1값(예컨대, 0)으로 시그널링될 수 있다.

[0261] 도 9는 luma 신호와 chroma 신호가 별도의 두 개의 트리 구조로 분할되는 예를 도시한 도면이다.

[0262] 도 9에 도시된 바와 같이, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 이 때, ctu_separate_coding_trees_flag는 제2값(예컨대, 1)으로 시그널링될 수 있다. 즉, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할될 수 있다.

[0263] 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다

[0264] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할될 수 있다. 여기서, 현재 CTU 부터 소정 크기의 블록까지는 공통의 트리 구조를 사용하여 luma 신호와 chroma 신호를 분할할 수 있다.

[0265] 즉, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0266] 또는, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0267] ctu_separate_coding_trees_flag가 제1값인 경우, CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할됨을 나타낼 수 있다.

[0268] ctu_separate_coding_trees_flag가 제2값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록부터 별도의 두 개의 트리 구조로 분할됨을 시그널링할 수 있다.

[0269] ctu_separate_coding_trees_flag가 시그널링될 때, 상기 소정 크기는 부호화 파라미터, 픽처 정보, 슬라이스 정보, 양자화 파리미터(QP), 부호화 블록 플래그(CBF), 블록 크기, 블록 깊이, 블록 형태, 엔트로피 부호화 방법, 주변 블록의 화면내 예측 모드, 시간적 계층 수준 중 적어도 하나에 기반하여 결정될 수 있다.

[0270] ctu_separate_coding_trees_flag가 제1값인 경우, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root임을 시그널링할 수 있다. 즉, 현재 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할됨을 나타낼 수 있다.

[0271] 도 10은 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0272] 도 10에 도시된 바와 같이, 현재 CTU의 luma 신호와 chroma 신호는 동일한 분할 트리 구조를 가질 수 있다. 이 때, ctu_separate_coding_trees_flag는 제1값(예컨대, 0)으로 시그널링될 수 있다.

[0273] 도 11은 luma 신호와 chroma 신호가 소정 크기의 블록부터 별도의 두 개의 트리 구조로 분할되는 예를 도시한 도면이다.

[0274] 도 11에 도시된 바와 같이, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 그 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 이 때, ctu_separate_coding_trees_flag는 제2값(예컨대, 1)으로 시그널링될 수 있다. 즉, ctu_separate_coding_trees_flag가 제2값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할됨을 나타낼 수 있다.

[0275] 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할될 수 있다.

[0276] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 별도의 두 개 트리 구조로 분할될 수 있다. 이 때, 소정 크기의 블록부터 CU(트리의 마지막 노드)까지, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할될 수 있다.

[0277] 즉, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할되고, 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할될 수 있다.

[0278] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할될 수 있다.

[0279] 즉, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0280] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0281] ctu_separate_coding_trees_flag가 제1값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할됨을 나타낼 수 있다.

[0282] ctu_separate_coding_trees_flag가 제2값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 별도의 두 개 트리 구조로 분할됨을 시그널링할 수 있다.

[0283] ctu_separate_coding_trees_flag가 시그널링될 때, 상기 소정 크기는 부호화 파라미터, 픽처 정보, 슬라이스 정보, 양자화 파리미터(QP), 부호화 블록 플래그(CBF), 블록 크기, 블록 깊이, 블록 형태, 엔트로피 부호화 방법, 주변 블록의 화면내 예측 모드, 시간적 계층 수준 중 적어도 하나에 기반하여 결정될 수 있다.

[0284] ctu_separate_coding_trees_flag가 제1값인 경우, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0285] 도 12는 현재 CTU의 luma 신호와 chroma 신호가 소정 크기의 블록까지 공통의 트리 구조로 분할되고, 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되는 예를 도시한 도면이다.

[0286] 이 경우, ctu_separate_coding_trees_flag는 제1값을 가질 수 있다.

[0287] 도 13은 현재 CTU의 luma 신호와 chroma 신호가 별도의 두 개 트리 구조로 분할되는 예를 도시한 도면이다.

[0288] 이 경우, ctu_separate_coding_trees_flag는 제2값을 가질 수 있다. 이 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0289] 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다. 또는, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다. 또는, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0290] ctu_pred_mode_flag가 제1값인 경우, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다. ctu_pred_mode_flag가 제2값인 경우, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다. ctu_pred_mode_flag가 제3값인 경우, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 시그널링할 수 있다.

[0291] 본 개시에 있어서, 화면간 예측은 inter 모드, skip 모드, merge 모드 중 적어도 하나에 따른 예측을 의미할 수 있다. 또한, 화면내 예측은 방향성/비방향성 모드에 따른 화면내 예측, 색 성분간 예측, IBC 모드, 팔레트(pallete) 모드 중 적어도 하나에 따른 예측을 의미할 수 있다.

[0292] 도 14는 현재 CTU의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

[0293] 이 경우, ctu_pred_mode_flag는 제1값으로 부호화/복호화될 수 있다.

[0294] ctu_pred_mode_flag가 제2값인 경우, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다. 즉, 현재 CTU의 leaf node(혹은 CU)의 일부는 화면내 예측되고 나머지는 화면간 예측될 수 있다. 다시 말해, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측 또는 화면간 예측 중 하나만을 사용하여 부호화되지 않았음을 의미할 수 있다.

[0295] 도 15는 현재 CTU의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

[0296] 이 경우, ctu_pred_mode_flag는 제2값으로 부호화/복호화될 수 있다.

[0297] ctu_pred_mode_flag가 제3값인 경우, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0298] 도 16은 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측되는 예를 도시한 도면이다.

[0299] 이 경우, ctu_pred_mode_flag는 제3값으로 부호화/복호화될 수 있다.

[0300] ctu_pred_mode_flag가 제1값 혹은 제3값인 경우, 현재 CTU의 모든 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않을 수 있다.

[0301] ctu_pred_mode_flag가 제1값인 경우, 현재 ctu의 모든 leaf node(혹은 CU)는 화면간 예측되고, ctu_pred_mode_flag가 제3값인 경우 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다. 따라서, 현재 CTU의 모든 leaf node(혹은 CU)들에 대하여 pred_mode_flag가 시그널링되지 않고, ctu_pred_mode_flag에 따라 현재 CTU의 모든 leaf node(혹은 CU)가 화면간 예측될지 화면내 예측될지를 결정할 수 있다.

[0302] ctu_pred_mode_flag가 제2값인 경우, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링될 수 있다.

[0303] pred_mode_flag가 제1값인 경우, 해당 leaf node(혹은 CU)는 화면간 예측으로 부호화/복호화됨을 나타낼 수 있다. pred_mode_flag가 제2값인 경우, 해당 leaf node(혹은 CU)는 화면내 예측으로 부호화/복호화됨을 나타낼 수 있다. 즉, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되고 pred_mode_flag에 따라 해당 leaf node(혹은 CU)가 화면간 예측될지 화면내 예측될지가 결정될 수 있다.

[0304] ctu_pred_mode_flag는 제1값, 제2값 및 제3값 중 적어도 하나를 포함할 수 있다. 상기 ctu_pred_mode_flag는 ctu_pred_mode_idx와 같이 인덱스 형식의 정보로 표현될 수도 있다.

[0305] 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되면서 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0306] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되면서 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0307] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되면서 현재 CTU의 leaf node(혹은 CU)는

화면내 예측되거나 화면간 예측될 수 있다.

[0308] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되면서 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0309] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되면서 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0310] 즉, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되면서 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0311] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되면서 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0312] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되면서 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0313] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 한 개 coding_tree 구문요소(syntax) 구조의 root가 되면서 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0314] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되면서 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0315] ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다.

[0316] ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 별개의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0317] ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0318] ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0319] ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 별개의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0320] ctu_pred_mode_and_separate_coding_trees_flag가 제1값, 제4값 또는 제5값을 가질 때, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않을 수 있다.

[0321] ctu_pred_mode_and_separate_coding_trees_flag가 제2값 또는 제3값을 가질 때, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링될 수 있다.

[0322] pred_mode_flag가 제1값인 경우, 현재 leaf node(혹은 CU)는 화면간 예측됨을 나타내고, pred_mode_flag가 제2값인 경우, 현재 leaf node(혹은 CU)가 화면내 예측됨을 나타낼 수 있다.

[0323] 상기 ctu_pred_mode_and_separate_coding_trees_flag는 제1값, 제2값, 제3값, 제4값 및 제5값 중 적어도 하나를 포함할 수 있다.

[0324] ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0325] 도 17은 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되고, CTU의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

[0326] ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 별개

의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)의 일부는 화면내 예측되고, 나머지는 화면간 예측될 수 있다. 즉, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측 혹은 화면간 예측 중 하나만을 사용하여 부호화되지 않았음을 나타낼 수 있다.

[0327] 도 18은 CTU의 luma 신호와 chroma 신호가 별도의 두 개 트리 구조로 분할되고, CTU의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

[0328] ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)의 일부는 화면내 예측되고, 나머지는 화면간 예측될 수 있다. 즉, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측 혹은 화면간 예측 중 하나만을 사용하여 부호화되지 않았음을 나타낼 수 있다.

[0329] 도 19는 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, CTU의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

[0330] ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0331] 도 20은 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, CTU의 모든 leaf node(혹은 CU)는 화면내 예측되는 예를 도시한 도면이다.

[0332] ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0333] 도 21은 CTU의 luma 신호와 chroma 신호가 별도의 두 개의 트리 구조로 분할되고, CTU의 모든 leaf node(혹은 CU)는 화면내 예측되는 예를 도시한 도면이다.

[0334] 상기 ctu_pred_mode_and_separate_coding_trees_flag는 ctu_pred_mode_and_separate_coding_trees_idx와 같이 인덱스 형식의 정보로 표현될 수도 있다.

[0335] 현재 CTU의 luma 신호와 chroma 신호는 소정의 블록 크기까지 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0336] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0337] 또는, 현재 CTU의 luma 신호와 chroma신호는 소정의 블록 크기까지 공통의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0338] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정의 블록 크기까지 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0339] 또는, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0340] 즉, 현재 CTU는 소정의 블록 크기까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0341] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0342] 또는, 현재 CTU는 소정의 블록 크기까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0343] 또는, 현재 CTU는 소정 블록 크기까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0344] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0345] 현재 CTU의 luma 신호와 chroma 신호는 소정 블록 크기까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 블록 크기부터 별도의 두 개의 트리 구조로 분할될 수 있다. 즉, 현재 CTU는 소정 블록 크기

까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 블록 크기의 블록들은 luma 신호와 chroma 신호에 대한 별도의 두 개의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0346] ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개의 coding_tree 구문요소(syntax) 구조의 root가 되며, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다.

[0347] ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0348] ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개의 coding_tree 구문요소(syntax) 구조의 root가 되며, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0349] ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대해 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되며, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0350] ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 CTU는 luma 신호와 chroma 신호에 대해 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0351] 현재 CTU의 luma 신호와 chroma 신호가 소정 크기 블록까지 공통의 한 개의 트리 구조로 분할되는 경우, luma 신호와 chroma 신호는 상기 소정 크기의 블록부터 별도의 두 개의 트리 구조로 분할될 수 있다. 즉, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대해 별도의 두 개의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0352] ctu_pred_mode_and_separate_coding_trees_flag가 시그널링되는 경우, 상기 소정 크기의 블록은 부호화 파라미터, 픽처 정보, 슬라이스 정보, 양자화 파리미터(QP), 부호화 블록 플래그(CBF), 블록 크기, 블록 깊이, 블록 형태, 엔트로피 부호화 방법, 주변 블록의 화면내 예측 모드 및 시간적 계층 수준 중 적어도 하나에 기반하여 결정될 수 있다.

[0353] ctu_pred_mode_and_separate_coding_trees_flag가 제1값, 제4값 또는 제5값을 가질 때, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않을 수 있다.

[0354] ctu_pred_mode_and_separate_coding_trees_flag가 제2값 또는 제3값을 가질 때, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링될 수 있다.

[0355] pred_mode_flag가 제1값인 경우, 현재 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다. pred_mode_flag가 제2값인 경우, 현재 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0356] ctu_pred_mode_and_separate_coding_trees_flag는 제1값, 제2값, 제3값, 제4값 및 제5값 중 적어도 하나를 포함할 수 있다.

[0357] ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기 블록까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 크기의 블록부터 별도의 두 개의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다.

[0358] 도 22는 ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0359] ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)의 일부는 화면내 예측되고 나머지는 화면간 예측됨을 나타낼 수 있다. 즉, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측과 화면간 예측 중 하나만을 사

용하여 부호화되지 않음을 나타낼 수 있다.

[0360] 도 23은 ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0361] ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)의 일부는 화면내 예측되고 나머지는 화면간 예측됨을 나타낼 수 있다. 즉, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측과 화면간 예측 중 하나만을 사용하여 부호화되지 않음을 나타낼 수 있다.

[0362] 도 24는 ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0363] ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0364] 도 25는 ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0365] ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0366] 도 26은 ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0367] 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0368] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0369] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0370] 또는, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0371] 또는, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0372] 즉, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0373] 또는, 현재 CTU의 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0374] 또는, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0375] 또는, 현재 CTU는 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0376] 또는, 현재 CTU의 소정 크기 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0377] 현재 CTU의 luma 신호와 chroma 신호가 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되는 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 한 개 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할될 수 있다.

[0378] 즉, 현재 CTU의 소정 크기의 블록이 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소

(syntax) 구조의 root가 되는 경우, 현재 CTU는 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0379] ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다.

[0380] ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 CTU가 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0381] ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0382] ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0383] ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 CTU가 소정 크기의 블록까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 상기 소정 크기의 블록은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0384] ctu_pred_mode_and_separate_coding_trees_flag가 시그널링될 때, 상기 소정의 블록 크기는 부호화 파라미터, 픽처 정보, 슬라이스 정보, 양자화 파리미터(QP), 부호화 블록 플래그(CBF), 블록 크기, 블록 깊이, 블록 형태, 엔트로피 부호화 방법, 주변 블록의 화면내 예측 모드 및 시간적 계층 수준 중 적어도 하나에 기반하여 결정될 수 있다.

[0385] ctu_pred_mode_and_separate_coding_trees_flag가 제1값, 제4값 또는 제5값을 가질 때, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않을 수 있다.

[0386] ctu_pred_mode_and_separate_coding_trees_flag가 제2값 또는 제3값을 가질 때, 현재 CTU의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링될 수 있다.

[0387] pred_mode_flag가 제1값인 경우, 현재 leaf node(혹은 CU)가 화면간 예측됨을 나타낼 수 있다. pred_mode_flag가 제2값인 경우, 현재 leaf node(혹은 CU)가 화면내 예측됨을 나타낼 수 있다.

[0388] ctu_pred_mode_and_separate_coding_trees_flag는 제1값, 제2값, 제3값, 제4 값 및 제5값 중 적어도 하나를 포함할 수 있다.

[0389] ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0390] 도 27은 ctu_pred_mode_and_separate_coding_trees_flag가 제1값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0391] ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 크기의 블록부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 일부는 화면내 예측되고 나머지는 화면간 예측됨을 나타낼 수 있다. 즉, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측과 화면간 예측 중 하나만을 사용하여 부호화되지 않음을 나타낼 수 있다.

[0392] 도 28은 ctu_pred_mode_and_separate_coding_trees_flag가 제2값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0393] ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통

의 트리 구조로 분할되고, 현재 CTU의 leaf node(혹은 CU)의 일부는 화면내 예측되고 나머지는 화면간 예측됨을 나타낼 수 있다. 즉, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측과 화면간 예측 중 하나만을 사용하여 부호화되지 않음을 나타낼 수 있다.

[0394] 도 29는 ctu_pred_mode_and_separate_coding_trees_flag가 제3값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0395] ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0396] 도 30은 ctu_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0397] ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 CTU의 luma 신호와 chroma 신호는 소정 크기의 블록까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 상기 소정 블록의 크기부터 별도의 두 개 트리 구조로 분할되고, 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측됨을 나타낼 수 있다.

[0398] 도 31은 ctu_pred_mode_and_separate_coding_trees_flag가 제5값인 경우의 CTU의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0399] 2. Region 수준에서 유도하는 방법

[0400] 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다. 또는, 현재 region의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할될 수 있다.

[0401] 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다. 또는, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다. 또는, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0402] CTU의 luma 신호와 chroma 신호는 소정의 블록 크기를 갖는 현재 region까지 공통의 트리 구조로 분할되고, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되거나 별도의 두 개 트리 구조로 분할될 수 있다.

[0403] Luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0404] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 별도의 두 개 트리 구조로 분할되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0405] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0406] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 별도의 두 개 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0407] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0408] 현재 region에 대한 분할 구조 및/또는 예측 모드를 유도 및/또는 선택함에 있어서, region_separate_coding_trees_flag, region_pred_mode_flag, region_pred_mode_and_separate_coding_trees_flag 중 적어도 하나가 이용될 수 있다.

[0409] 현재 region의 분할 구조를 유도 혹은 선택함에 있어, region_separate_coding_trees_flag 및 region_pred_mode_and_separate_coding_trees_flag 중 적어도 하나에 따라 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는지 별도의 두 개 트리 구조로 분할되는지 나타낼 수 있다.

[0410] 현재 region의 예측 모드를 유도 혹은 선택함에 있어, region_pred_mode_flag 및 region_pred_mode_and_separate_coding_trees_flag 중 적어도 하나에 따라 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측될지 화면내 예측될지 또는 현재 region의 leaf node(혹은 CU)가 화면간 예측되거나 화면내 예측될지 나타낼 수 있다.

[0411] 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되거나, 별도의 두 개 트리 구조로 분할될 수 있다.

[0412] region_separate_coding_trees_flag가 제1값인 경우, 현재 region이 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 됨을 나타낼 수 있다.

[0413] region_separate_coding_trees_flag가 제2값인 경우, 현재 region이 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 됨을 나타낼 수 있다.

[0414] region_separate_coding_trees_flag가 제1값인 경우, 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다.

[0415] 도 32는 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0416] region_separate_coding_trees_flag가 제2값인 경우, 현재 region의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할될 수 있다.

[0417] 도 33은 현재 region의 luma 신호와 chroma 신호가 별도의 두 개 트리 구조로 분할되는 예를 도시한 도면이다.

[0418] 상기에서 현재 CTU에서 소정 크기의 블록인 region까지는 공통의 트리 구조가 사용될 수 있다. 즉, 현재 CTU는 소정 크기의 블록인 region까지 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0419] 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다. 또는, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다. 또는, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다. 현재 region의 모든 leaf node(혹은 CU)에 대한 예측 모드를 나타내는 region 예측 정보는 시그널링되거나 부호화기/복호화기에서 유도될 수 있다. 이하의 설명에서 region_pred_mode_flag는 상기 region 예측 정보의 일 예이다.

[0420] region_pred_mode_flag가 제1값인 경우, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다.

[0421] region_pred_mode_flag가 제2값인 경우, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0422] region_pred_mode_flag가 제3값인 경우, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0423] region_pred_mode_flag가 제1값 또는 제3값을 가질 때, 현재 region의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않을 수 있다. 즉, 현재 region의 모든 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않고 region_pred_mode_flag에 따라 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측되는지 화면내 예측되는지 결정될 수 있다.

[0424] region_pred_mode_flag가 제2값을 가질 때, 현재 region의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링될 수 있다. 즉, 현재 region의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되고 pred_mode_flag에 따라 현재 region의 leaf node(혹은 CU)가 화면간 예측되는지 화면내 예측되는지 결정될 수 있다

[0425] pred_mode_flag가 제1값인 경우, 해당 leaf node(혹은 CU)가 화면간 예측됨을 나타낼 수 있다.

[0426] pred_mode_flag가 제2값인 경우, 해당 leaf node(혹은 CU)가 화면내 예측됨을 나타낼 수 있다.

[0427] region_pred_mode_flag가 제1값인 경우, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0428] 도 34는 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

[0429] region_pred_mode_flag가 제2값인 경우, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다. 즉, 현재 region의 leaf node(혹은 CU)의 일부는 화면내 예측되고, 나머지는 화면간 예측될 수 있다.

[0430] 도 35는 현재 region의 leaf node(혹은 CU)가 화면내 예측되거나 화면간 예측되는 예를 도시한 도면이다.

[0431] region_pred_mode_flag가 제3값인 경우, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0432] 도 36은 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측되는 예를 도시한 도면이다.

[0433] 한편, 본 명세서의 region_pred_mode_flag는 region_pred_mode_idx와 같이 인덱스 형식의 정보로 표현될 수도 있다.

[0434] Luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0435] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 별도의 두 개 트리 구조로 분할되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0436] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0437] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 별도의 두 개 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0438] 또는, luma 신호와 chroma 신호는 현재 region부터 leaf node(혹은 CU)까지 공통의 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0439] 다시 말해, 현재 region은 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측으로 유도될 수 있다.

[0440] 또는, 현재 region은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0441] 또는, 현재 region은 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측될 수 있다.

[0442] 또는, 현재 region은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0443] 또는, 현재 region은 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0444] 상기에서 CTU의 luma 신호와 chroma 신호는 현재 region까지 공통의 트리 구조로 분할될 수 있다. 다시 말해, CTU는 소정의 블록 크기를 갖는 현재 region까지 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0445] region_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 region이 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측됨을 나타낼 수 있다.

[0446] region_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 region이 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0447] region_pred_mode_and_separate_coding_trees_flag가 제3값인 경우, 현재 region이 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되거나 화면간 예측됨을 나타낼 수 있다.

[0448] region_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 region이 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측됨을 나타낼 수 있다.

[0449] region_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 region은 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측됨을 시그널링할 수 있다.

