



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0132036
(43) 공개일자 2017년12월01일

(51) 국제특허분류(Int. Cl.)

H04N 19/176 (2014.01) H04N 19/105 (2014.01)
H04N 19/107 (2014.01) H04N 19/124 (2014.01)
H04N 19/174 (2014.01)

(52) CPC특허분류

H04N 19/176 (2015.01)
H04N 19/105 (2015.01)

(21) 출원번호 10-2016-0063020

(22) 출원일자 2016년05월23일

심사청구일자 없음

(71) 출원인

가온미디어 주식회사

경기도 성남시 분당구 성남대로 884-3, 가온미디어빌딩 (야탑동)

(72) 발명자

임정윤

서울특별시 노원구 상계로5길 12, 105동 302호(상계동, 동양메이저아파트)

(74) 대리인

특허법인 다해

전체 청구항 수 : 총 23 항

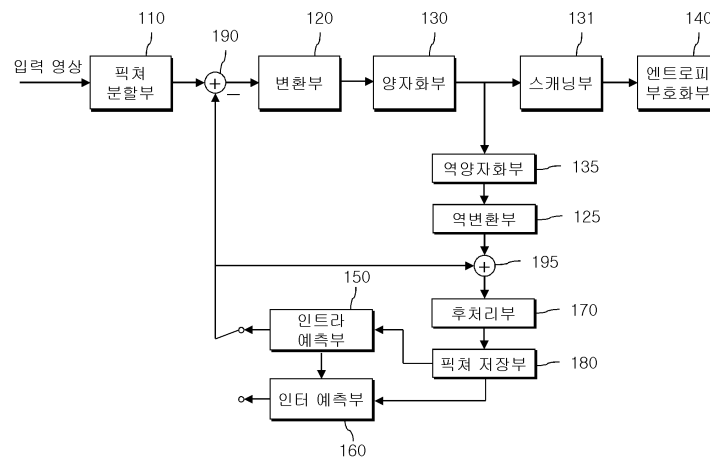
(54) 발명의 명칭 영상 처리를 위한 블록 분할 방법, 그를 이용한 영상 부호화 및 복호화 방법

(57) 요약

본 발명은 영상 처리를 위한 블록 분할 방법, 그를 이용한 영상 부호화 및 복호화 방법에 관한 것으로, 블록 분할 방법은 영상의 픽처를 인터 예측 또는 인트라 예측이 수행되는 기본 단위인 복수의 코딩 유닛들로 분할하는 단계; 및 분할된 코딩 유닛을 잔차 블록에 대한 변환이 수행되는 기본 단위인 하나 또는 2 이상의 변환 유닛들로 분할하는 단계;를 포함하고, 코딩 유닛 분할 단계는 픽처 또는 분할된 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함하며, 변환 유닛 분할 단계는 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함한다.

대표도 - 도1

10



(52) CPC특허분류

H04N 19/107 (2015.01)

H04N 19/124 (2015.01)

H04N 19/174 (2015.01)

명세서

청구범위

청구항 1

영상 처리를 위한 블록 분할 방법에 있어서,

영상의 픽처를 인터 예측(inter prediction) 또는 인트라 예측(intra prediction)이 수행되는 기본 단위인 복수의 코딩 유닛(Coding Unit)들로 분할하는 단계; 및

상기 분할된 코딩 유닛을 잔차 블록(residual block)에 대한 변환이 수행되는 기본 단위인 하나 또는 2 이상의 변환 유닛(Transform Unit)들로 분할하는 단계;를 포함하고,

상기 코딩 유닛 분할 단계는 상기 픽처 또는 상기 분할된 코딩 유닛을 이진 트리(binary tree) 구조로 분할하는 단계를 포함하며, 상기 변환 유닛 분할 단계는 상기 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함하는 영상 처리를 위한 블록 분할 방법.

청구항 2

제1항에 있어서,

상기 코딩 유닛은 정사각형 또는 직사각형의 형태를 가지는 영상 처리를 위한 블록 분할 방법.

청구항 3

제1항에 있어서,

상기 픽처를 구성하는 2 이상의 슬라이스(slice)들은 각각 복수의 코딩 트리 유닛(Coding Tree Unit)들로 분할되며,

상기 코딩 유닛 분할 단계는

상기 코딩 트리 유닛을 하나 또는 2 이상의 코딩 유닛들로 분할하는 단계를 포함하고, 상기 코딩 트리 유닛은 최대 256x256 픽셀 크기를 가지는 영상 처리를 위한 블록 분할 방법.