[0450] region_pred_mode_and_separate_coding_trees_flag가 제1값, 제4값 또는 제5값을 가질 때, 현재 region의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링되지 않을 수 있다.

[0451] region_pred_mode_and_separate_coding_trees_flag가 제2값 또는 제3값을 가질 때, 현재 region의 leaf node(혹은 CU)에 대하여 pred_mode_flag가 시그널링될 수 있다.

[0452] pred_mode_flag가 제1값인 경우, 해당 leaf node(혹은 CU)가 화면간 예측됨을 나타낼 수 있다.

[0453] pred_mode_flag가 제2값인 경우, 해당 leaf node(혹은 CU)가 화면내 예측됨을 나타낼 수 있다.

[0454] region_pred_mode_and_separate_coding_trees_flag가 제1값인 경우, 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다.

[0455] 도 37은 region_pred_mode_and_separate_coding_trees_flag가 제1값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0456] region_pred_mode_and_separate_coding_trees_flag가 제2값인 경우, 현재 region의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 region의 leaf node(혹은 CU)의 일부는 화면내 예측되고 나머지는 화면간 예측됨을 나타낼 수 있다. 즉, 현재 region의 모든 leaf node(혹은 CU)가 화면내 예측 및 화면간 예측 중 하나만을 사용하여 부호화되지 않았음을 나타낼 수 있다.

[0457] 도 38은 region_pred_mode_and_separate_coding_trees_flag가 제2값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0458] region_pred_mode_and_separate_coding_trees_flag가 제3의 값인 경우, 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 region의 leaf node(혹은 CU)는 화면내 예측되고 나머지는 화면간 예측됨을 나타낼 수 있다. 즉, 현재 region의 모든 leaf node(혹은 CU)가 화면내 예측 및 화면간 예측 중 하나만을 사용하여 부호화되지 않았음을 나타낼 수 있다.

[0459] 도 39는 region_pred_mode_and_separate_coding_trees_flag가 제3값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0460] region_pred_mode_and_separate_coding_trees_flag가 제4값인 경우, 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0461] 도 40은 region_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0462] region_pred_mode_and_separate_coding_trees_flag가 제5값인 경우, 현재 region의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할되고, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다.

[0463] 도 41은 region_pred_mode_and_separate_coding_trees_flag가 제4값인 경우의 region의 분할 및 CU의 예측 모드를 예시한 도면이다.

[0464] region_pred_mode_and_separate_coding_trees_flag는 제1값, 제2값, 제3값, 제4값 및 제5값 중 적어도 하나를 포함할 수 있다

[0465] region_pred_mode_and_separate_coding_trees_flag는 region_pred_mode_and_separate_coding_trees_idx와 같이 인덱스 형식의 정보로 표현될 수도 있다

[0466] 3. 예측 모드 수준에서 유도하는 방법

[0467] 현재 CTU의 모든 leaf node(혹은 CU)가 화면간 예측되는 경우, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다. 즉, 현재 CTU가 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0468] 현재 CTU의 모든 leaf node(혹은 CU)가 화면내 예측되는 경우, 현재 CTU의 luma 신호와 chroma 신호는 별도의 두 개 트리 구조로 분할될 수 있다. 즉, 현재 CTU가 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0469] 현재 region의 모든 leaf node(혹은 CU)가 화면간 예측되는 경우, 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다. 즉, 현재 region이 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0470] 현재 region의 모든 leaf node(혹은 CU)가 화면내 예측되는 경우, 현재 region의 luma 신호와 chroma 신호는

별도의 두 개 트리 구조로 분할될 수 있다. 즉, 현재 region이 luma 신호와 chroma 신호에 대한 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다.

[0471] 현재 region의 모든 leaf node(혹은 CU)가 화면내 예측되는지의 여부는 예컨대, 전술한 region 예측 정보에 기초하여 판단될 수 있다.

[0472] ctu_pred_mode_flag 지시자에 따라 현재 CTU의 분할 구조를 유도하거나 선택할 수 있다.

[0473] region_pred_mode_flag 지시자에 따라 현재 region의 분할 구조를 유도하거나 선택할 수 있다.

[0474] ctu_pred_mode_flag가 제1값을 가질 때, 현재 CTU의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다. 이 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 즉, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다.

[0475] 도 42는 ctu_pred_mode_flag가 제1값을 가질 때, 현재 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0476] ctu_pred_mode_flag가 제3값을 가질 때, 현재 CTU의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다. 이 경우, 현재 CTU는 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 즉, 현재 CTU의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다.

[0477] 도 43은 ctu_pred_mode_flag가 제3값을 가질 때, 현재 CTU의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0478] region_pred_mode_flag가 제1값을 가질 때, 현재 region의 모든 leaf node(혹은 CU)는 화면간 예측될 수 있다. 이 경우, 현재 region은 luma 신호와 chroma 신호에 대한 공통의 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 즉, 현재 region의 luma 신호와 chroma 신호는 공통의 트리 구조로 분할될 수 있다.

[0479] 도 44는 region_pred_mode_flag가 제1값을 가질 때, 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0480] region_pred_mode_flag가 제3값을 가질 때, 현재 region의 모든 leaf node(혹은 CU)는 화면내 예측될 수 있다. 이 경우, 현재 region은 luma 신호와 chroma 신호에 대해 별도의 두 개 coding_tree 구문요소(syntax) 구조의 root가 될 수 있다. 즉, 현재 region의 luma 신호와 chroma 신호는 별도의 두 개의 트리 구조로 분할될 수 있다.

[0481] 도 45는 region_pred_mode_flag가 제3값을 가질 때, 현재 region의 luma 신호와 chroma 신호가 공통의 트리 구조로 분할되는 예를 도시한 도면이다.

[0482] 본 개시에 있어서, region은 소정 크기의 블록(예컨대, CU)을 의미할 수 있다. region은 복수의 CU로 분할될 수도 있고, 더 이상 분할되지 않는 하나의 CU를 의미할 수도 있다. 또는 region은 CTU의 분할 트리 구조에서 하나의 노드 및 그의 하위 노드에 대응하는 영역을 의미할 수 있다. 상기 하나의 노드는 CTU에 해당하는 root node, CU에 해당하는 leaf node 또는 중간 노드일 수 있다.

[0483] 도 46은 본 개시에 따른 region을 설명하기 위한 도면이다.

[0484] 도 46의 (a)는 분할 트리 구조를 나타내고, 도 46의 (b)는 분할 트리 구조에 따른 블록의 분할 예를 나타낸다. 도 46의 (a) 및 도 46의 (b)에 있어서, region #1, region #2, region #3 등이 각각 본 개시의 region에 대응될 수 있다. 즉, 본 개시의 region은 분할 트리 구조의 각 노드를 의미할 수도 있다.

[0485] region에 포함된 모든 leaf node(혹은 CU)의 예측 모드(region의 예측 모드)는 region의 크기, region의 분할 형태 및/또는 region의 부호화 파라미터에 기초하여 결정될 수 있다.

[0486] region의 크기는 region의 가로, 세로, 면적, 픽셀수 등으로 표현될 수 있다. region의 분할 형태는 분할 여부, 쿼드트리 분할, 이진트리 분할, 삼분할트리 분할, 수평 분할, 수직 분할 등으로 표현될 수 있다.

[0487] region의 부호화 파라미터는, 해당 region을 포함하는 슬라이스의 타입, 해당 region을 포함하는 CTU의 분할 구조, 해당 region에 포함된 모든 leaf node(혹은 CU)에 적용될 예측 모드를 시그널링하는 파라미터 등을 포함할 수 있다.

[0488] 예컨대, region의 크기가 소정의 크기이고, region의 분할 형태가 소정의 분할 형태인 경우, region의 예측 모

드는 화면내 예측으로 결정될 수 있다. 이 때, 소정의 크기는 region에 포함된 픽셀수로 나타낼 수 있다.

[0489] 예컨대, region이 64 픽셀을 포함하고 쿼드트리 분할될 때, 해당 region의 예측 모드는 화면내 예측으로 결정될 수 있다.

[0490] 예컨대, region이 64 픽셀을 포함하고 삼분할트리 분할될 때, 해당 region의 예측 모드는 화면내 예측으로 결정될 수 있다.

[0491] 예컨대, region이 32 픽셀을 포함하고 이진트리 분할될 때, 해당 region의 예측 모드는 화면내 예측으로 결정될 수 있다.

[0492] 예컨대, region의 크기가 소정의 크기이고, region의 분할 형태가 소정의 분할 형태인 경우, 해당 region을 포함하는 슬라이스의 타입 및/또는 region의 부호화 파라미터에 따라, region의 예측 모드가 결정될 수 있다. 이 때, 소정의 크기는 region에 포함된 픽셀수로 나타낼 수 있다.

[0493] 예컨대, region이 64 픽셀을 포함하고 이진트리 분할될 때, 해당 region을 포함하는 슬라이스가 I 슬라이스이면, region의 예측 모드는 화면내 예측으로 결정될 수 있다. 이 때, 해당 region을 포함하는 슬라이스가 I 슬라이스가 아니면, region의 예측 모드는 region의 부호화 파라미터에 기초하여 화면내 예측 또는 화면간 예측으로 결정될 수 있다. 상기 부호화 파라미터는 비트스트림을 통해 시그널링되는 신택스 요소일 수 있다.

[0494] 예컨대, region이 128 픽셀을 포함하고 삼분할트리 분할될 때, 해당 region을 포함하는 슬라이스가 I 슬라이스이면, region의 예측 모드는 화면내 예측으로 결정될 수 있다. 이 때, 해당 region을 포함하는 슬라이스가 I 슬라이스가 아니면, region의 예측 모드는 region의 부호화 파라미터에 기초하여 화면내 예측 또는 화면간 예측으로 결정될 수 있다. 상기 부호화 파라미터는 비트스트림을 통해 시그널링되는 신택스 요소일 수 있다.

[0495] region의 예측 모드가 결정되면, region에 포함된 모든 leaf node(혹은 CU)에 대한 pred_mode_flag는 시그널링되지 않을 수 있다.

[0496] 본 개시에 있어서, 소정의 동작을 수행하는 영상 부호화기 또는 영상 복호화기는 해당 동작의 수행 조건이나 상황을 확인하는 동작을 수행할 수 있다. 예컨대, region의 크기가 소정의 크기이고, region의 분할 형태가 소정의 분할 형태인 경우, region의 예측 모드는 화면내 예측으로 결정될 때, 영상 부호화기 또는 영상 복호화기는 region의 예측 모드를 결정하기 위해, 해당 region의 크기가 소정의 크기인지, 해당 region의 분할 형태가 소정의 분할 형태인지 여부를 확인할 수 있다.

[0497] 상기의 실시예들은 부호화기 및 복호화기에서 같은 방법으로 수행될 수 있다.

[0498] 상기 실시예들 중 적어도 하나 혹은 적어도 하나의 조합을 이용해서 영상을 부호화/복호화할 수 있다.

[0499] 상기 실시예를 적용하는 순서는 부호화기와 복호화기에서 상이할 수 있고, 상기 실시예를 적용하는 순서는 부호화기와 복호화기에서 동일할 수 있다.

[0500] 휘도 및 색차 신호 각각에 대하여 상기 실시예를 수행할 수 있고, 휘도 및 색차 신호에 대한 상기 실시예를 동일하게 수행할 수 있다.

[0501] 본 발명의 상기 실시예들이 적용되는 블록의 형태는 정방형(square) 형태 혹은 비정방형(non-square) 형태를 가질 수 있다.

[0502] 본 발명의 상기 실시예들은 부호화 블록, 예측 블록, 변환 블록, 블록, 현재 블록, 부호화 유닛, 예측 유닛, 변환 유닛, 유닛, 현재 유닛 중 적어도 하나의 크기에 따라 적용될 수 있다. 여기서의 크기는 상기 실시예들이 적용되기 위해 최소 크기 및/또는 최대 크기로 정의될 수도 있고, 상기 실시예가 적용되는 고정 크기로 정의될 수도 있다. 또한, 상기 실시예들은 제1 크기에서는 제1의 실시예가 적용될 수도 있고, 제2 크기에서는 제2의 실시예가 적용될 수도 있다. 즉, 상시 실시예들은 크기에 따라 복합적으로 적용될 수 있다. 또한, 본 발명의 상기 실시예들은 최소 크기 이상 및 최대 크기 이하일 경우에만 적용될 수도 있다. 즉, 상기 실시예들을 블록 크기가 일정한 범위 내에 포함될 경우에만 적용될 수도 있다.

[0503] 예를 들어, 현재 블록의 크기가 8x8 이상일 경우에만 상기 실시예들이 적용될 수 있다. 예를 들어, 현재 블록의 크기가 4x4일 경우에만 상기 실시예들이 적용될 수 있다. 예를 들어, 현재 블록의 크기가 16x16 이하일 경우에만 상기 실시예들이 적용될 수 있다. 예를 들어, 현재 블록의 크기가 16x16 이상이고 64x64 이하일 경우에만 상기 실시예들이 적용될 수 있다.

[0504] 본 발명의 상기 실시예들은 시간적 계층(temporal layer)에 따라 적용될 수 있다. 상기 실시예들이 적용 가능한 시간적 계층을 식별하기 위해 별도의 식별자(identifier)가 시그널링되고, 해당 식별자에 의해 특정된 시간적 계층에 대해서 상기 실시예들이 적용될 수 있다. 여기서의 식별자는 상기 실시예가 적용 가능한 최하위 계층 및/또는 최상위 계층으로 정의될 수도 있고, 상기 실시예가 적용되는 특정 계층을 지시하는 것으로 정의될 수도 있다. 또한, 상기 실시예가 적용되는 고정된 시간적 계층이 정의될 수도 있다.

[0505] 예를 들어, 현재 영상의 시간적 계층이 최하위 계층일 경우에만 상기 실시예들이 적용될 수 있다. 예를 들어, 현재 영상의 시간적 계층 식별자가 1 이상인 경우에만 상기 실시예들이 적용될 수 있다. 예를 들어, 현재 영상의 시간적 계층이 최상위 계층일 경우에만 상기 실시예들이 적용될 수 있다.

[0506] 본 발명의 상기 실시예들이 적용되는 슬라이스 종류(slice type) 혹은 타일 그룹 종류가 정의되고, 해당 슬라이스 종류 혹은 타일 그룹 종류에 따라 본 발명의 상기 실시예들이 적용될 수 있다.

[0507] 상술한 실시예들에서, 방법들은 일련의 단계 또는 유닛으로서 순서도를 기초로 설명되고 있으나, 본 발명은 단계들의 순서에 한정되는 것은 아니며, 어떤 단계는 상술한 바와 다른 단계와 다른 순서로 또는 동시에 발생할 수 있다. 또한, 당해 기술 분야에서 통상의 지식을 가진 자라면 순서도에 나타난 단계들이 배타적이지 않고, 다른 단계가 포함되거나, 순서도의 하나 또는 그 이상의 단계가 본 발명의 범위에 영향을 미치지 않고 삭제될 수 있음을 이해할 수 있을 것이다.

[0508] 상술한 실시예는 다양한 양태의 예시들을 포함한다. 다양한 양태들을 나타내기 위한 모든 가능한 조합을 기술할 수는 없지만, 해당 기술 분야의 통상의 지식을 가진 자는 다른 조합이 가능함을 인식할 수 있을 것이다. 따라서, 본 발명은 이하의 특허청구범위 내에 속하는 모든 다른 교체, 수정 및 변경을 포함한다고 할 것이다.

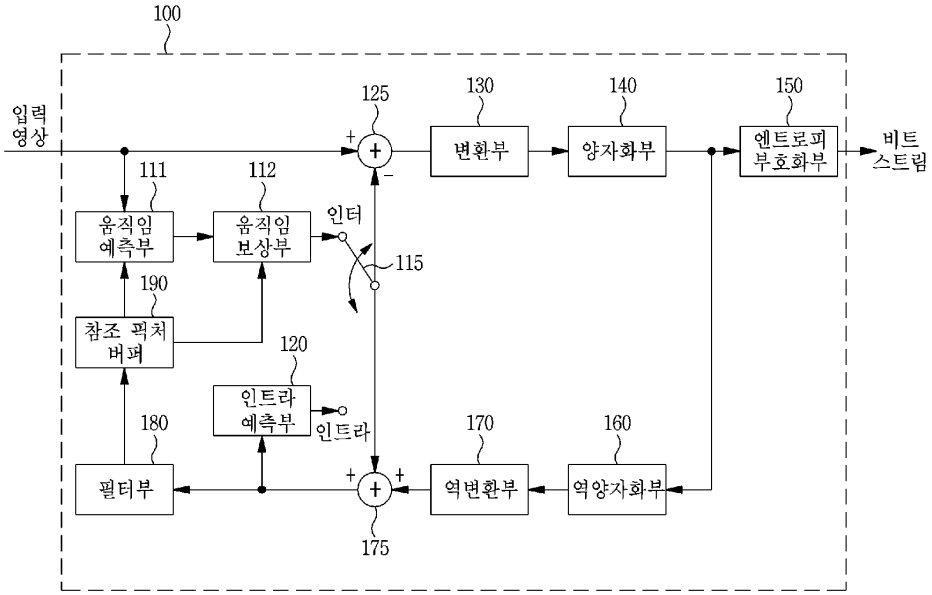
[0509] 이상 설명된 본 발명에 따른 실시예들은 다양한 컴퓨터 구성요소를 통하여 수행될 수 있는 프로그램 명령어의 형태로 구현되어 컴퓨터 판독 가능한 기록 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능한 기록 매체는 프로그램 명령어, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 컴퓨터 판독 가능한 기록 매체에 기록되는 프로그램 명령어는 본 발명을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 분야의 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능한 기록 매체의 예에는, 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체, CD-ROM, DVD와 같은 광기록 매체, 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 ROM, RAM, 플래시 메모리 등과 같은 프로그램 명령어를 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령어의 예에는, 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드도 포함된다. 상기 하드웨어 장치는 본 발명에 따른 처리를 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

[0510] 이상에서 본 발명이 구체적인 구성요소 등과 같은 특정 사항들과 한정된 실시예 및 도면에 의해 설명되었으나, 이는 본 발명의 보다 전반적인 이해를 돕기 위해서 제공된 것일 뿐, 본 발명이 상기 실시예들에 한정되는 것은 아니며, 본 발명이 속하는 기술분야에서 통상적인 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형을 꾀할 수 있다.

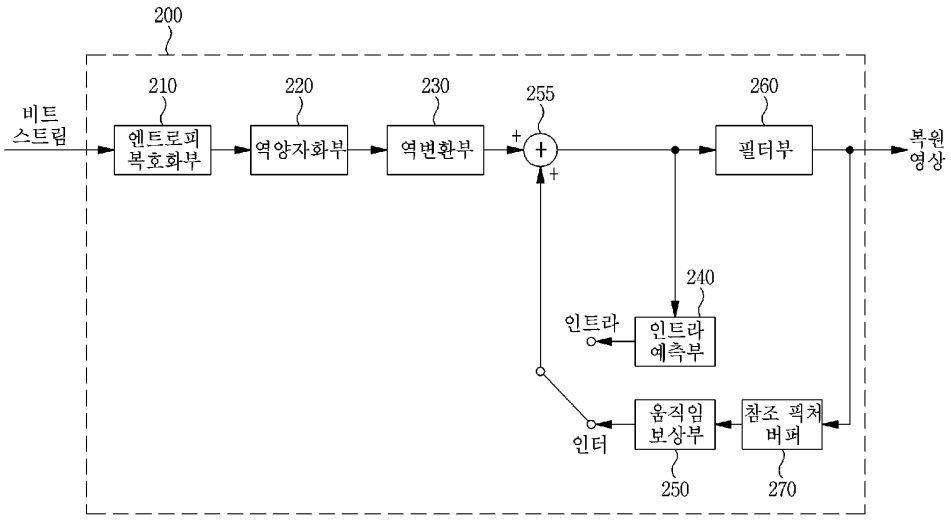
[0511] 따라서, 본 발명의 사상은 상기 설명된 실시예에 국한되어 정해져서는 아니 되며, 후술하는 특허청구범위뿐만 아니라 이 특허청구범위와 균등하게 또는 등가적으로 변형된 모든 것들은 본 발명의 사상의 범주에 속한다고 할 것이다.

도면

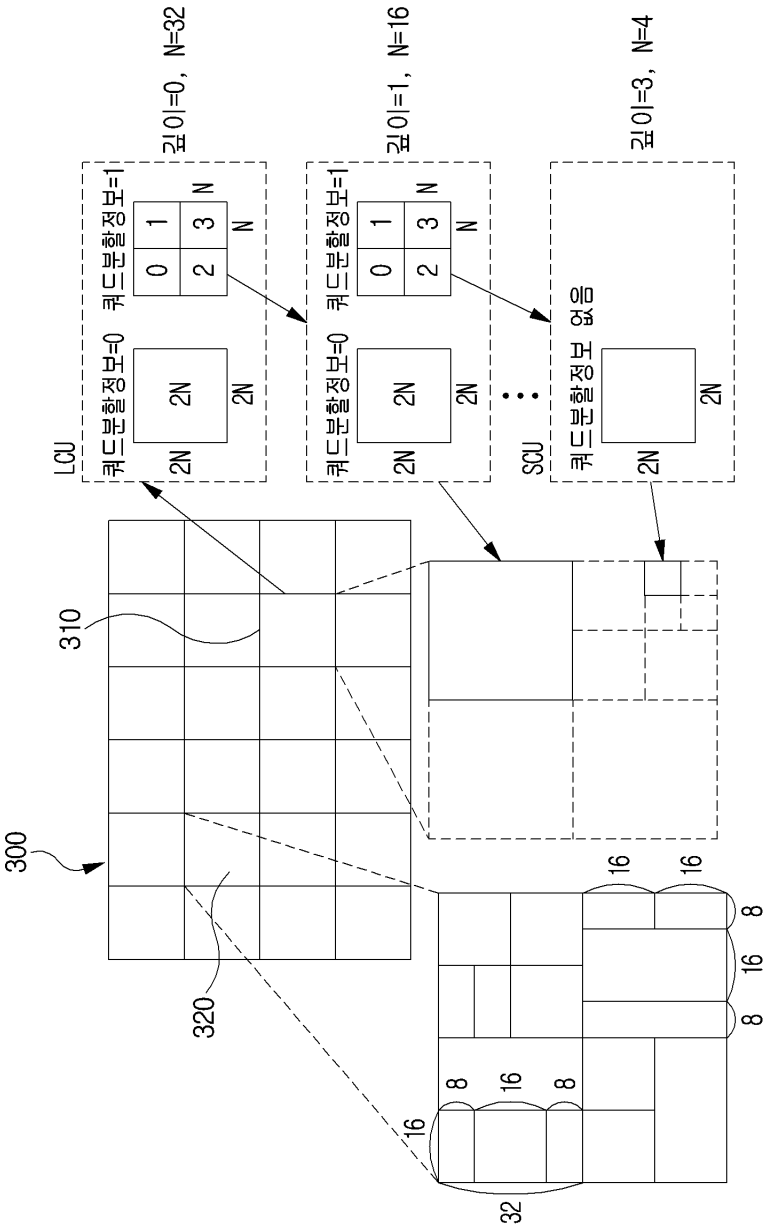
도면1


100
입력 영상
125
130
변환부
140
양자화부
150
엔트로피 부호화부
비트 스트림
111
움직임 예측부
112
움직임 보상부
인터
115
190
참조 픽처 버퍼
120
인트라 예측부
인트라
180
필터부
170
역변환부
160
역양자화부
175

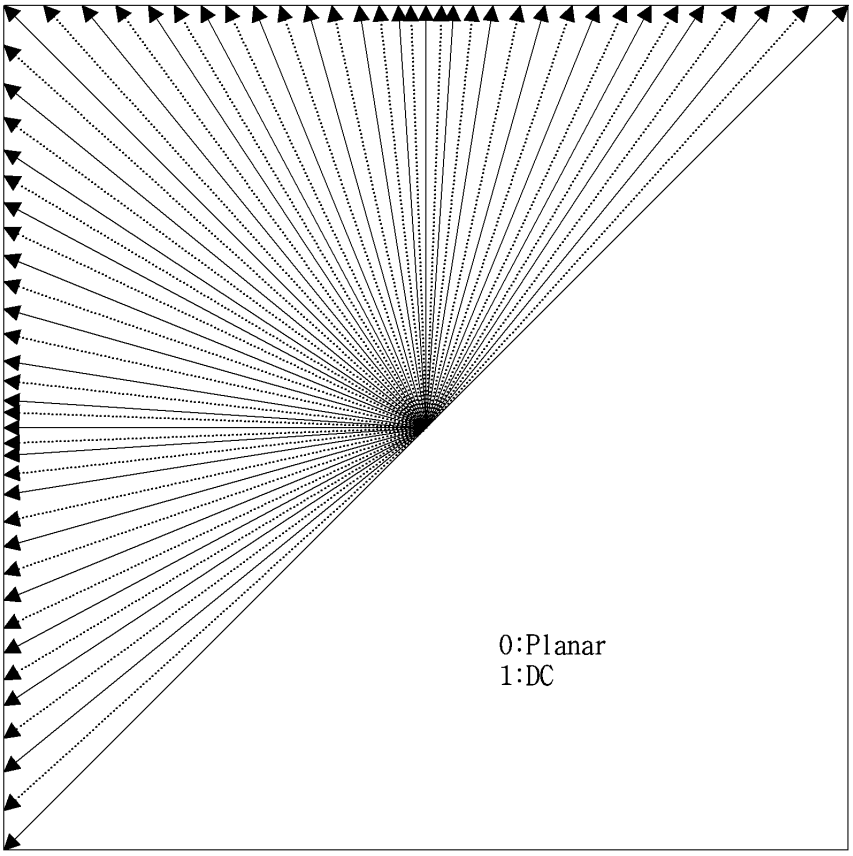
도면2


200
비트 스트림
210
엔트로피 복호화부
220
역양자화부
230
역변환부
255
260
필터부
복원 영상
240
인트라
인트라 예측부
움직임 보상부
인터
참조 픽처 버퍼
250
270

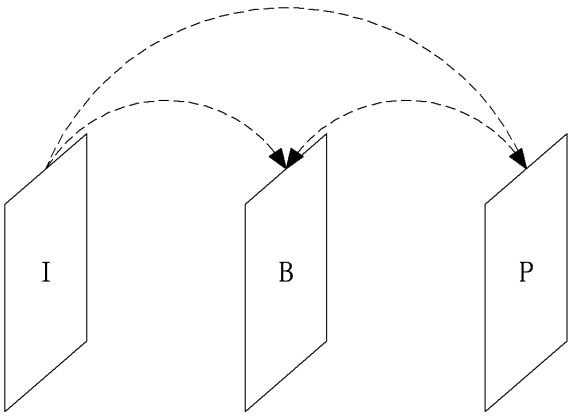
도면3


깊이=0, N=32
깊이=1, N=16
깊이=3, N=4
LCU
SCU
쿼드분할정보=0
쿼드분할정보=1
쿼드분할정보 없음
2N
N
0
1
2
3
310
320
300
16
8

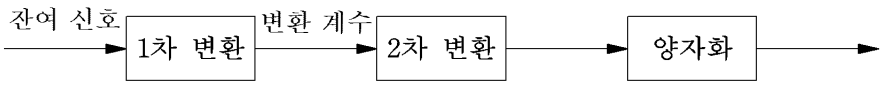
도면4



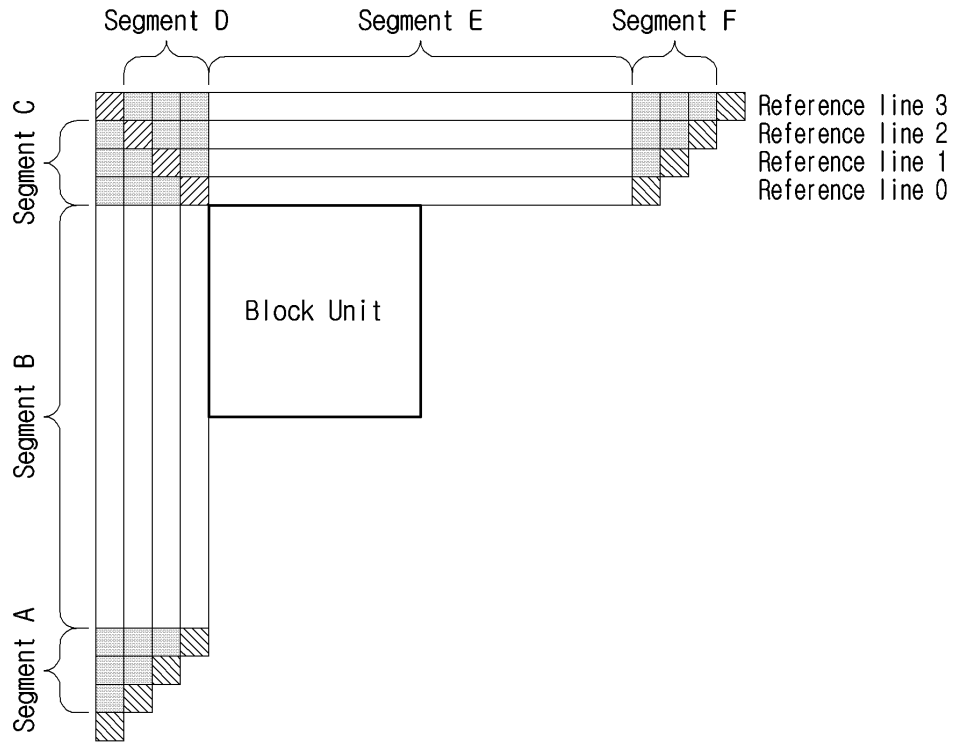
도면5



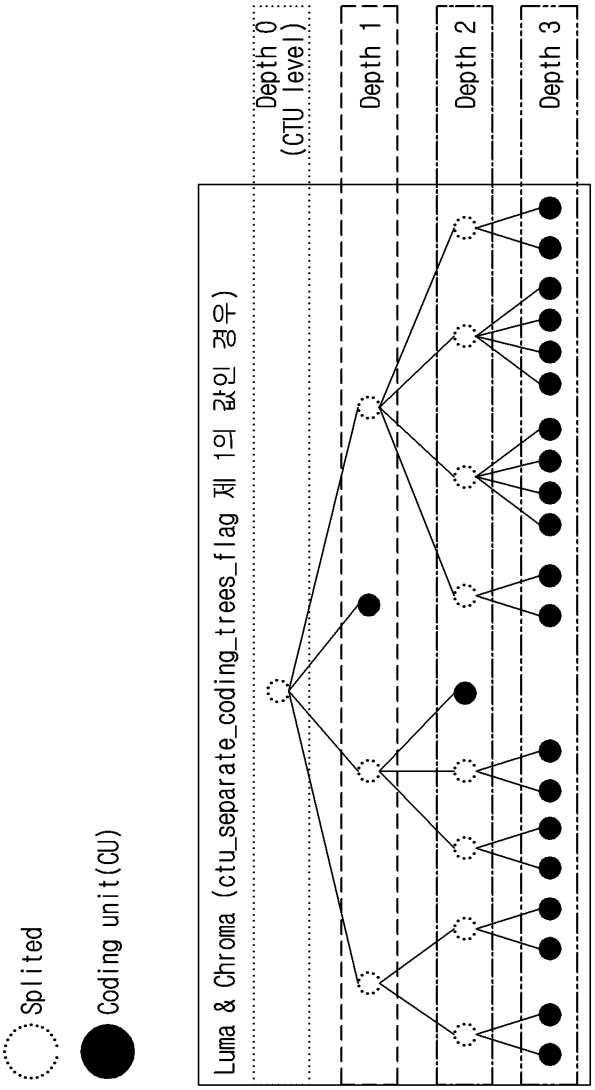
도면6



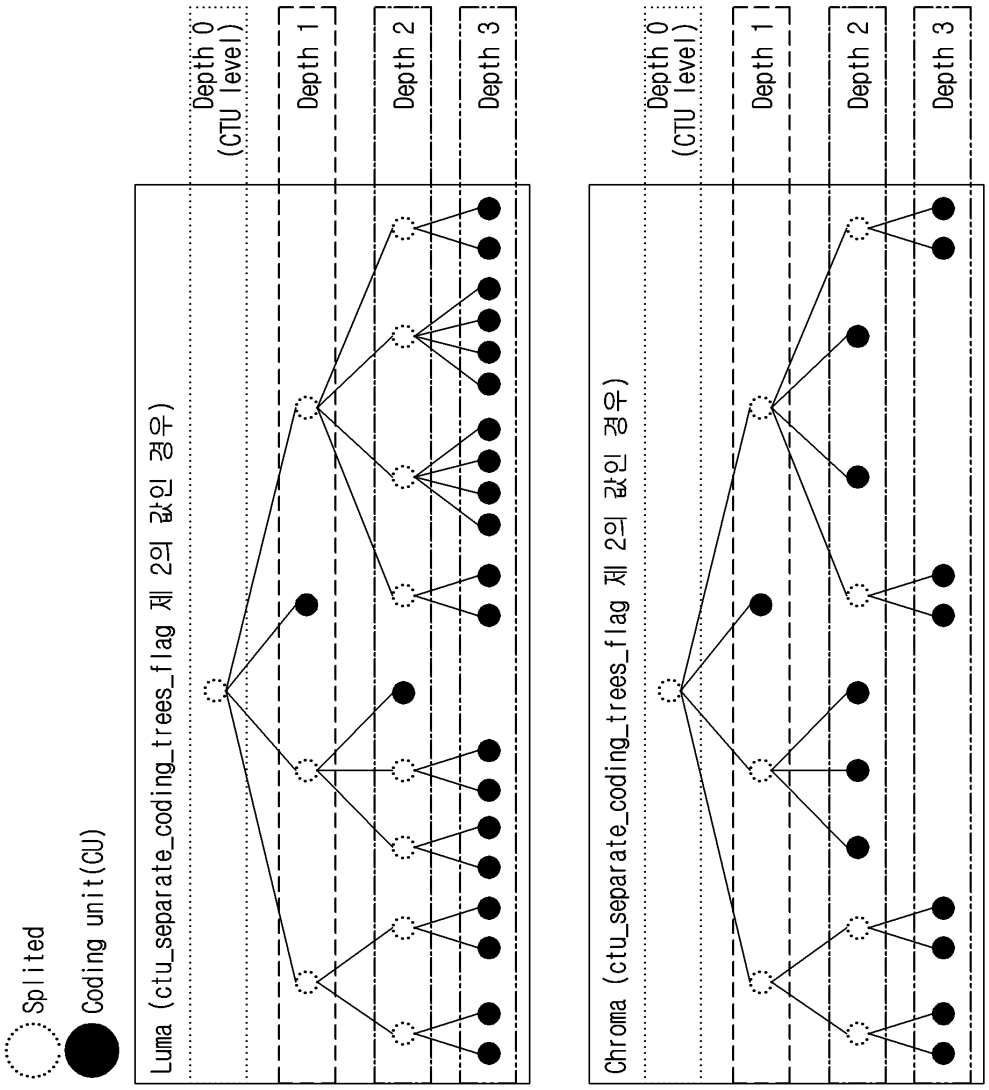
도면7


Segment D
Segment E
Segment F
Segment C
Segment B
Segment A
Reference line 3
Reference line 2
Reference line 1
Reference line 0
Block Unit

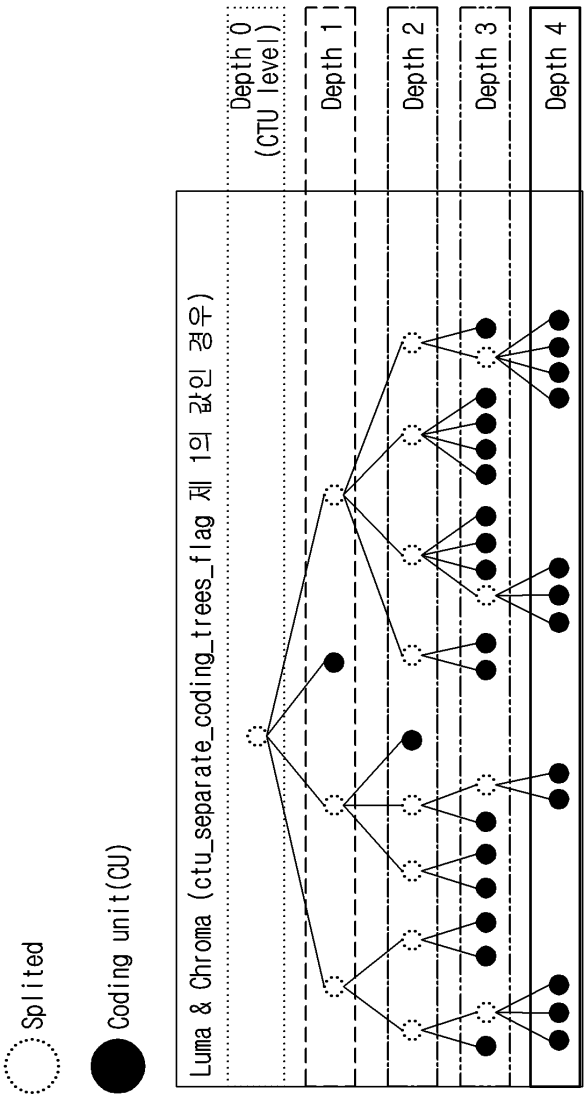
도면8


Splited
Coding unit(CU)
Luma & Chroma (ctu_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

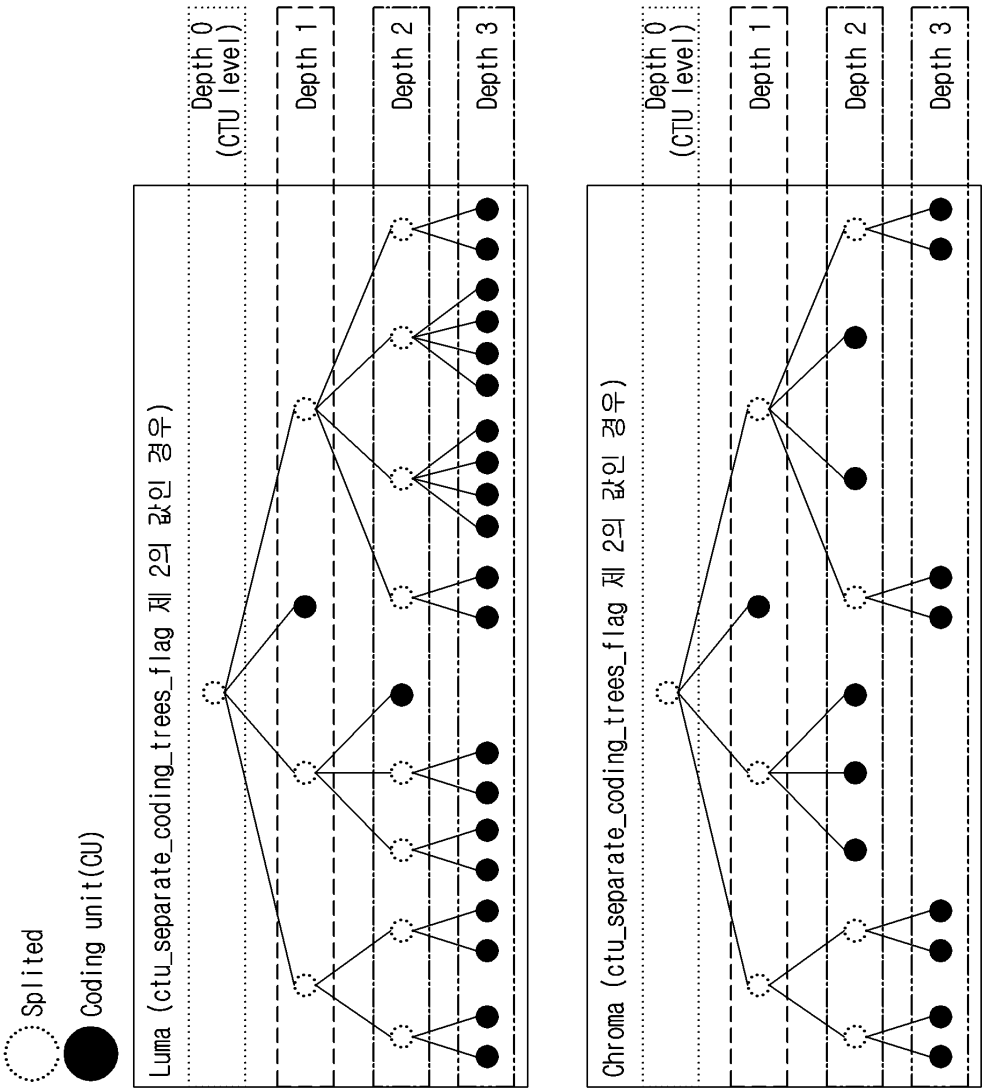
도면9


Splited
Coding unit(CU)
Luma (ctu_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Chroma (ctu_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3

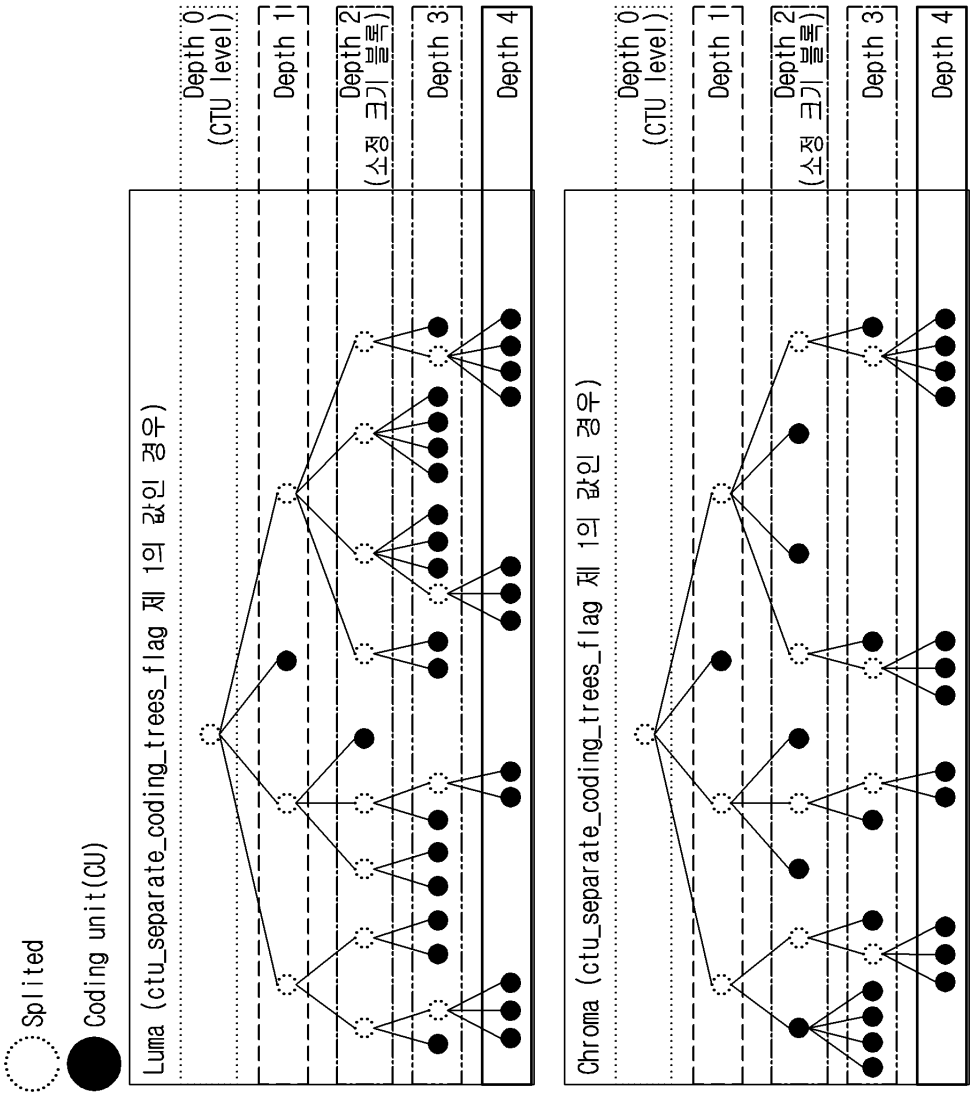
도면10


Splited
Coding unit(CU)
Luma & Chroma (ctu_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