청구항 4

제3항에 있어서, 상기 코딩 유닛 분할 단계는

상기 코딩 트리 유닛을 정사각형의 형태를 가지는 4개의 코딩 유닛들로 분할하는 단계; 및

상기 분할된 정사각형의 코딩 유닛들 중 적어도 하나를 직사각형 형태를 가지는 2개의 코딩 유닛들로 재 분할하는 단계;를 포함하는 영상 처리를 위한 블록 분할 방법.

청구항 5

제1항에 있어서,

상기 변환 유닛은 정사각형 또는 직사각형의 형태를 가지며, 최대 64x64 픽셀 크기를 가지는 영상 처리를 위한 블록 분할 방법.

청구항 6

제1항에 있어서, 상기 변환 유닛 분할 단계는

상기 이진 트리 구조로 분할된 코딩 유닛을 2개의 변환 유닛들로 분할하는 단계를 포함하는 영상 처리를 위한 블록 분할 방법.

청구항 7

제1항에 있어서, 상기 변환 유닛 분할 단계는

상기 코딩 유닛을 이진 트리 구조로 분할하는 단계를 2회 이상 반복하여 수행하는 영상 처리를 위한 블록 분할 방법.

청구항 8

제7항에 있어서, 상기 변환 유닛 분할 단계는

직사각형의 상기 코딩 유닛을 정사각형 형태를 가지는 2개의 변환 유닛들로 분할하는 단계; 및

상기 정사각형의 변환 유닛들 중 적어도 하나를 직사각형 형태를 가지는 2개의 변환 유닛들로 재 분할하는 단계;를 포함하는 영상 처리를 위한 블록 분할 방법.

청구항 9

제7항에 있어서, 상기 변환 유닛 분할 단계는

정사각형의 상기 코딩 유닛을 직사각형 형태를 가지는 2개의 변환 유닛들로 분할하는 단계; 및

상기 직사각형의 변환 유닛들 중 적어도 하나를 정사각형 형태를 가지는 2개의 변환 유닛들로 재 분할하는 단계;를 포함하는 영상 처리를 위한 블록 분할 방법.

청구항 10

제7항에 있어서, 상기 변환 유닛 분할 단계는

직사각형의 상기 코딩 유닛을 정사각형 형태를 가지는 2개의 변환 유닛들로 분할하는 단계; 및

상기 정사각형의 변환 유닛들 중 적어도 하나를 정사각형 형태를 가지는 4개의 변환 유닛들로 재 분할하는 단계;를 포함하는 영상 처리를 위한 블록 분할 방법.

청구항 11

영상을 부호화하는 방법에 있어서,

영상의 픽처를 복수의 코딩 유닛들로 분할하는 단계;

상기 분할된 코딩 유닛을 기본 단위로 하여, 인터 예측 또는 인트라 예측을 수행하는 단계;

상기 분할된 코딩 유닛을 하나 또는 2 이상의 변환 유닛들로 분할하는 단계; 및

상기 분할된 변환 유닛을 기본 단위로 하여, 상기 인터 예측 또는 인트라 예측에 따른 잔차 블록에 대해 변환을 수행하는 단계를 포함하고,

상기 코딩 유닛 분할 단계는 상기 픽처 또는 상기 분할된 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함 하며, 상기 변환 유닛 분할 단계는 상기 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함하는 영상 부호화 방법.

청구항 12

제11항에 있어서,

상기 픽처를 구성하는 2 이상의 슬라이스들은 각각 복수의 코딩 트리 유닛들로 분할되며,

상기 코딩 유닛 분할 단계는

상기 코딩 트리 유닛을 하나 또는 2 이상의 코딩 유닛들로 분할하는 단계를 포함하고, 상기 코딩 트리 유닛은 최대 256x256 픽셀 크기를 가지는 영상 부호화 방법.

청구항 13

제12항에 있어서, 상기 코딩 유닛 분할 단계는

상기 코딩 트리 유닛을 정사각형의 형태를 가지는 4개의 코딩 유닛들로 분할하는 단계; 및

상기 분할된 정사각형의 코딩 유닛들 중 적어도 하나를 직사각형 형태를 가지는 2개의 코딩 유닛들로 재 분할하는 단계;를 포함하는 영상 부호화 방법.

청구항 14

제11항에 있어서,

상기 변환 유닛은 정사각형 또는 직사각형의 형태를 가지며, 최대 64x64 픽셀 크기를 가지는 영상 부호화 방법.

청구항 15

제11항에 있어서, 상기 변환 유닛 분할 단계는

상기 이진 트리 구조로 분할된 코딩 유닛을 2개의 변환 유닛들로 분할하는 단계를 포함하는 영상 부호화 방법.