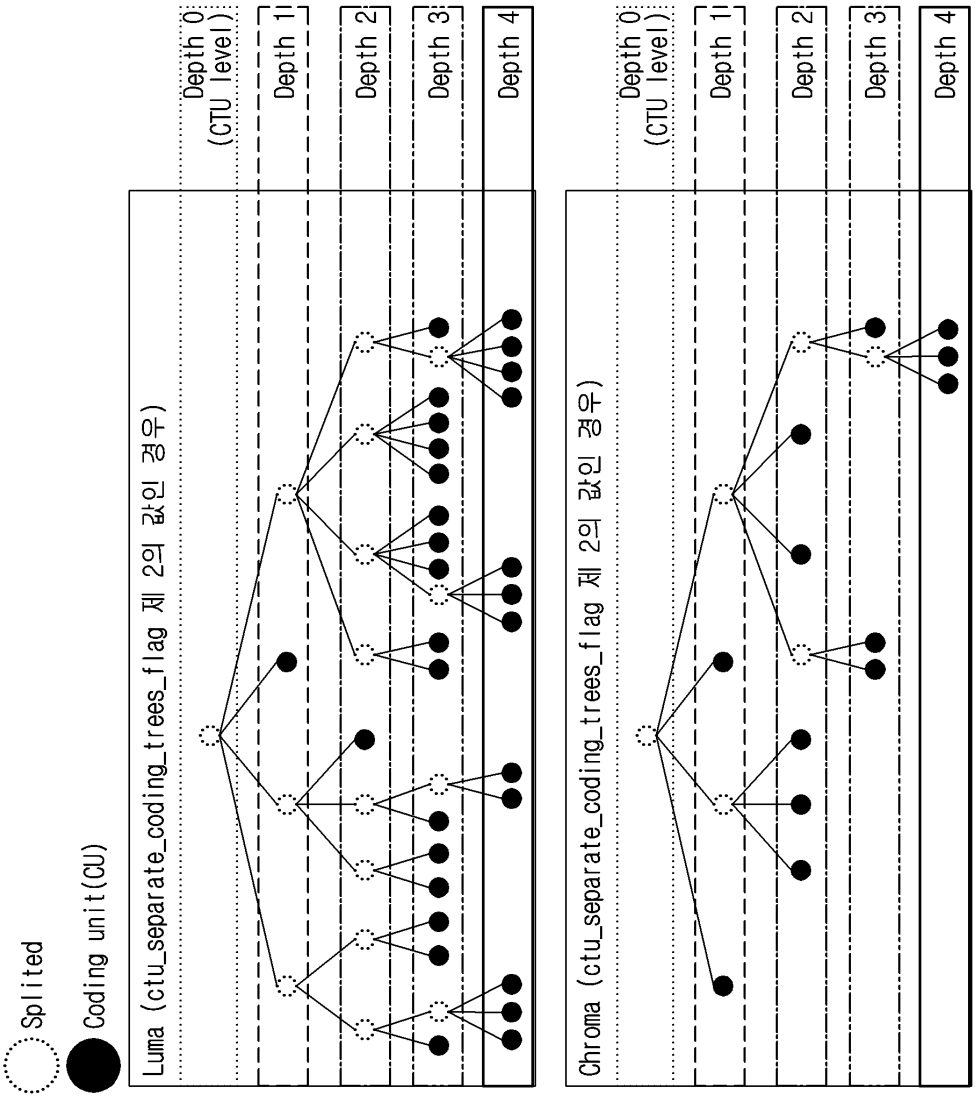
도면11


Splited
Coding unit(CU)
Luma (ctu_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3
Chroma (ctu_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

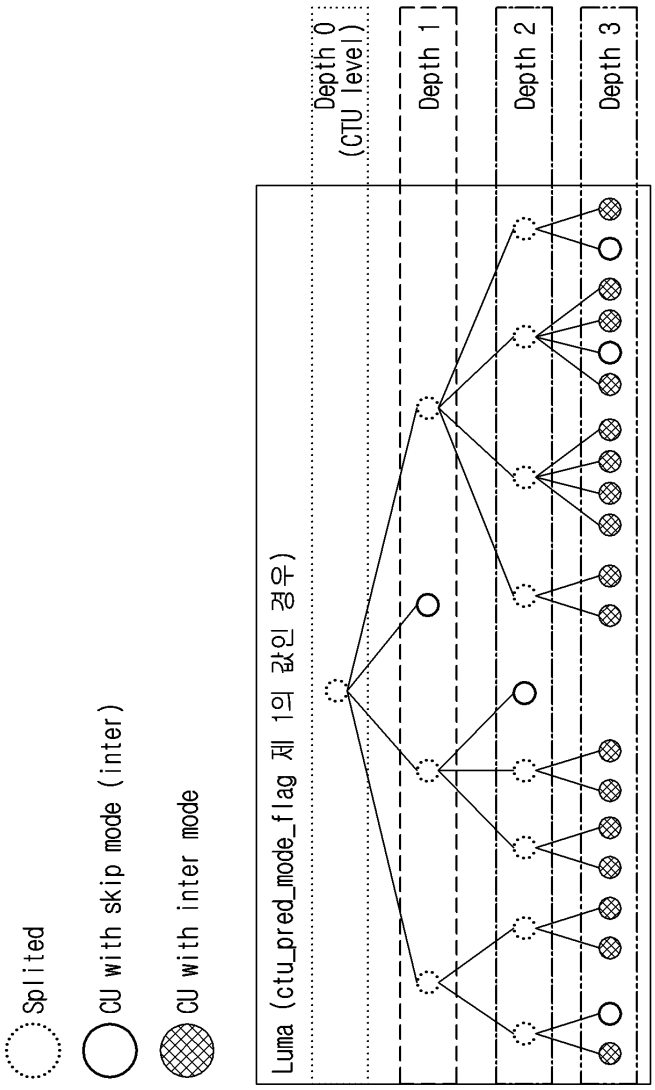
도면12


Splited
Coding unit(CU)
Luma (ctu_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 크기 블록)
Depth 3
Depth 4
Chroma (ctu_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 크기 블록)
Depth 3
Depth 4

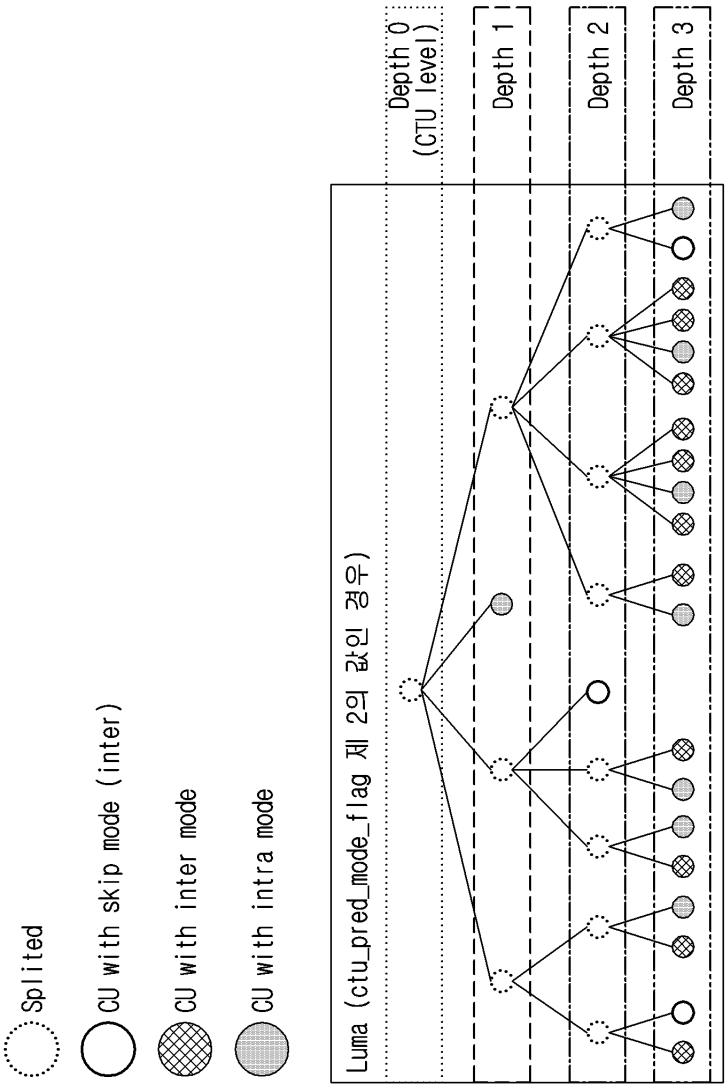
도면13


Splited
Coding unit(CU)
Luma (ctu_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Depth 4
Chroma (ctu_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

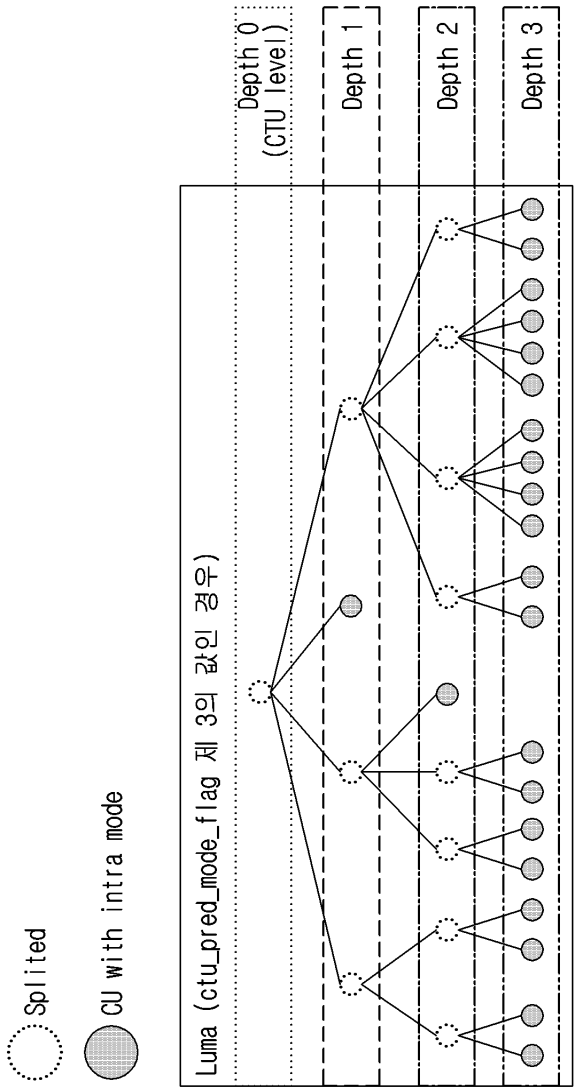
도면14


Splited
CU with skip mode (inter)
CU with inter mode
Luma (ctu_pred_mode_flag 제 1의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

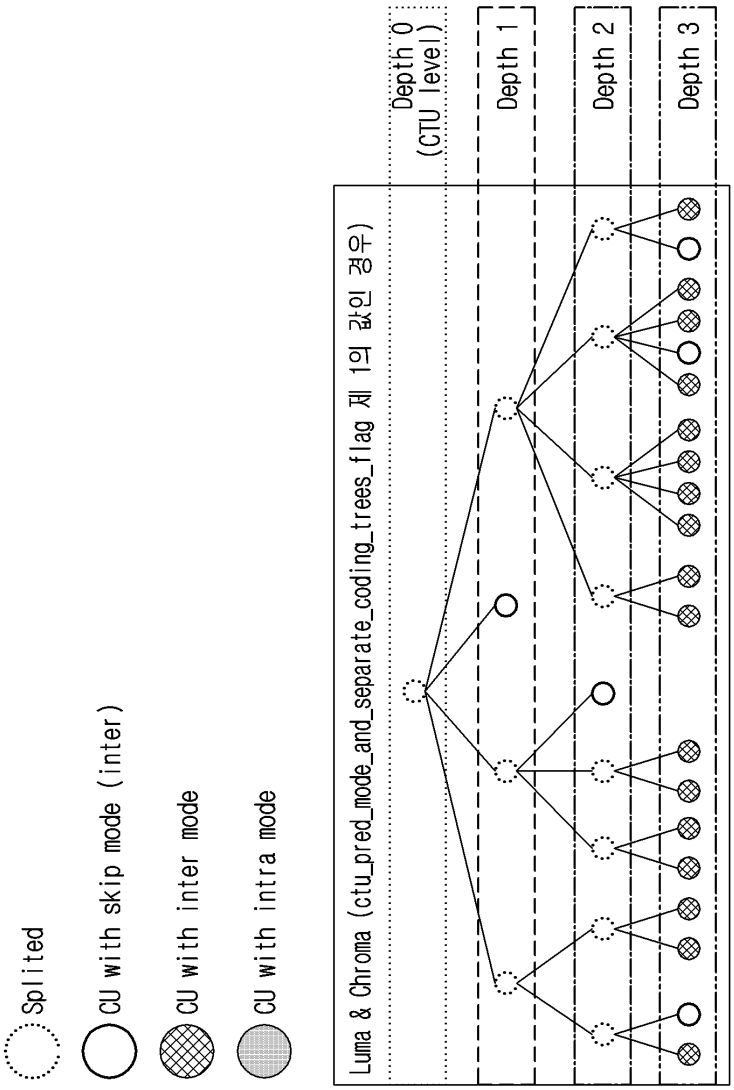
도면15


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_flag 제 2의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

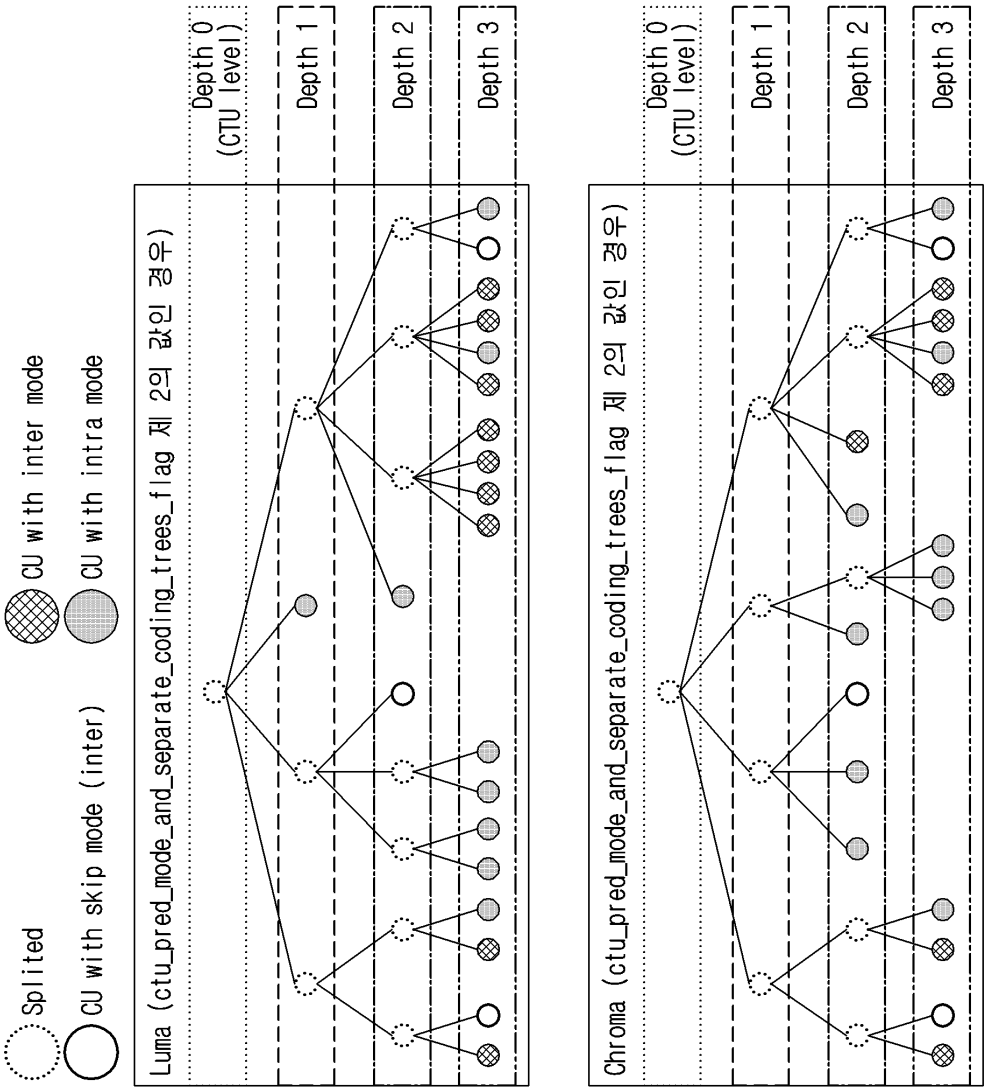
도면16


Splited
CU with intra mode
Luma (ctu_pred_mode_flag 제 3의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

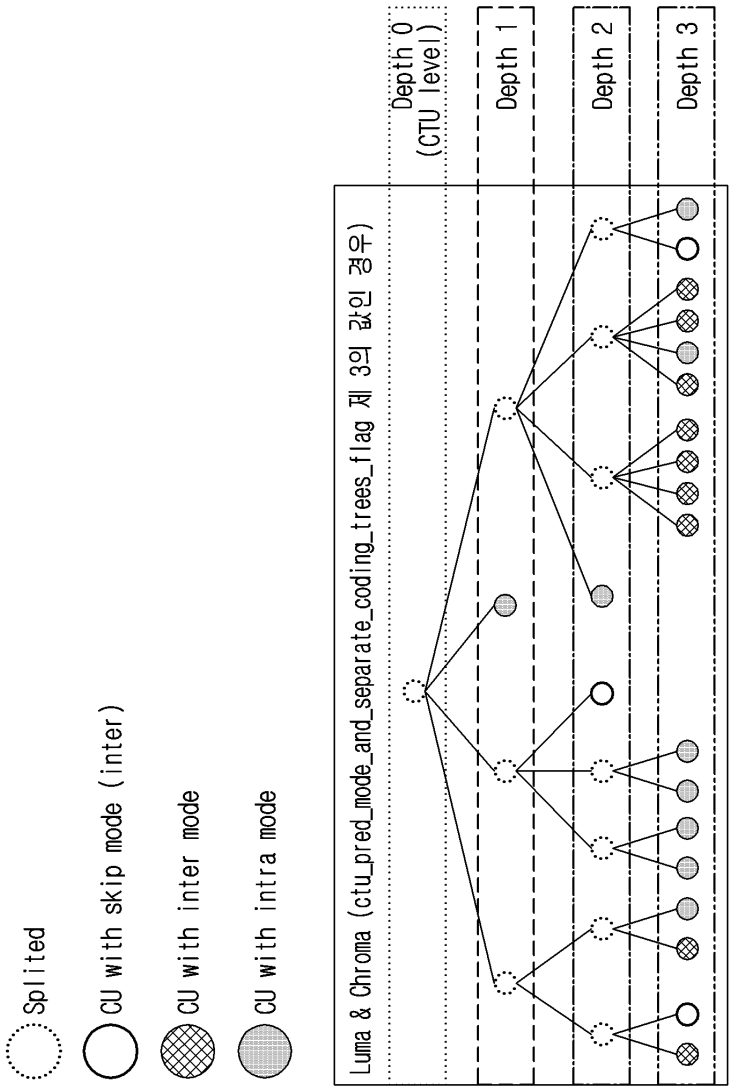
도면17


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

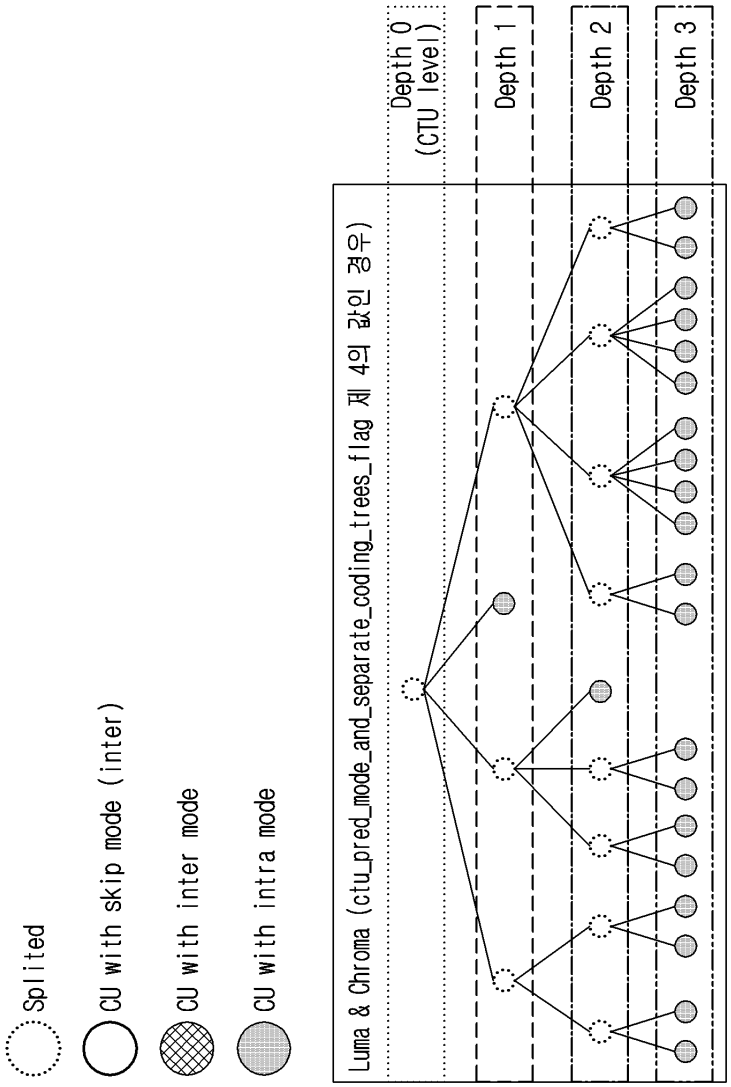
도면18


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3

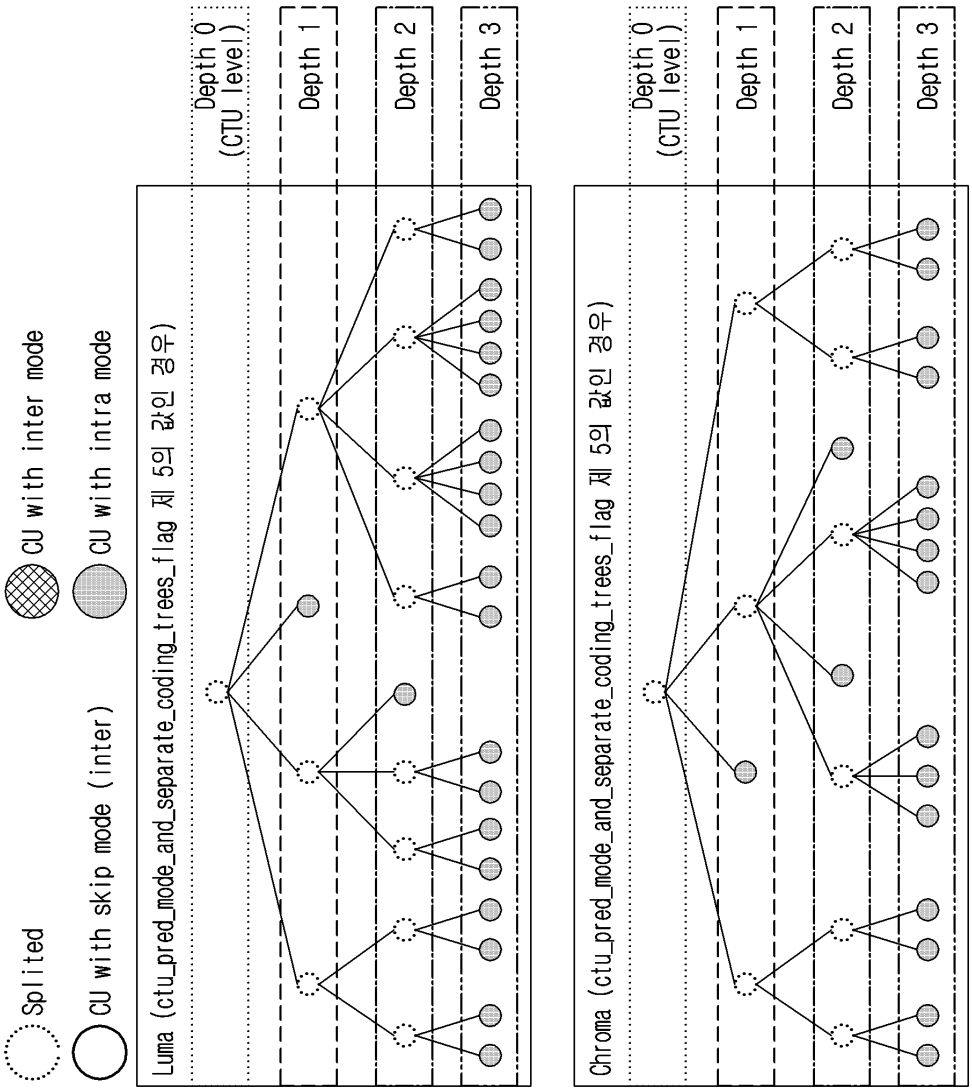
도면19


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 3의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

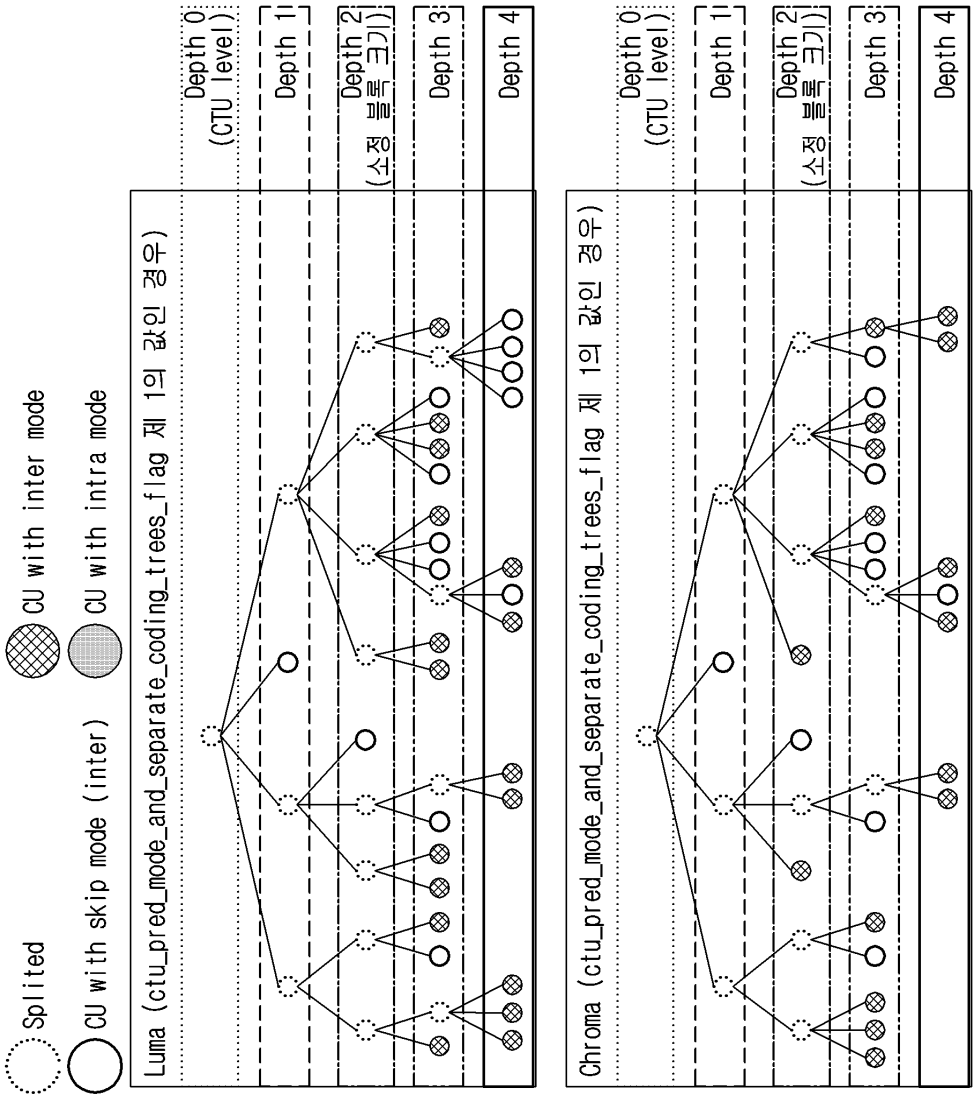
도면20


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 4의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

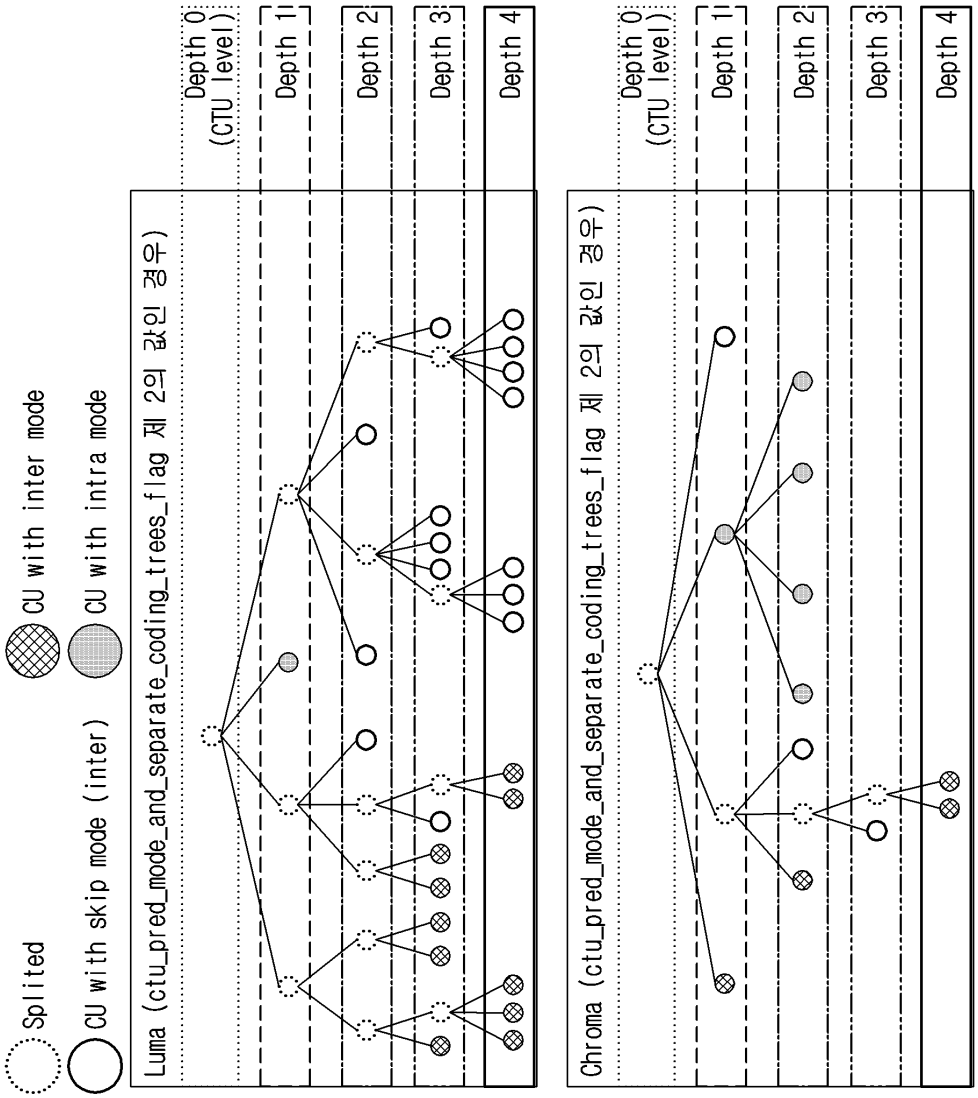
도면21


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 5의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 5의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3

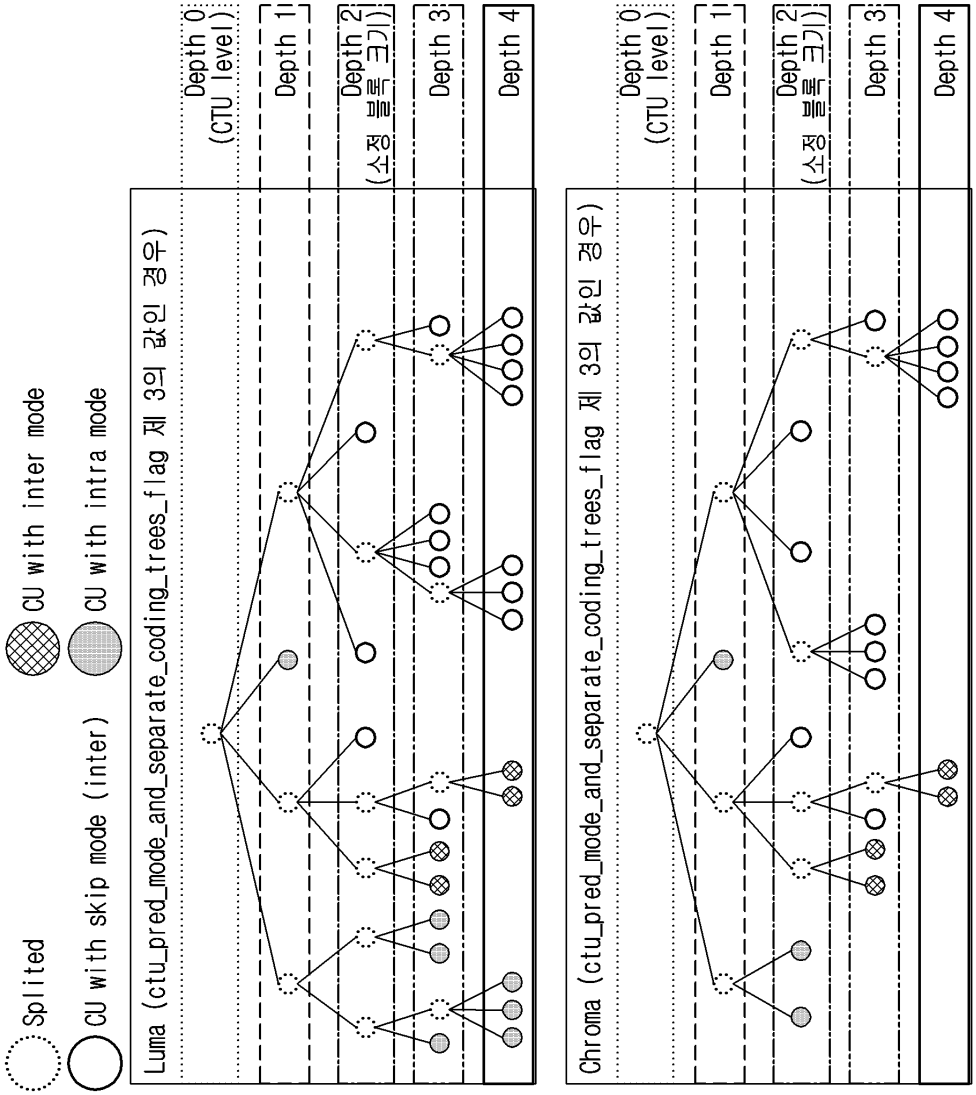
도면22


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정
Depth 3
Depth 4

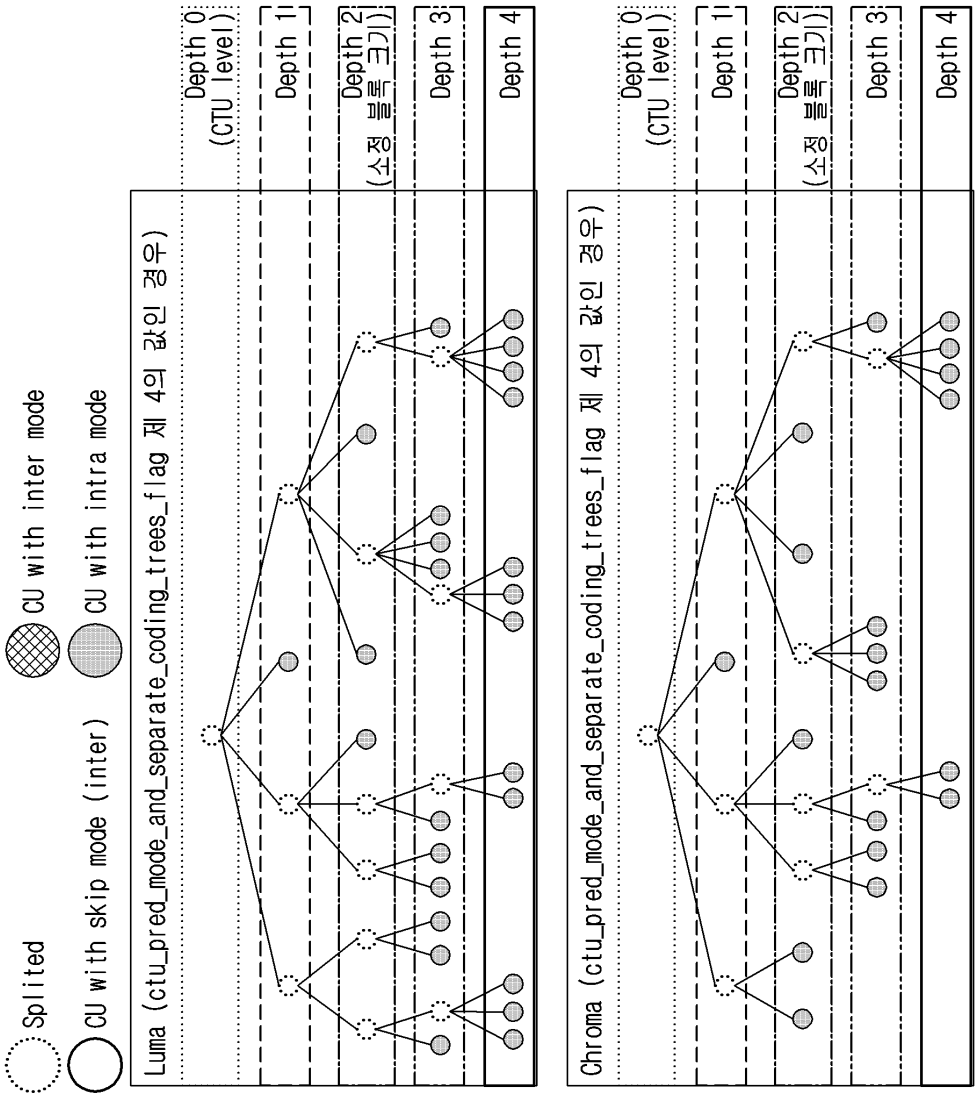
도면23


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

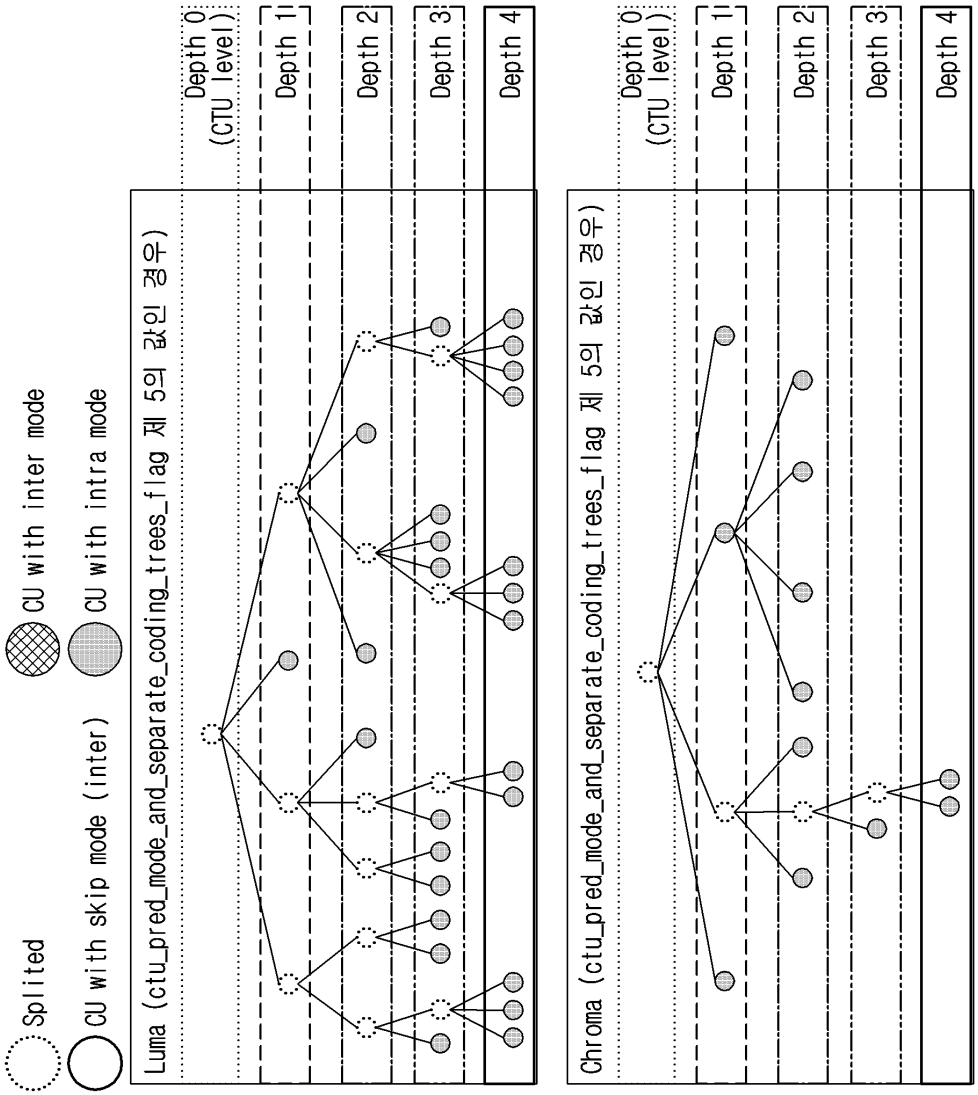
도면24


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 3의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 3의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정
Depth 3
Depth 4