청구항 16

제11항에 있어서, 상기 변환 유닛 분할 단계는

상기 코딩 유닛을 이진 트리 구조로 분할하는 단계를 2회 이상 반복하여 수행하는 영상 부호화 방법.

청구항 17

제11항에 있어서,

상기 코딩 유닛이 2Nx2N 픽셀 크기의 형태를 가지는 경우,

상기 변환 유닛 분할 단계는 2Nx2N, NxN, Nx2N, 2NxN 픽셀 크기의 변환 유닛 순서로 RDO(Rate distortion Optimization)를 수행하여 상기 변환 유닛의 분할 구조를 결정하는 단계를 포함하는 영상 부호화 방법.

청구항 18

제11항에 있어서,

상기 코딩 유닛이 Nx2N 또는 2NxN 픽셀 크기의 형태를 가지는 경우,

상기 변환 유닛 분할 단계는 Nx2N 또는 2NxN, NxN, N/2xN 또는 NxN/2, N/2xN/2 픽셀 크기의 변환 유닛 순서로 RDO를 수행하여 상기 변환 유닛의 분할 구조를 결정하는 단계를 포함하는 영상 부호화 방법.

청구항 19

영상을 복호화하는 방법에 있어서,

부호화된 비트스트림을 입력받는 단계;

상기 입력된 비트스트림에 대해 역양자화 및 역변환을 수행하여 잔차 블록을 획득하는 단계;

인터 예측 또는 인트라 예측을 수행하여 예측 블록을 획득하는 단계; 및

상기 획득된 잔차 블록과 예측 블록을 합하여 영상을 복원하는 단계를 포함하고,

상기 인터 예측 또는 인트라 예측이 수행되는 기본 단위인 코딩 유닛은 이진 트리 구조를 이용하여 코딩 트리 유닛으로부터 분할된 블록이며, 상기 역변환이 수행되는 기본 단위인 변환 유닛은 이진 트리 구조를 이용하여 상기 코딩 유닛으로부터 분할된 블록인 영상 복호화 방법.

청구항 20

제19항에 있어서,

상기 코딩 트리 유닛은 최대 256x256 픽셀 크기를 가지며, 상기 변환 유닛은 최대 64x64 픽셀 크기를 가지는 영상 복호화 방법.

청구항 21

제19항에 있어서,

상기 코딩 트리 유닛은 정사각형의 형태를 가지는 4개의 코딩 유닛들로 분할되며, 상기 분할된 정사각형의 코딩 유닛들 중 적어도 하나는 직사각형 형태를 가지는 2개의 코딩 유닛들로 재 분할되는 영상 복호화 방법.

청구항 22

제19항에 있어서,

상기 이진 트리 구조로 분할된 직사각형의 코딩 유닛은 정사각형 형태를 가지는 2개의 변환 유닛들로 분할되는 영상 복호화 방법.

청구항 23

제19항에 있어서, 상기 변환 유닛은

상기 코딩 유닛을 이진 트리 구조로 분할하는 과정을 2회 이상 반복함에 의해 형성되는 블록인 영상 복호화 방법.

발명의 설명

기술 분야

[0001] 본 발명은 영상 부호화 및 복호화에 관한 것으로, 더욱 상세하게는 동영상 픽처(picture)를 복수의 블록(block)들로 분할하여 부호화 및 복호화를 수행하는 방법 및 그 장치에 관한 것이다.

배경 기술

[0002] 영상 압축 방법에서는 하나의 픽처(picture)를 소정 크기를 가지는 복수의 블록(block)으로 구분하여 부호화를 수행한다. 또한, 압축 효율을 높이기 위해 픽처들 사이의 중복도를 제거하는 인터 예측(inter prediction) 및 인트라 예측(intra prediction) 기술을 이용한다.

[0003] 이 경우, 인트라 예측과 인터 예측을 이용하여 잔차 신호(residual signal)를 만들며, 잔차 신호를 구하는 이유는 잔차 신호를 가지고 코딩을 했을 때 데이터의 양이 작아서 데이터 압축률이 높아지고, 예측이 좋을수록 잔차 신호의 값이 작게 되기 때문이다.

[0004] 인트라 예측 방법은 현재 블록의 주변의 픽셀을 이용하여 현재 블록의 데이터를 예측한다. 실제 값과 예측 값의 차이를 잔차 신호 블록이라고 한다. HEVC의 경우는 인트라 예측 방법이 기존 H.264/AVC에서 사용되던 9개의 예측모드에서 35개의 예측모드로 증가하여 더욱 세분화하여 예측을 한다.