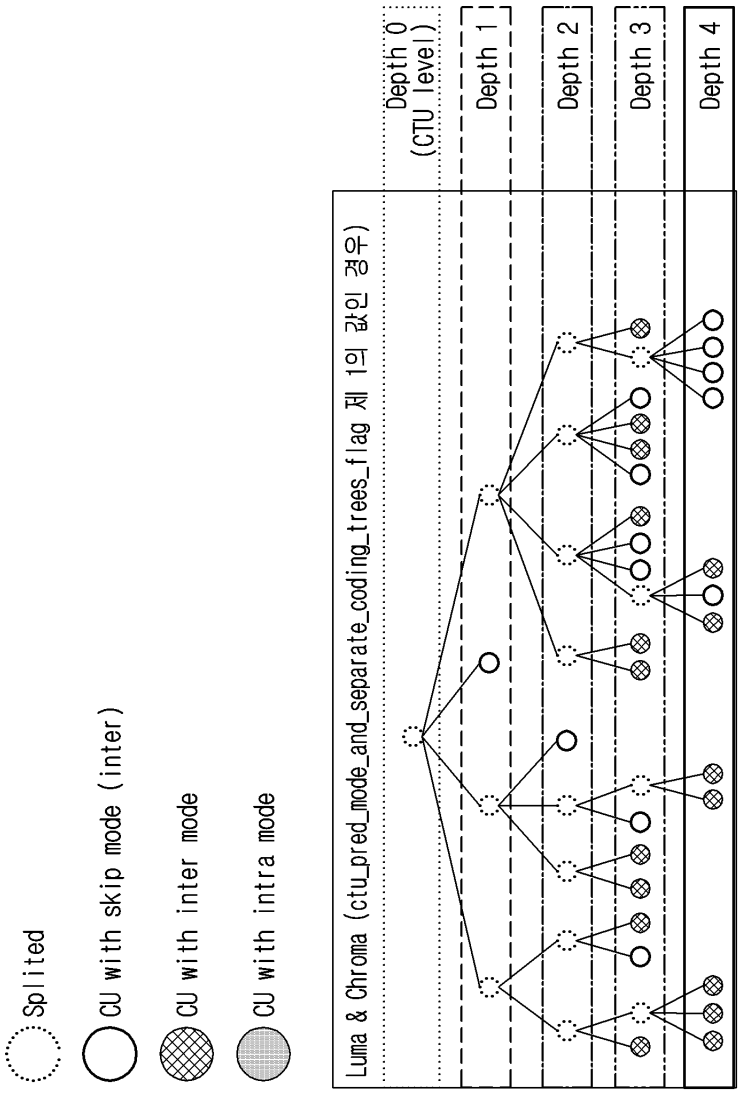
도면25


Splited
CU with inter mode
CU with skip mode (inter)
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 4의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 4의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4

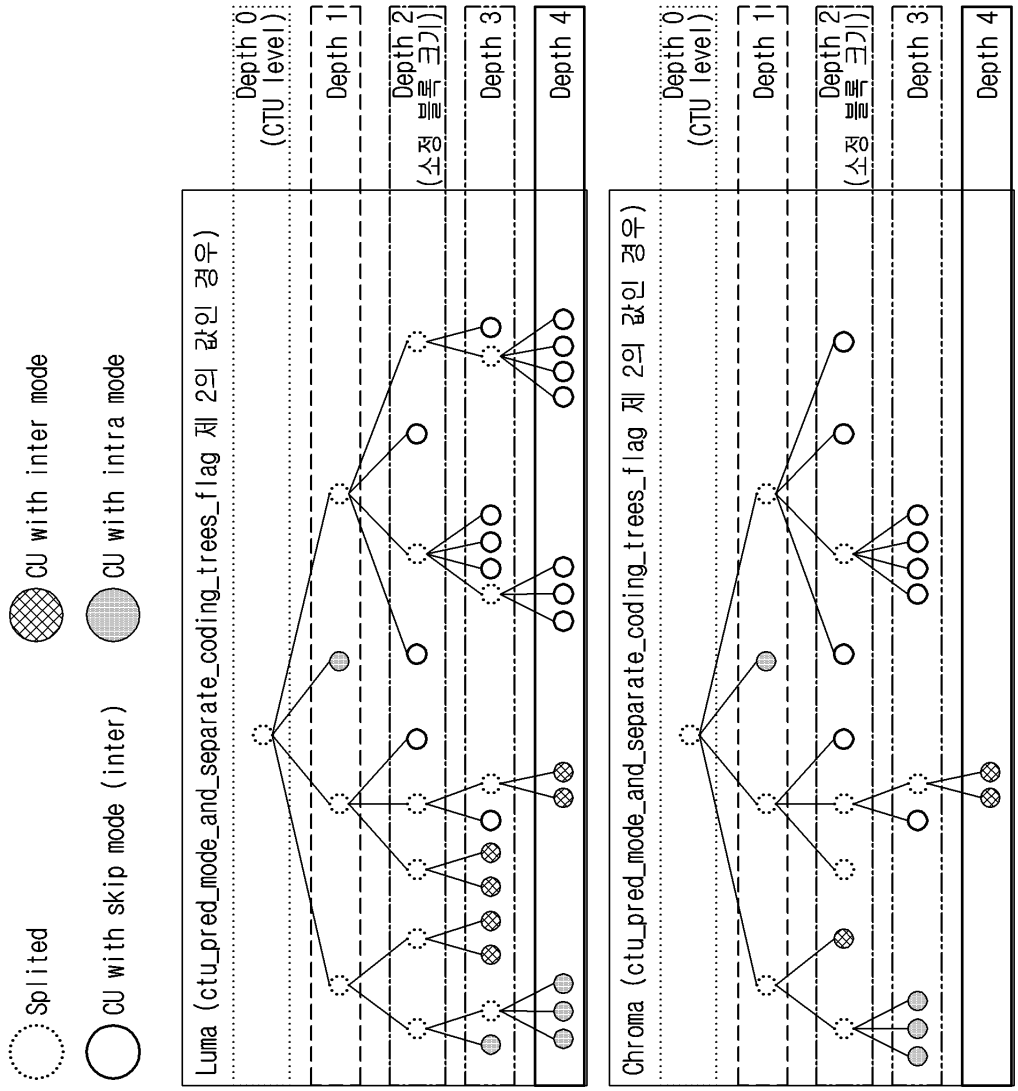
도면26


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 5의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 5의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

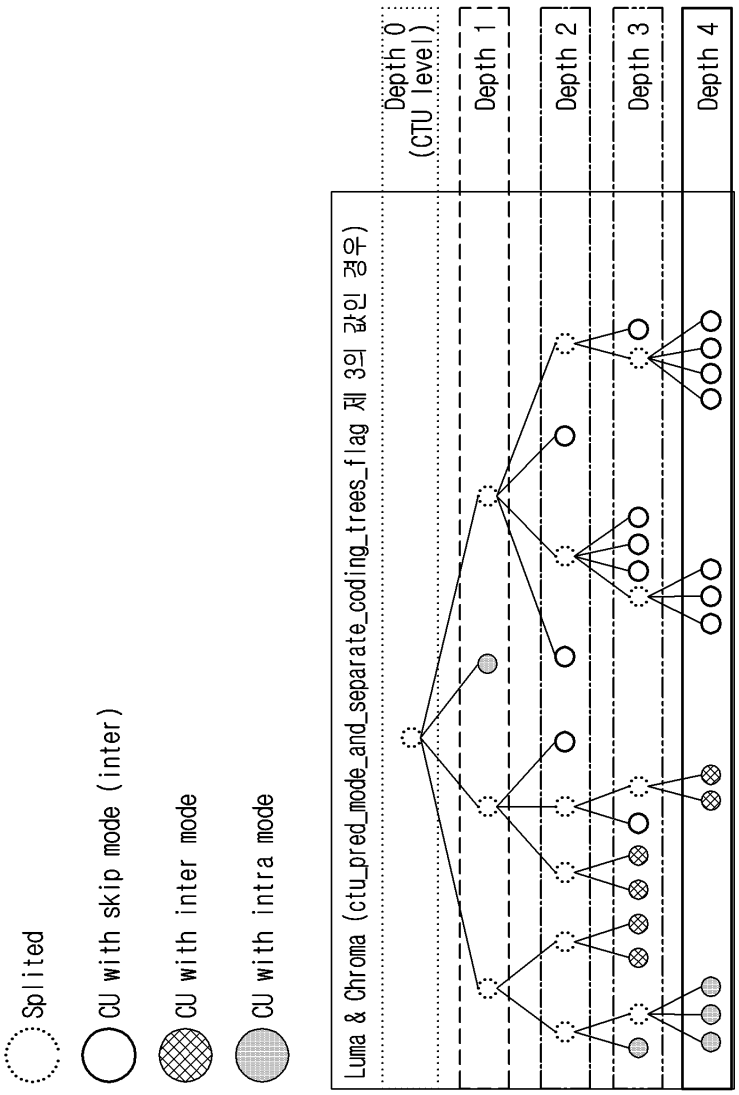
도면27


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 1의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

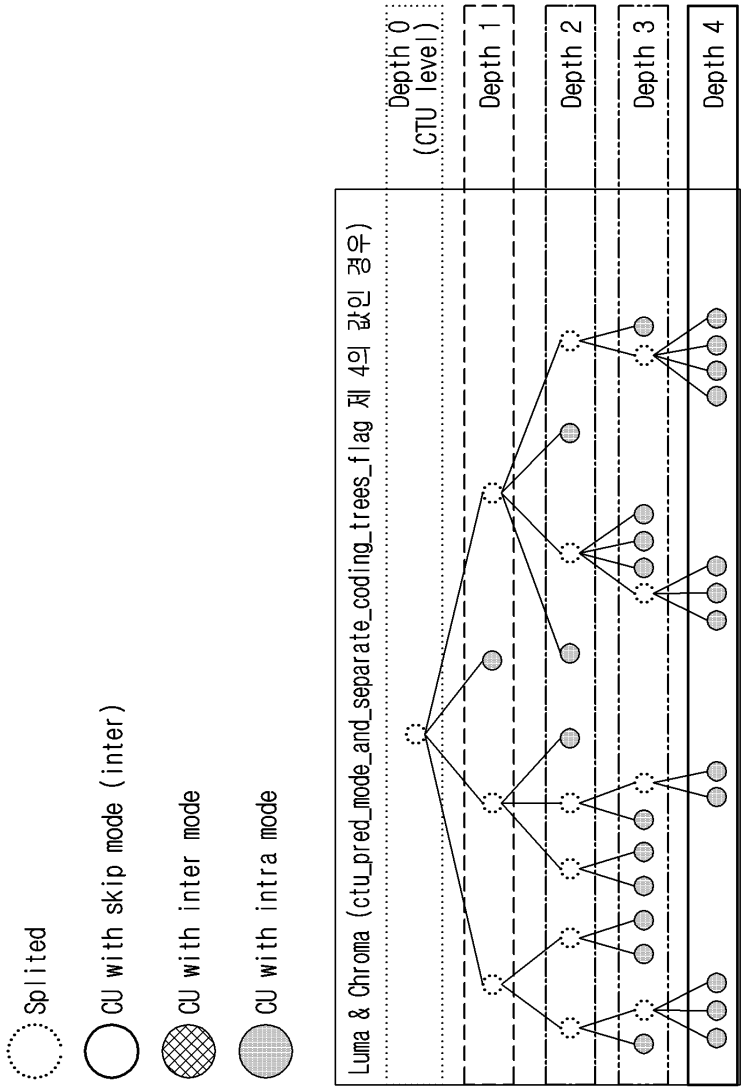
도면28


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 2의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4

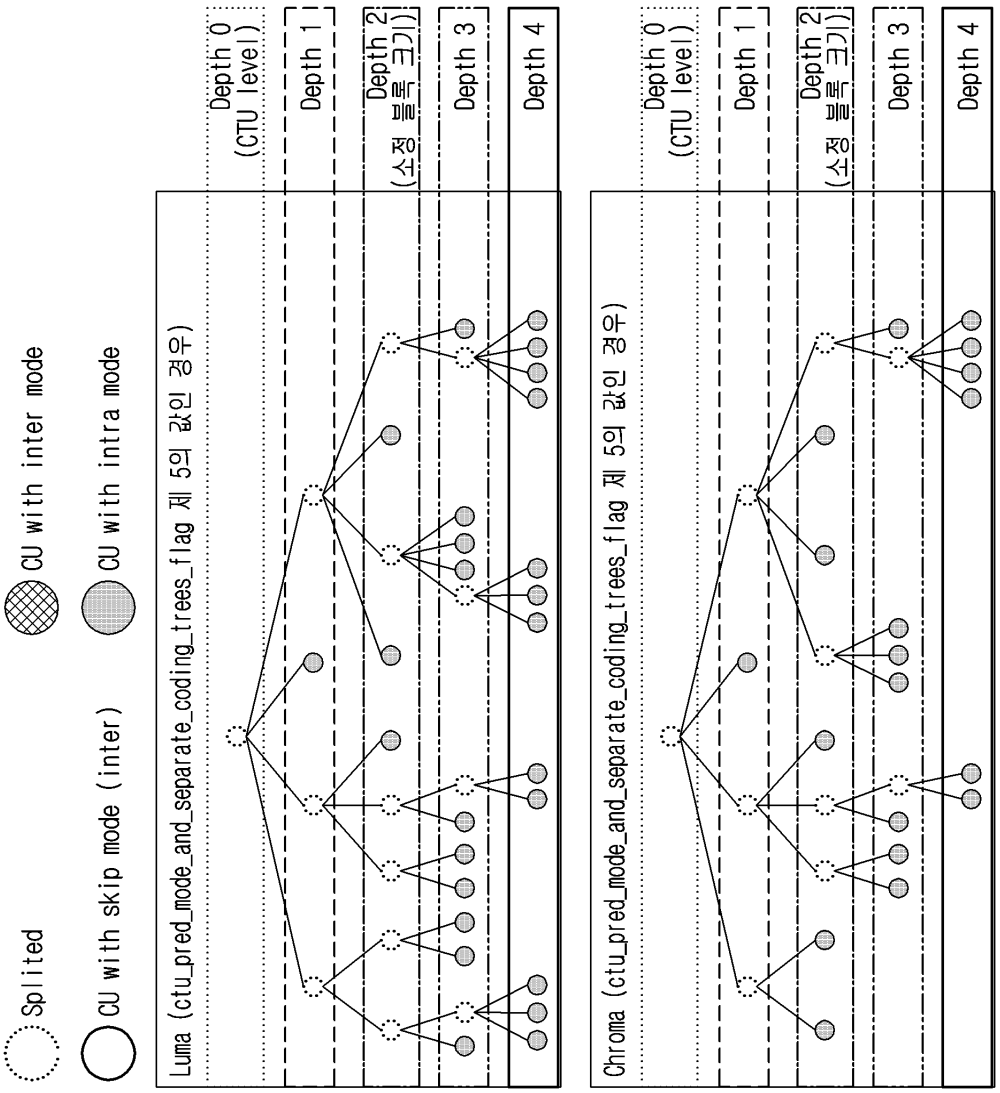
도면29


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 3의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

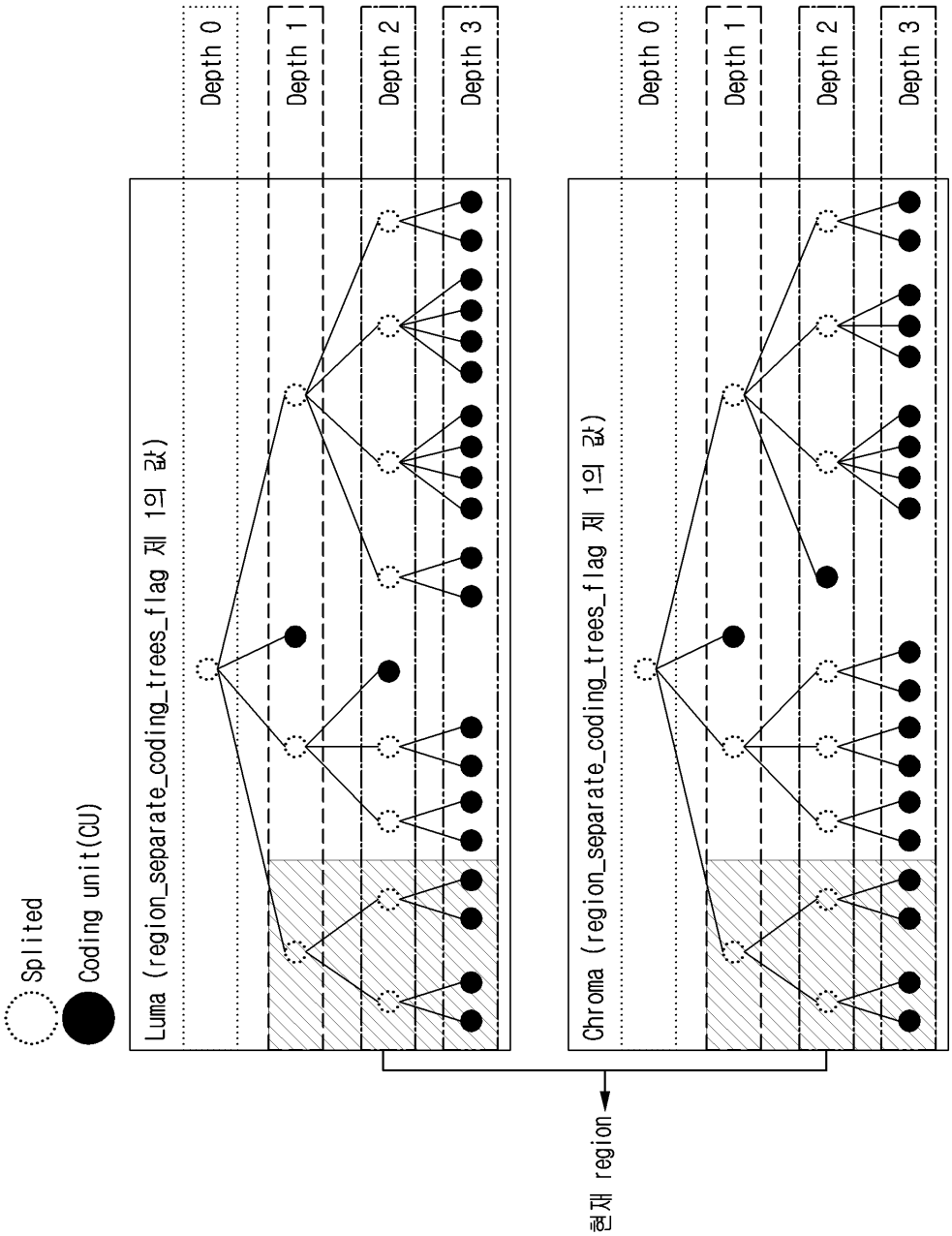
도면30


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 4의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3
Depth 4

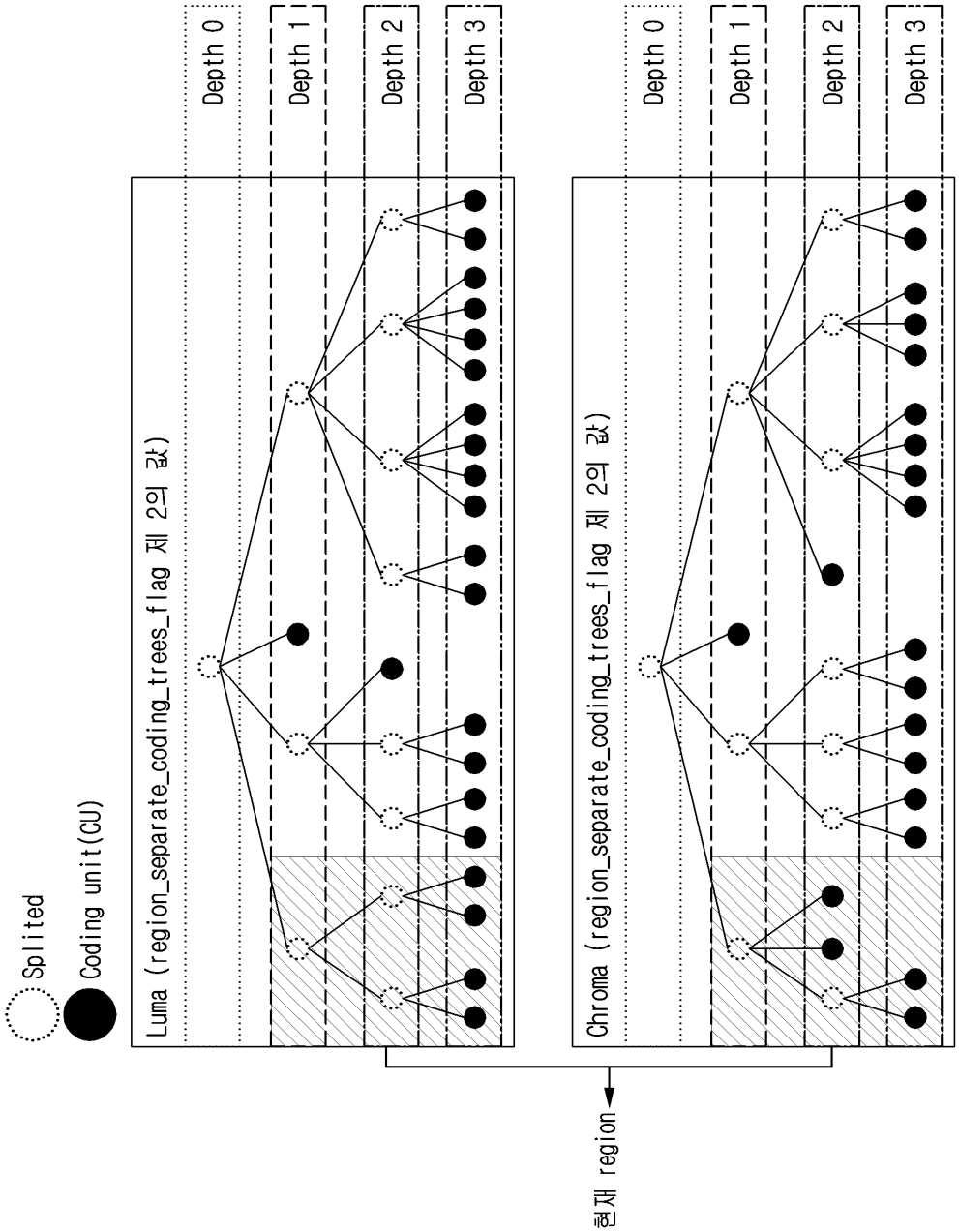
도면31


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (ctu_pred_mode_and_separate_coding_trees_flag 제 5의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4
Chroma (ctu_pred_mode_and_separate_coding_trees_flag 제 5의 값인 경우)
Depth 0 (CTU level)
Depth 1
Depth 2 (소정 블록 크기)
Depth 3
Depth 4

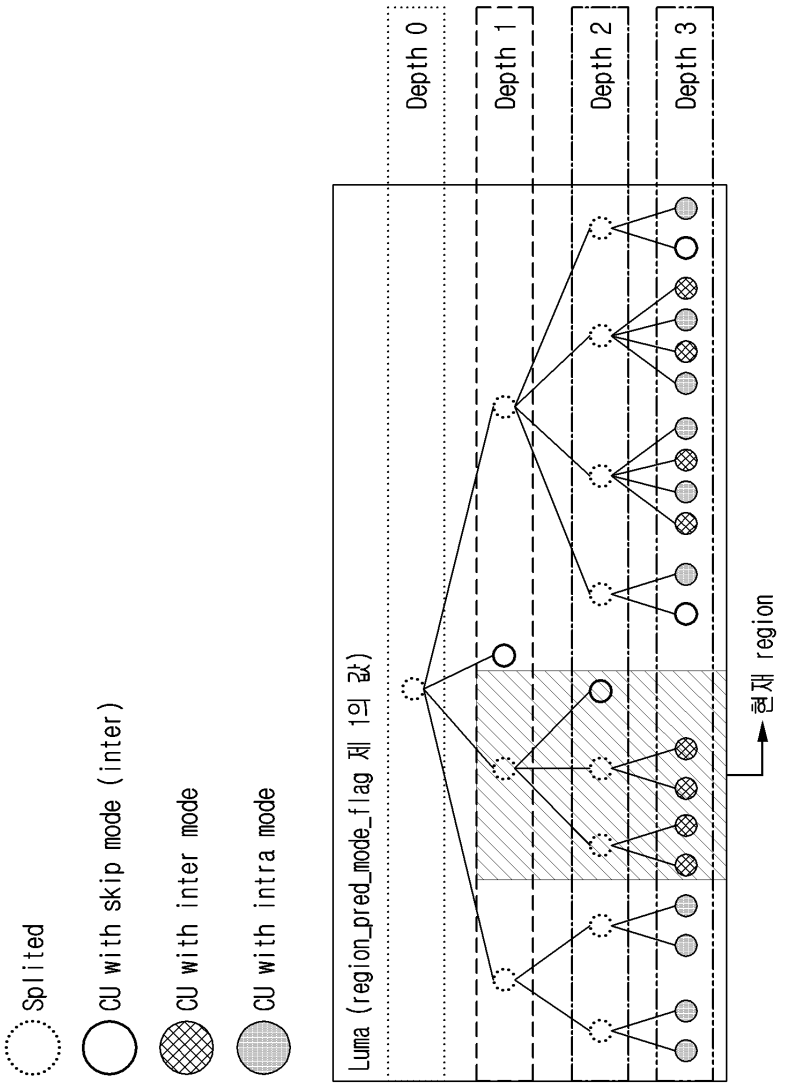
도면32


Splited
Coding unit(CU)
Luma (region_separate_coding_trees_flag 제 1의 값)
Depth 0
Depth 1
Depth 2
Depth 3
Chroma (region_separate_coding_trees_flag 제 1의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

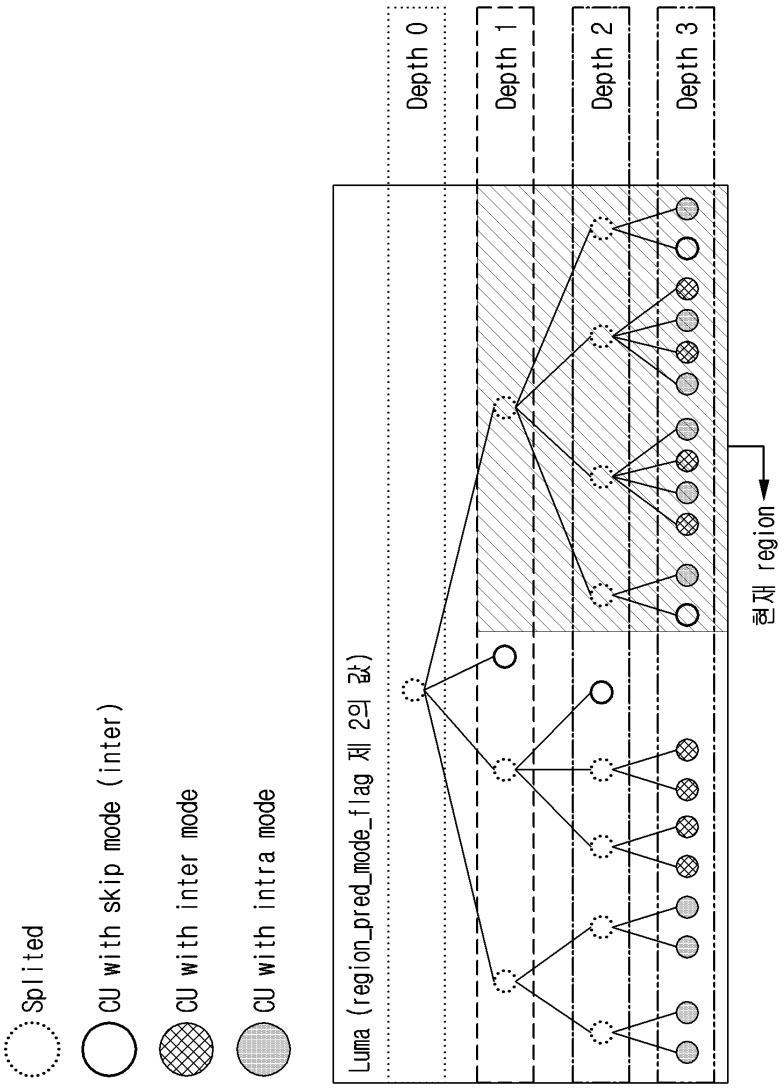
도면33


Splited
Coding unit(CU)
Luma (region_separate_coding_trees_flag 제 2의 값)
Depth 0
Depth 1
Depth 2
Depth 3
Chroma (region_separate_coding_trees_flag 제 2의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

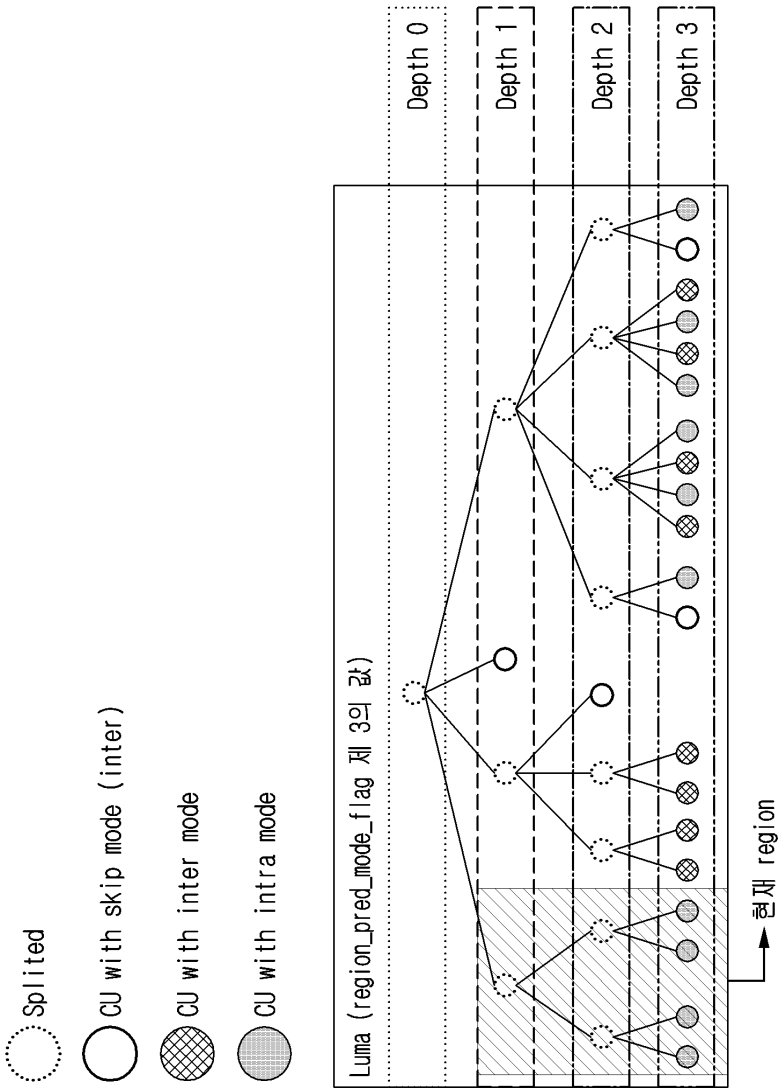
도면34


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (region_pred_mode_flag 제 1의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

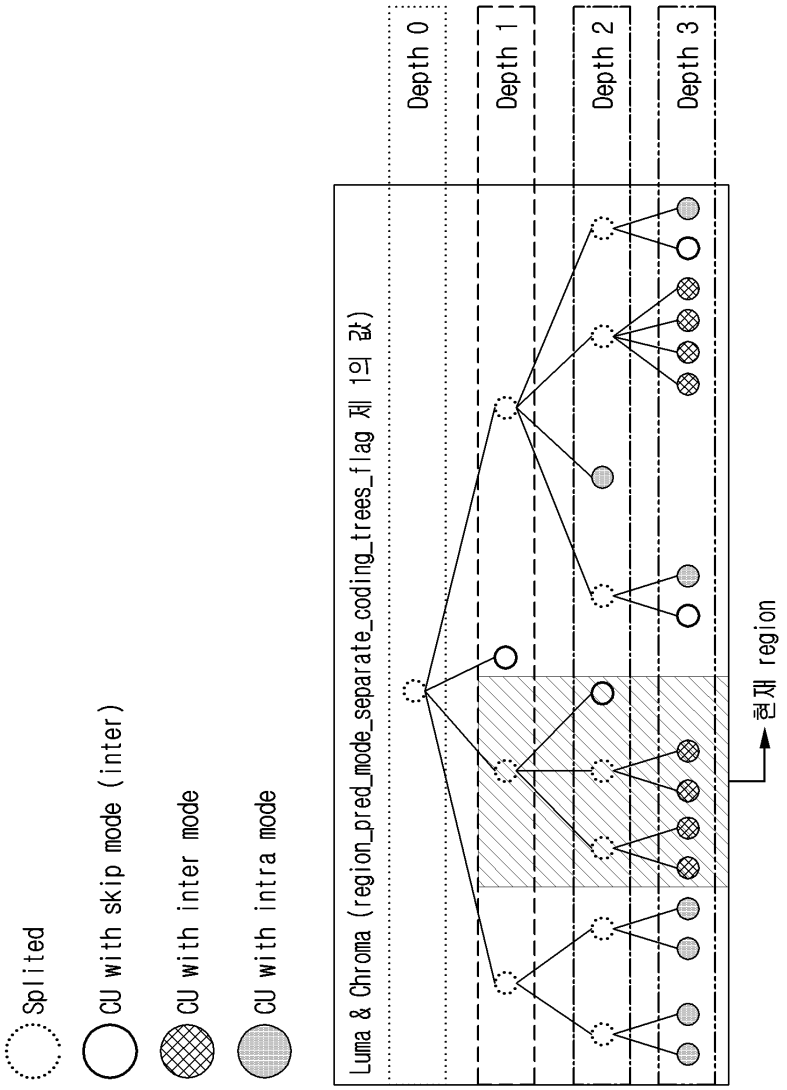
도면35


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (region_pred_mode_flag 제 2의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

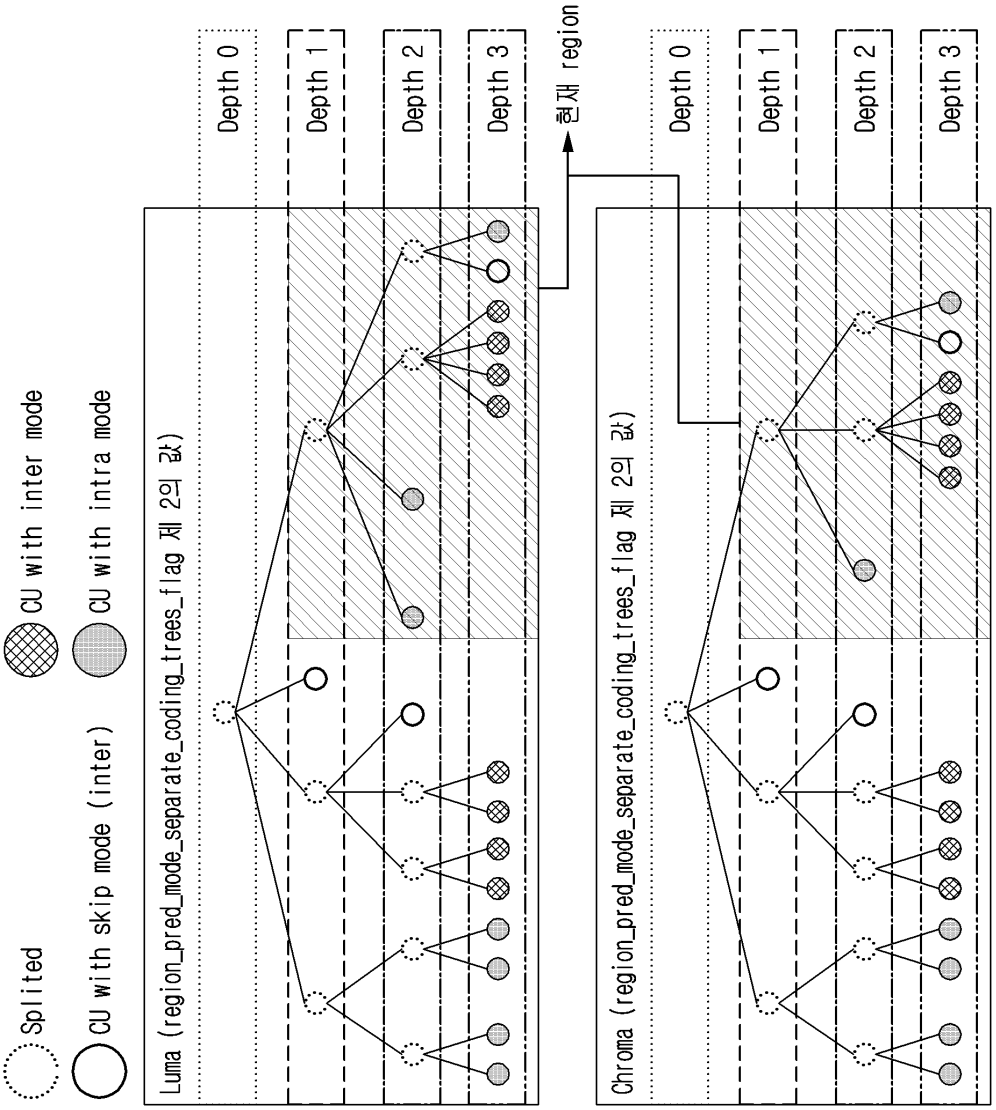
도면36


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (region_pred_mode_flag 제 3의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

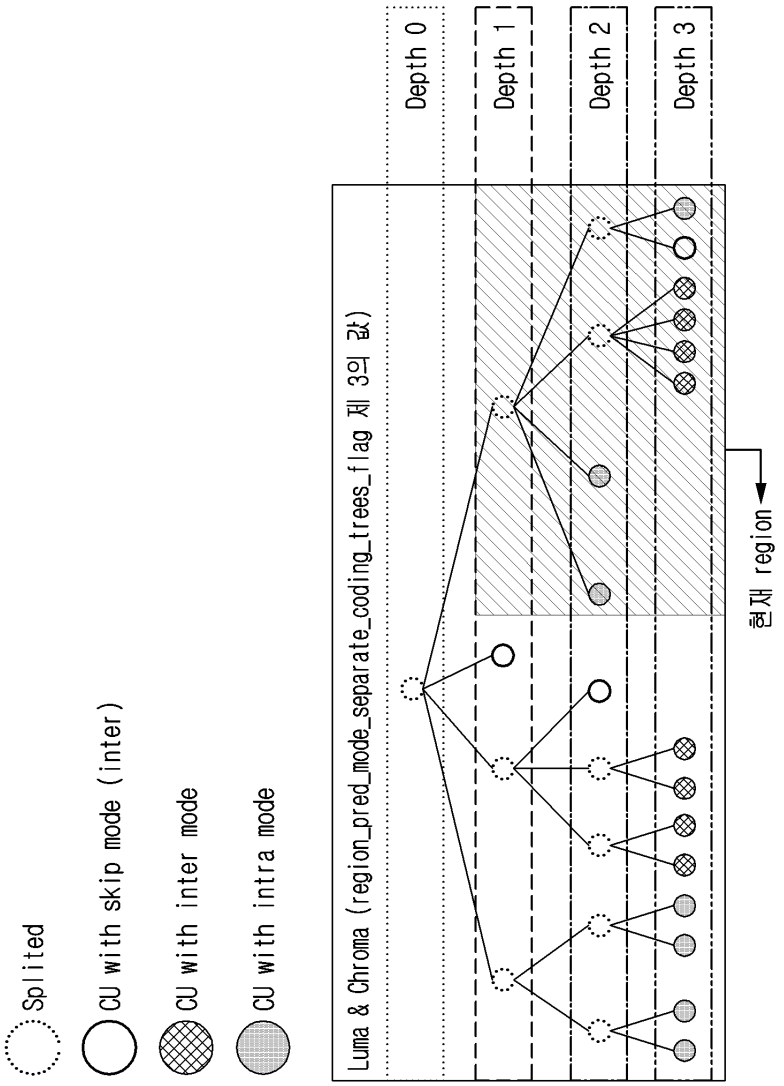
도면37


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (region_pred_mode_separate_coding_trees_flag 제 1의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

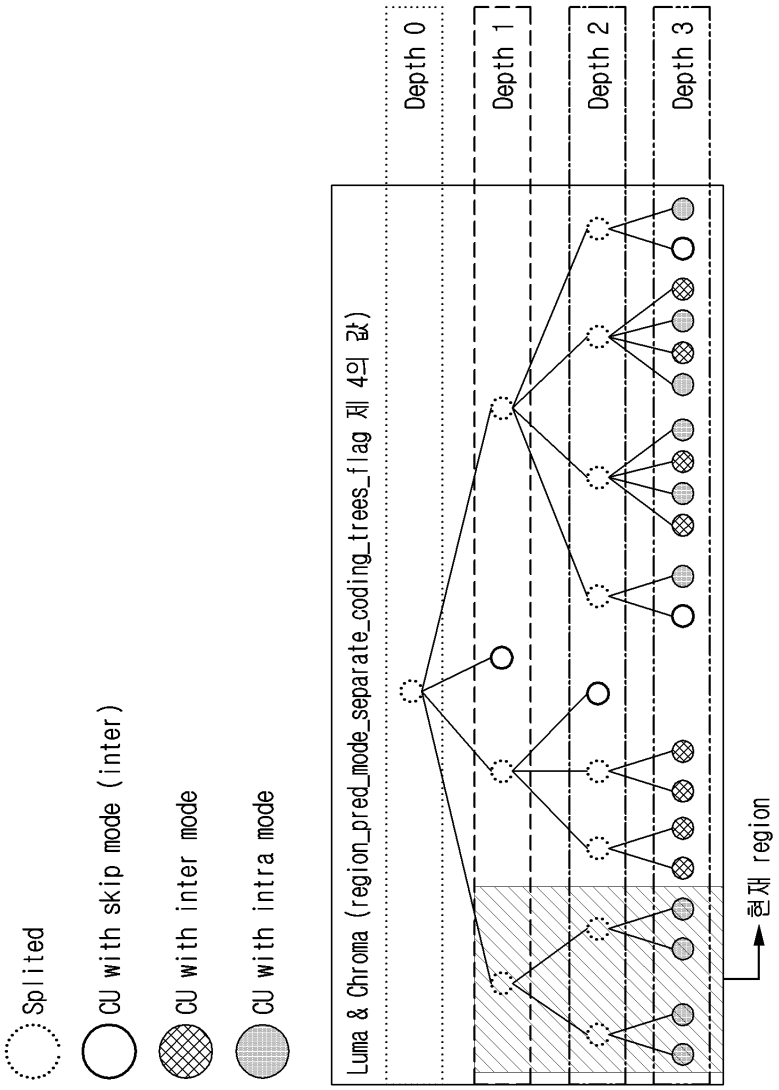
도면38


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (region_pred_mode_separate_coding_trees_flag 제 2의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region
Chroma (region_pred_mode_separate_coding_trees_flag 제 2의 값)
Depth 0
Depth 1
Depth 2
Depth 3