[0005] 인터 예측 방법의 경우는 현재 블록을 주변의 픽처들 내의 블록들과 비교하여 가장 비슷한 블록을 찾는다. 이때 찾은 블록에 대한 위치정보(V_x , V_y)를 모션 벡터라고 한다. 현재블록과 모션 벡터에 의하여 예측된 예측블록간의 블록내 화소 값들의 차이를 잔차 신호(residual signal) 블록이라고 한다 (motion-compensated residual block).

[0006] 이와 같이 인트라 예측과 인터 예측이 더 세분화 되어 잔차 신호의 데이터 양이 줄어들고 있으나, 동영상의 처리를 위한 연산량은 크게 증가하였다.

[0007] 특히, 영상 부호화 및 복호화를 위한 픽처 내 분할 구조를 결정하는 과정에서의 복잡도 증가량으로 인해 파이프라인 구현 등에 있어서의 어려움이 존재하며, 기존의 블록 분할 방법 및 그에 따라 분할된 블록의 크기가 고해상도 영상의 부호화에는 적합하지 않을 수 있다.

발명의 내용

해결하려는 과제

[0008] 본 발명은 상기와 같은 과제를 해결하기 위한 것으로, 고해상도 영상의 부호화 및 복호화에 적합한 코딩 유닛 분할 방법, 그를 이용한 영상 부호화 및 복호화 방법을 제공하는데 그 목적이 있다.

과제의 해결 수단

[0009] 상기와 같은 과제를 해결하기 위한 본 발명의 일실시예에 따른 영상 처리를 위한 블록 분할 방법은, 영상의 픽

처를 인터 예측(inter prediction) 또는 인트라 예측(intra prediction)이 수행되는 기본 단위인 복수의 코딩 유닛(Coding Unit)들로 분할하는 단계; 및 상기 분할된 코딩 유닛을 잔차 블록(residual block)에 대한 변환이 수행되는 기본 단위인 하나 또는 2 이상의 변환 유닛(Transform Unit)들로 분할하는 단계;를 포함하고, 상기 코딩 유닛 분할 단계는 상기 픽처 또는 상기 분할된 코딩 유닛을 이진 트리(binary tree) 구조로 분할하는 단계를 포함하며, 상기 변환 유닛 분할 단계는 상기 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함한다.

[0010] 본 발명의 일실시예에 따른 영상 부호화 방법은, 영상의 픽처를 복수의 코딩 유닛들로 분할하는 단계; 상기 분할된 코딩 유닛을 기본 단위로 하여, 인터 예측 또는 인트라 예측을 수행하는 단계; 상기 분할된 코딩 유닛을 하나 또는 2 이상의 변환 유닛들로 분할하는 단계; 및 상기 분할된 변환 유닛을 기본 단위로 하여, 상기 인터 예측 또는 인트라 예측에 따른 잔차 블록에 대해 변환을 수행하는 단계를 포함하고, 상기 코딩 유닛 분할 단계는 상기 픽처 또는 상기 분할된 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함하며, 상기 변환 유닛 분할 단계는 상기 코딩 유닛을 이진 트리 구조로 분할하는 단계를 포함한다.

[0011] 본 발명의 일실시예에 따른 영상 복호화 방법은, 부호화된 비트스트림을 입력받는 단계; 상기 입력된 비트스트림에 대해 역양자화 및 역변환을 수행하여 잔차 블록을 획득하는 단계; 인터 예측 또는 인트라 예측을 수행하여 예측 블록을 획득하는 단계; 상기 획득된 잔차 블록과 예측 블록을 합하여 영상을 복원하는 단계를 포함하고, 상기 인터 예측 또는 인트라 예측이 수행되는 기본 단위인 코딩 유닛은 이진 트리 구조를 이용하여 코딩 트리 유닛으로부터 분할된 블록이며, 상기 역변환이 수행되는 기본 단위인 변환 유닛은 이진 트리 구조를 이용하여 상기 코딩 유닛으로부터 분할된 블록이다.

[0012] 한편, 상기한 방법들은 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체로 구현될 수 있다.

발명의 효과

[0013] 본 발명의 실시예에 따르면, 인터 예측 또는 인트라 예측이 수행되는 기본 단위인 코딩 유닛과 변환이 수행되는 기본 단위인 변환 유닛을 이진 트리 구조로 분할하여 영상의 부호화 및 복호화에 이용함으로써, 고해상도 영상에 대한 코딩 효율을 향상시킬 수 있다.

도면의 간단한 설명

[0014] 도 1은 본 발명의 일실시예에 따른 영상 부호화 장치의 구성을 나타내는 블록도이다.
 도 2 내지 도 5는 영상을 블록 단위로 분할하여 처리하는 방법에 대한 제1 실시예를 설명하기 위한 도면이다.
 도 6은 영상 부호화 장치에서 인터 예측을 수행하는 방법에 대한 일실시예를 설명하기 위한 블록도이다.
 도 7은 본 발명의 일실시예에 따른 영상 복호화 장치의 구성을 나타내는 블록도이다.
 도 8은 영상 복호화 장치에서 인터 예측을 수행하는 방법에 대한 일실시예를 설명하기 위한 블록도이다.
 도 9는 영상을 블록 단위로 분할하여 처리하는 방법에 대한 제2 실시예를 설명하기 위한 도면이다.
 도 10은 영상을 블록 단위로 분할하여 처리하기 위해 사용되는 선택스(syntax) 구조에 대한 일실시예를 나타내는 도면이다.
 도 11은 영상을 블록 단위로 분할하여 처리하는 방법에 대한 제3 실시예를 설명하기 위한 도면이다.
 도 12는 코딩 유닛을 이진 트리 구조로 분할하여 변환 유닛을 구성하는 방법에 대한 일실시예를 설명하기 위한 도면이다.
 도 13은 영상을 블록 단위로 분할하여 처리하는 방법에 대한 제4 실시예를 설명하기 위한 도면이다.
 도 14 내지 도 16은 영상을 블록 단위로 분할하여 처리하는 방법에 대한 또 다른 실시예들을 설명하기 위한 도면들이다.
 도 17 및 도 18은 RDO(Rate distortion Optimization)를 수행하여 변환 유닛의 분할 구조를 결정하는 방법에 대한 실시예들을 설명하기 위한 도면들이다.

발명을 실시하기 위한 구체적인 내용

- [0015] 이하, 도면을 참조하여 본 발명의 실시 형태에 대하여 구체적으로 설명한다. 본 명세서의 실시 예를 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 명세서의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략한다.
- [0016] 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. 아울러, 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.
- [0017] 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.
- [0018] 또한 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.
- [0019] 또한, 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 요소를 포함한 구조도 본 발명의 권리범위에 포함된다.
- [0020] 도 1은 본 발명의 일실시예에 따른 영상 부호화 장치의 구성을 블록도로 도시한 것으로, 영상 부호화 장치(10)는 픽처 분할부(110), 변환부(120), 양자화부(130), 스캐닝부(131), 엔트로피 부호화부(140), 인트라 예측부(150), 인터 예측부(160), 역양자화부(135), 역변환부(125), 후처리부(170), 픽처 저장부(180), 감산부(190) 및 가산부(195)를 포함한다.
- [0021] 도 1을 참조하면, 픽처 분할부(110)는 입력되는 비디오 신호를 분석하여 픽처를 코딩 유닛으로 분할하여 예측 모드를 결정하고, 상기 코딩 유닛별로 예측 유닛의 크기를 결정한다.
- [0022] 또한, 픽처 분할부(110)는 부호화할 예측 유닛을 예측 모드(또는 예측 방법)에 따라 인트라 예측부(150) 또는 인터 예측부(160)로 보낸다. 또한, 픽처 분할부(110)는 부호화할 예측 유닛을 감산부(190)로 보낸다.
- [0023] 여기서, 영상의 픽처(picture)는 복수의 슬라이스로 구성되고, 슬라이스는 픽처를 분할하는 기본 단위인 복수의 코딩 트리 유닛(Coding Tree Unit: CTU)들로 분할될 수 있다.
- [0024] 상기 코딩 트리 유닛은 인터 예측(inter prediction) 또는 인트라 예측(intra prediction)이 수행되는 기본 단위인 하나 또는 2 이상의 코딩 유닛(Coding Unit: CU)들로 분할될 수 있다.
- [0025] 코딩 유닛(CU)은 예측이 수행되는 기본 단위인 하나 또는 그 이상의 예측 유닛(Prediction unit: PU)들로 분할될 수 있다.
- [0026] 이 경우, 부호화 장치(10)는 상기 분할된 코딩 유닛(CU)들 각각에 대해 인터 예측과 인트라 예측 중 어느 하나를 예측 방법으로 결정하나, 각각의 예측 유닛(PU)에 대해 서로 다르게 예측 블록을 생성할 수 있다.
- [0027] 한편, 코딩 유닛(CU)은 잔차 블록(residual block)에 대한 변환이 수행되는 기본 단위인 하나 또는 2 이상의 변환 유닛(Transform Unit: TU)들로 분할될 수 있다.
- [0028] 이 경우, 픽처 분할부(110)는 상기과 같이 분할된 블록 단위(예를 들어, 예측 유닛(PU) 또는 변환 유닛(TU))로 영상 데이터를 감산부(190)에 전달할 수 있다.
- [0029] 도 2를 참조하면, 최대 256x256 픽셀 크기를 가지는 코딩 트리 유닛(CTU)는 쿼드 트리(quad tree) 구조로 분할되어, 정사각형의 형태를 가지는 4개의 코딩 유닛(CU)들로 분할될 수 있다.