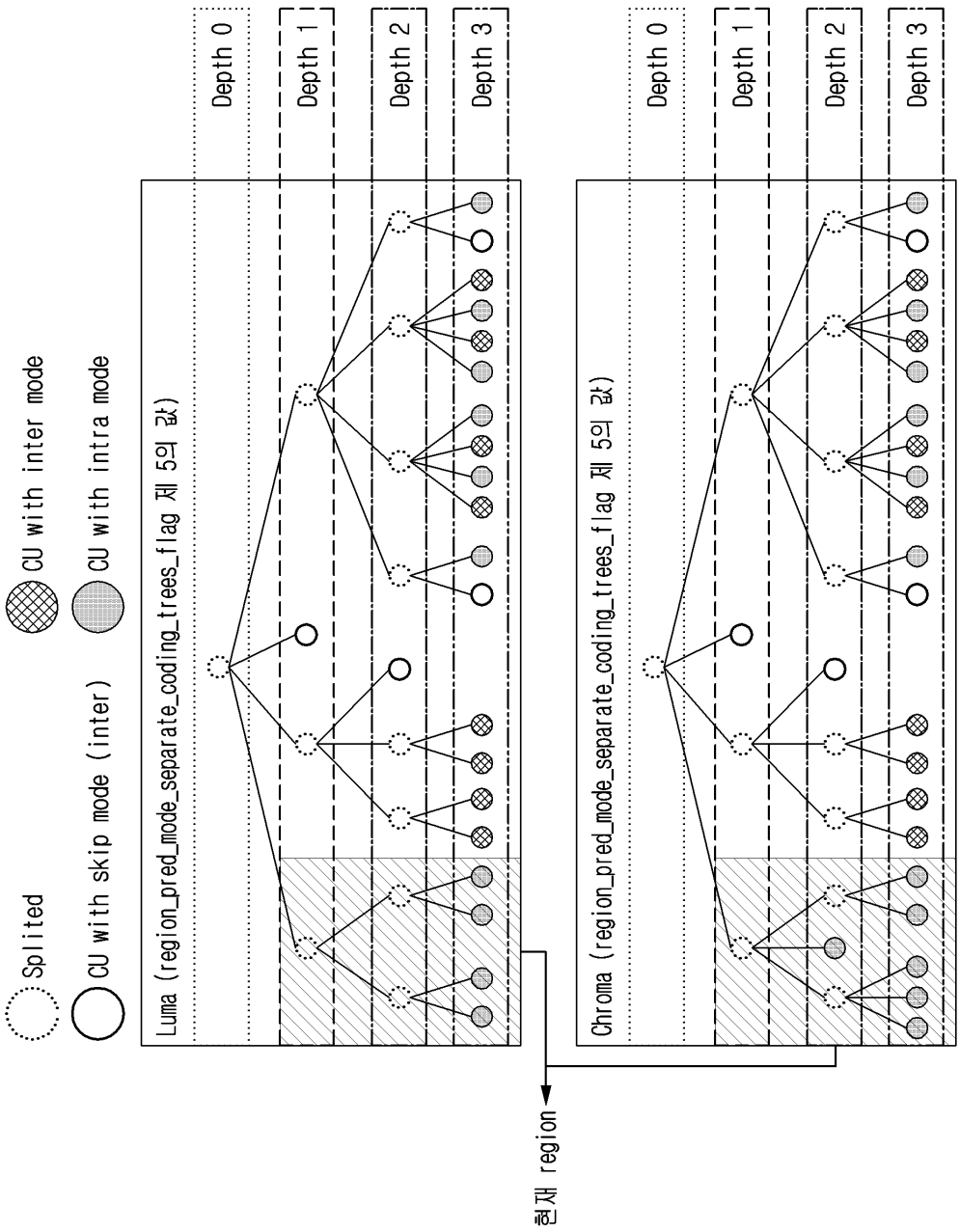
도면39


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (region_pred_mode_separate_coding_trees_flag 제 3의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

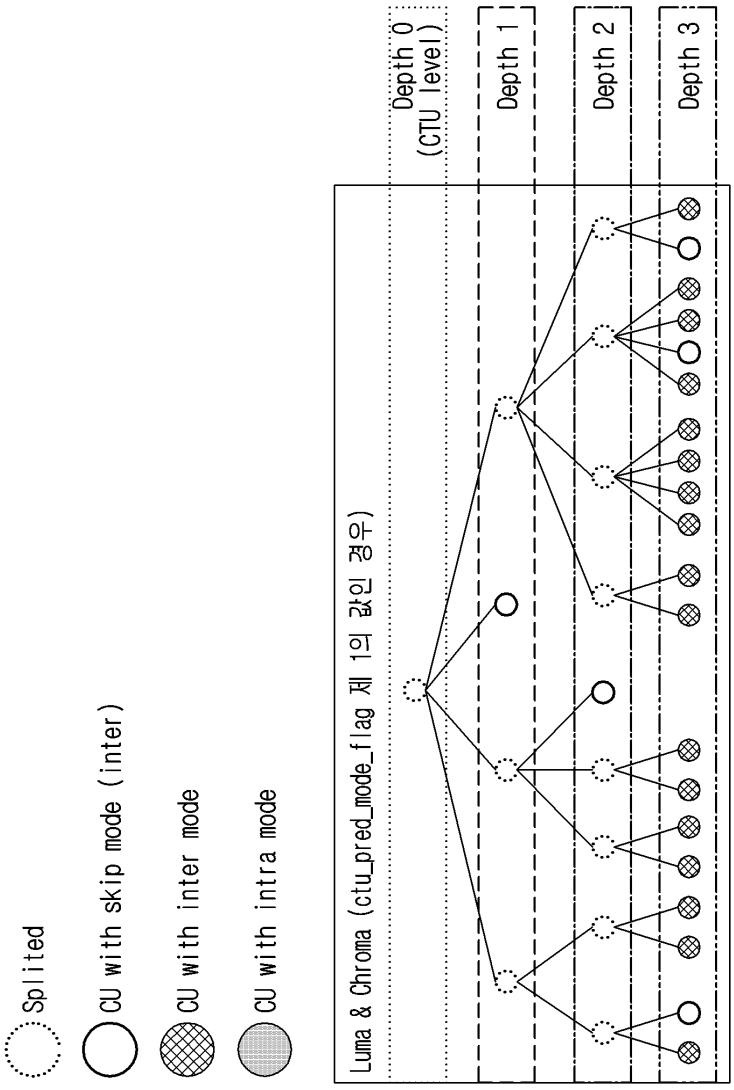
도면40


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (region_pred_mode_separate_coding_trees_flag 제 4의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

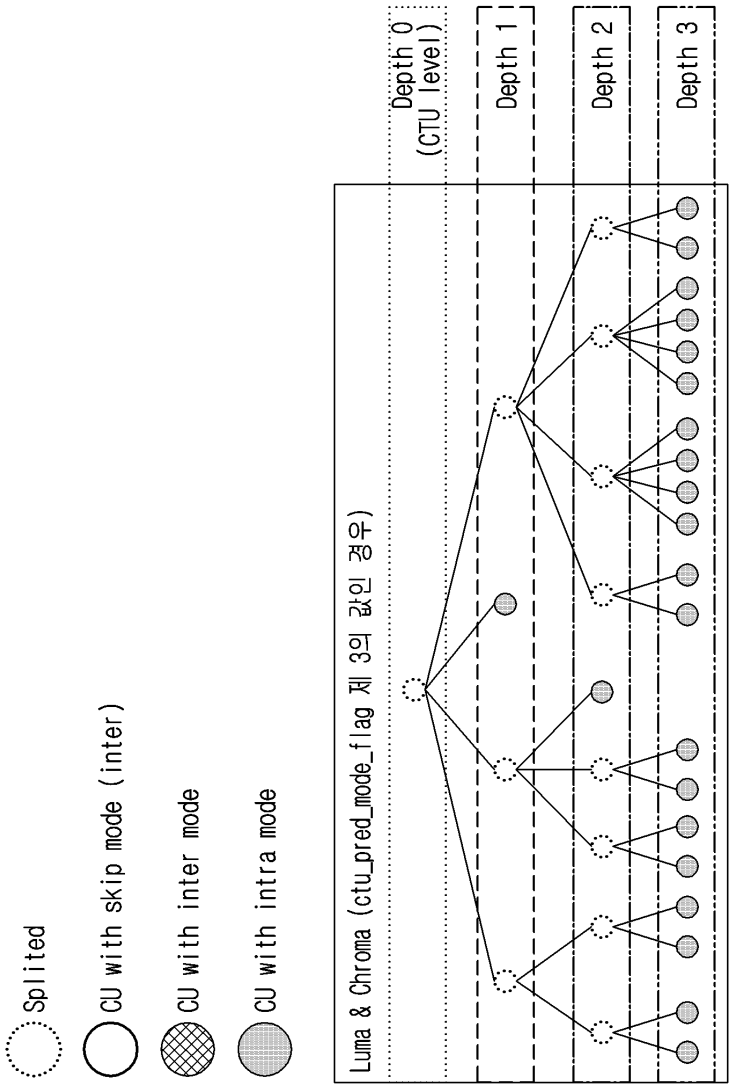
도면41


Splited
CU with inter mode
CU with skip mode (inter)
CU with intra mode
Luma (region_pred_mode_separate_coding_trees_flag 제 5의 값)
Depth 0
Depth 1
Depth 2
Depth 3
Chroma (region_pred_mode_separate_coding_trees_flag 제 5의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

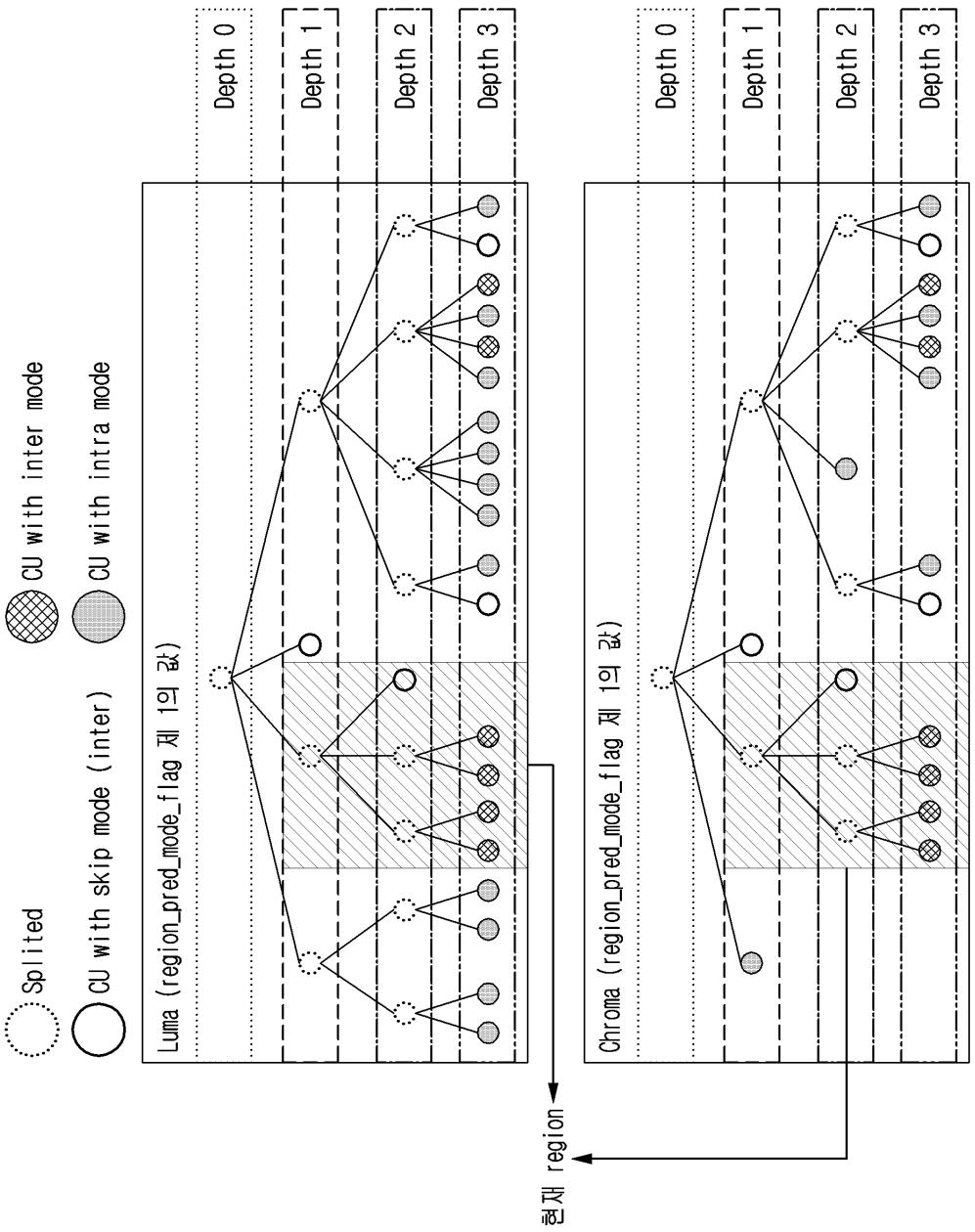
도면42


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_flag 제 1의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

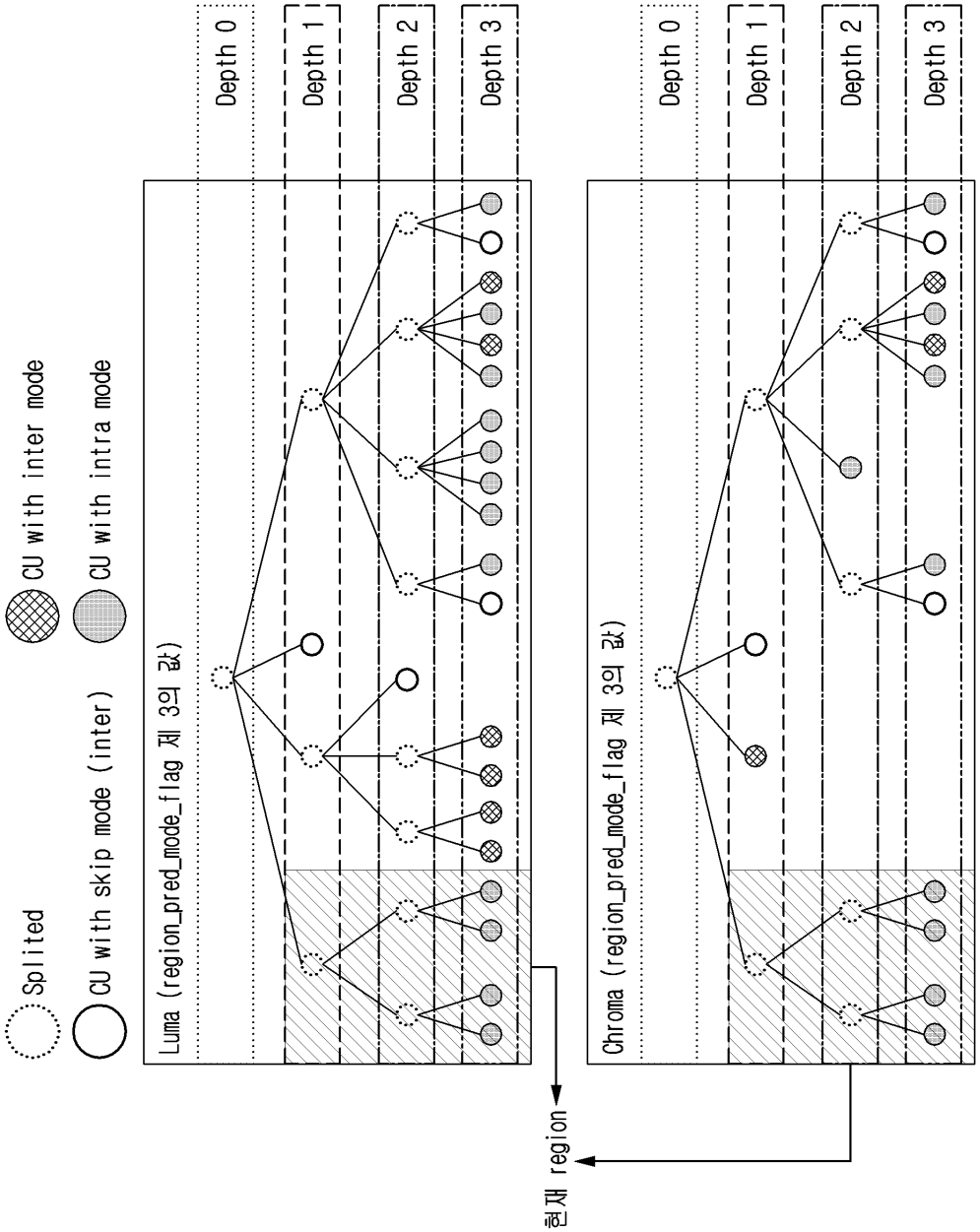
도면43


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma & Chroma (ctu_pred_mode_flag 제 3의 값인 경우)
Depth 0
(CTU level)
Depth 1
Depth 2
Depth 3

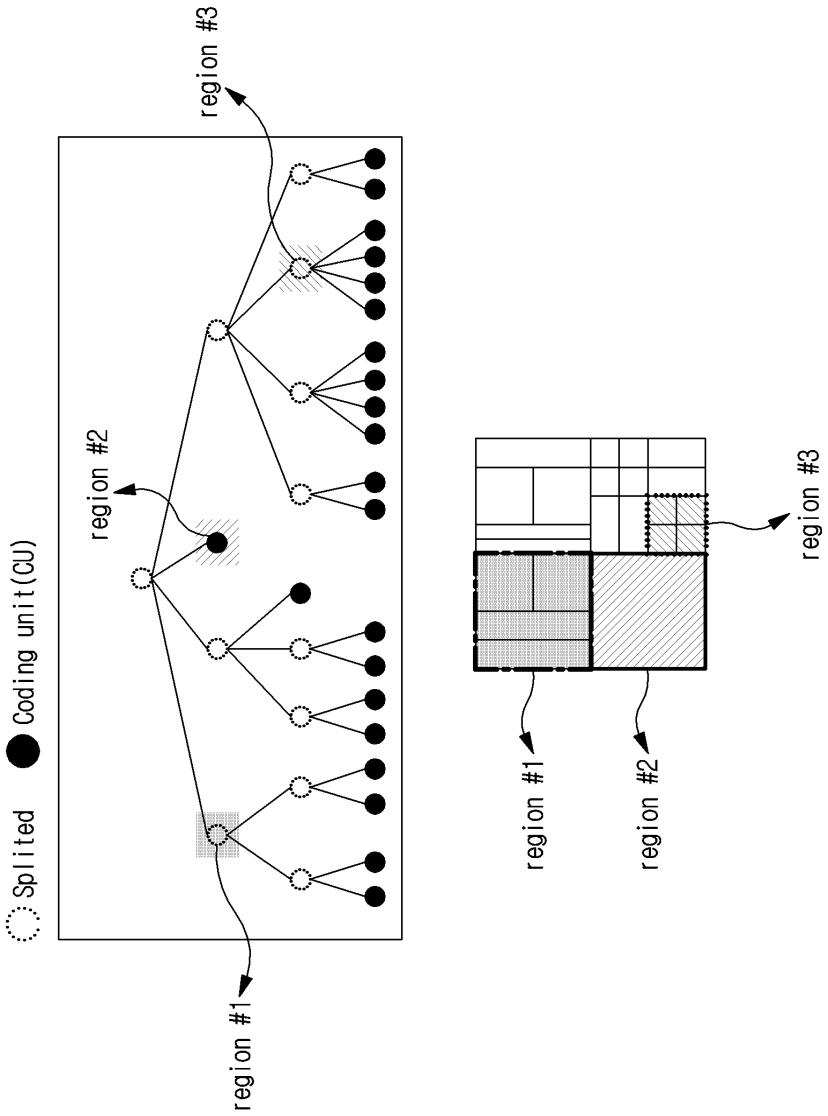
도면44


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (region_pred_mode_flag 제 1의 값)
Depth 0
Depth 1
Depth 2
Depth 3
Chroma (region_pred_mode_flag 제 1의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

도면45


Splited
CU with skip mode (inter)
CU with inter mode
CU with intra mode
Luma (region_pred_mode_flag 제 3의 값)
Depth 0
Depth 1
Depth 2
Depth 3
Chroma (region_pred_mode_flag 제 3의 값)
Depth 0
Depth 1
Depth 2
Depth 3
현재 region

도면46


region #3
region #2
region #1
Splited
Coding unit(CU)
region #3
region #1
region #2