- [0030] 상기 정사각형의 형태를 가지는 4개의 코딩 유닛(CU)들은 각각 쿼드 트리 구조로 재 분할될 수 있으며, 상기과 같이 쿼드 트리 구조로 분할되는 코딩 유닛(CU)의 깊이(Depth)는 0부터 3 중 어느 하나의 정수 값을 가질 수 있다.
- [0031] 코딩 유닛(CU)은 예측 모드에 따라 하나 또는 2 이상의 예측 유닛(PU)들로 분할될 수 있다.
- [0032] 인트라 예측 모드의 경우, 코딩 유닛(CU)의 크기가 $2N \times 2N$ 일 때, 예측 유닛(PU)은 도 3의 (a)에 도시된 $2N \times 2N$ 또는 도 3의 (b)에 도시된 $N \times N$ 의 크기를 가질 수 있다.
- [0033] 한편, 인터 예측 모드의 경우, 코딩 유닛(CU)의 크기가 $2N \times 2N$ 일 때, 예측 유닛(PU)은 도 4의 (a)에 도시된 $2N \times 2N$, 도 4의 (b)에 도시된 $2N \times N$, 도 4의 (c)에 도시된 $N \times 2N$, 도 4의 (d)에 도시된 $N \times N$, 도 4의 (e)에 도시된 $2N \times nU$, 도 4의 (f)에 도시된 $2N \times nD$, 도 4의 (g)에 도시된 $nL \times 2N$ 및 도 4의 (h)에 도시된 $nR \times 2N$ 중 어느 하나의 크기를 가질 수 있다.
- [0034] 도 5를 참조하면, 코딩 유닛(CU)은 쿼드 트리(quad tree) 구조로 분할되어, 정사각형의 형태를 가지는 4개의 변환 유닛(TU)들로 분할될 수 있다.
- [0035] 상기 정사각형의 형태를 가지는 4개의 변환 유닛(TU)들은 각각 쿼드 트리 구조로 재 분할될 수 있으며, 상기과 같이 쿼드 트리 구조로 분할되는 변환 유닛(TU)의 깊이(Depth)는 0부터 3 중 어느 하나의 정수 값을 가질 수 있다.
- [0036] 여기서, 코딩 유닛(CU)이 인터 예측 모드인 경우, 해당 코딩 유닛(CU)으로부터 분할된 예측 유닛(PU)과 변환 유닛(TU)은 서로 독립적인 분할 구조를 가질 수 있다.
- [0037] 코딩 유닛(CU)이 인트라 예측 모드인 경우, 해당 코딩 유닛(CU)으로부터 분할된 변환 유닛(TU)은 예측 유닛(PU)의 크기보다 클 수 없다.
- [0038] 또한, 상기과 같이 분할되는 변환 유닛(TU)은 최대 64×64 픽셀 크기를 가질 수 있다.
- [0039] 변환부(120)는 입력된 예측 유닛(PU)의 원본 블록과 인트라 예측부(150) 또는 인터 예측부(160)에서 생성된 예측 블록 사이의 잔차 신호인 잔차 블록을 변환하며, 상기 변환은 변환 유닛(TU)을 기본 단위로 하여 수행될 수 있다.
- [0040] 상기 변환 과정에서 예측 모드(intra or inter)에 따라 서로 다른 변환 매트릭스가 결정될 수 있으며, 인트라 예측의 잔차 신호는 인트라 예측 모드에 따라 방향성을 가지므로 인트라 예측 모드에 따라 적응적으로 변환 매트릭스가 결정될 수 있다.
- [0041] 변환 단위는 2개(수평, 수직)의 1차원 변환 매트릭스에 의해 변환될 수 있으며, 예를 들어 인터 예측의 경우에는 미리 결정된 1개의 변환 매트릭스가 결정될 수 있다.
- [0042] 한편, 인트라 예측의 경우, 인트라 예측 모드가 수평인 경우에는 잔차 블록이 수직방향으로의 방향성을 가질 확률이 높아지므로, 수직방향으로는 DCT 기반의 정수 매트릭스를 적용하고, 수평방향으로는 DST 기반 또는 KLT 기반의 정수 매트릭스를 적용한다. 인트라 예측 모드가 수직인 경우에는 수직방향으로는 DST 기반 또는 KLT 기반의 정수 매트릭스를, 수평 방향으로는 DCT 기반의 정수 매트릭스를 적용할 수 있다.
- [0043] 또한, DC 모드의 경우에는 양방향 모두 DCT 기반 정수 매트릭스를 적용할 수 있다.
- [0044] 그리고, 인트라 예측의 경우, 변환 유닛(TU)의 크기에 기초하여 변환 매트릭스가 적응적으로 결정될 수도 있다.
- [0045] 양자화부(130)는 상기 변환 매트릭스에 의해 변환된 잔차 블록의 계수들을 양자화하기 위한 양자화 스텝 사이즈를 결정하며, 양자화 스텝 사이즈는 미리 정해진 크기 이상의 양자화 유닛별로 결정될 수 있다.
- [0046] 양자화 유닛의 크기는 8×8 또는 16×16 일 수 있으며, 양자화부(130)는 양자화 스텝 사이즈 및 예측 모드에 따라 결정되는 양자화 매트릭스를 이용하여 변환 블록의 계수들을 양자화한다.
- [0047] 또한, 양자화부(130)는 현재 양자화 유닛의 양자화 스텝 사이즈 예측자로서 현재 양자화 유닛에 인접한 양자화 유닛의 양자화 스텝 사이즈를 이용할 수 있다.
- [0048] 양자화부(130)는 현재 양자화 유닛의 좌측 양자화 유닛, 상측 양자화 유닛, 좌상측 양자화 유닛 순서로 검색하여 1개 또는 2개의 유효한 양자화 스텝 사이즈를 이용하여 현재 양자화 유닛의 양자화 스텝 사이즈 예측자를 생성할 수 있다.

- [0049] 예를 들어, 양자화부(130)는 상기 순서로 검색된 유효한 첫번째 양자화 스텝 사이즈를 양자화 스텝 사이즈 예측자로 결정하거나, 상기 순서로 검색된 유효한 2개의 양자화 스텝 사이즈의 평균값을 양자화 스텝 사이즈 예측자로 결정하거나, 또는 1개의 양자화 스텝 사이즈만이 유효한 경우에는 이를 양자화 스텝 사이즈 예측자로 결정할 수 있다.
- [0050] 상기 양자화 스텝 사이즈 예측자가 결정되면, 양자화부(130)는 현재 양자화 유닛의 양자화 스텝 사이즈와 양자화 스텝 사이즈 예측자 사이의 차분값을 엔트로피 부호화부(140)로 전송한다.
- [0051] 한편, 현재 코딩 유닛의 좌측 코딩 유닛, 상측 코딩 유닛, 좌상측 코딩 유닛 모두가 존재하지 않거나, 또는 최대 코딩 유닛 내의 부호화 순서 상으로 이전에 존재하는 코딩 유닛이 존재할 수 있다.
- [0052] 따라서, 현재 코딩 유닛에 인접한 양자화 유닛들과 상기 최대 코딩 유닛 내에서는 부호화 순서상 바로 이전의 양자화 유닛의 양자화 스텝 사이즈가 후보자가 될 수 있다.
- [0053] 이 경우, 1) 현재 코딩 유닛의 좌측 양자화 유닛, 2) 현재 코딩 유닛의 상측 양자화 유닛, 3) 현재 코딩 유닛의 좌상측 양자화 유닛, 4) 부호화 순서상 바로 이전의 양자화 유닛 순서로 우선순위가 설정될 수 있다. 상기 순서는 바뀔 수 있고, 상기 좌상측 양자화 유닛은 생략될 수도 있다.
- [0054] 한편, 상기와 같이 양자화된 변환 블록은 역양자화부(135)와 스케닝부(131)로 전달된다.
- [0055] 스케닝부(131)는 양자화된 변환 블록의 계수들을 스케닝하여 1차원의 양자화 계수들로 변환하며, 이 경우 양자화 후의 변환 블록의 계수 분포가 인트라 예측 모드에 의존적일 수 있으므로 스케닝 방식은 인트라 예측 모드에 따라 결정될 수 있다.
- [0056] 또한, 계수 스케닝 방식은 변환 단위의 크기에 따라 달리 결정될 수도 있고, 상기 스캔 패턴은 방향성 인트라 예측 모드에 따라 달라질 수 있으며, 이 경우 양자화 계수들의 스캔 순서는 역방향으로 스캔될 수 있다.
- [0057] 상기 양자화된 계수들이 복수의 서브셋(sub-set)들로 분할된 경우, 각각의 서브셋 내의 양자화 계수들에 동일한 스캔 패턴이 적용될 수 있으며, 서브셋 간의 스캔 패턴은 지그재그 스캔 또는 대각선 스캔이 적용될 수 있다.
- [0058] 한편, 상기 스캔 패턴은 DC를 포함하는 메인 서브셋으로부터 순방향으로 잔여 서브셋들로 스캔하는 것이 바람직하나, 그 역방향도 가능하다.
- [0059] 또한, 서브셋 내의 양자화된 계수들의 스캔 패턴과 동일하게 서브셋 간의 스캔 패턴을 설정할 수도 있으며, 서브셋 간의 스캔 패턴은 인트라 예측 모드에 따라 결정될 수 있다.
- [0060] 한편, 부호화 장치(10)는 상기 변환 유닛(PU) 내에서 0이 아닌 마지막 양자화 계수의 위치 및 각 서브셋 내의 0이 아닌 마지막 양자화 계수의 위치를 나타낼 수 있는 정보를 비트스트림에 포함시켜 복호화 장치(10)로 전송할 수 있다.
- [0061] 역양자화부(135)는 상기와 같이 양자화된 양자화 계수를 역양자화하며, 역변환부(125)는 변환 유닛(TU) 단위로 역변환을 수행하여 상기 역양자화된 변환 계수를 공간 영역의 잔차 블록으로 복원할 수 있다.
- [0062] 가산기(195)는 상기 역변환부(125)에 의해 복원된 잔차 블록과 인트라 예측부(150) 또는 인터 예측부(160)로부터의 수신된 예측 블록을 합하여 복원 블록을 생성할 수 있다.
- [0063] 또한, 후처리부(170)는 복원된 픽처에 발생하는 블록킹 효과의 제거하기 위한 디블록킹(deblocking) 필터링 과정, 화소 단위로 원본 영상과의 차이 값을 보완하기 위한 샘플 적응적 오프셋(Sample Adaptive Offset : SAO) 적용 과정 및 코딩 유닛으로 원본 영상과의 차이 값을 보완하기 위한 적응적 루프 필터링(Adaptive Loop Filtering : ALF) 과정을 수행할 수 있다.
- [0064] 디블록킹 필터링 과정은 미리 정해진 크기 이상의 크기를 갖는 예측 유닛(PU) 또는 변환 유닛(TU)의 경계에 적용될 수 있다.
- [0065] 예를 들어, 디블록킹 필터링 과정은, 필터링할 경계(boundary)를 결정하는 단계, 상기 경계에 적용할 경계 필터링 강도(boundary filtering strength)를 결정하는 단계, 디블록킹 필터의 적용 여부를 결정하는 단계, 상기 디블록킹 필터를 적용할 것으로 결정된 경우, 상기 경계에 적용할 필터를 선택하는 단계를 포함할 수 있다.
- [0066] 한편, 상기 디블록킹 필터의 적용 여부는 i) 상기 경계 필터링 강도가 0보다 큰지 여부 및 ii) 상기 필터링할 경계에 인접한 2개의 블록(P 블록, Q블록) 경계 부분에서의 화소값들이 변화 정도를 나타내는 값이 양자화 파라

미터에 의해 결정되는 제1 기준값보다 작은지 여부에 의해 결정될 수 있다.

- [0067] 상기 필터는 적어도 2개 이상인 것이 바람직하다. 블록 경계에 위치한 2개의 화소들간의 차이값의 절대값이 제2 기준값보다 크거나 같은 경우에는 상대적으로 약한 필터링을 수행하는 필터를 선택한다.
- [0068] 상기 제2 기준값은 상기 양자화 파라미터 및 상기 경계 필터링 강도에 의해 결정된다.
- [0069] 또한, 샘플 적응적 오프셋(SAO) 적용 과정은 디블록킹 필터가 적용된 영상 내의 화소와 원본 화소 간의 차이값(distortion)을 감소시키기 위한 것으로, 픽처 또는 슬라이스 단위로 샘플 적응적 오프셋(SAO) 적용 과정을 수행할지 여부가 결정될 수 있다.
- [0070] 픽처 또는 슬라이스는 복수의 오프셋 영역들로 분할될 수 있고, 각 오프셋 영역별로 오프셋 타입이 결정될 수 있으며, 상기 오프셋 타입은 미리 정해진 개수(예를 들어, 4개)의 에지 오프셋 타입과 2개의 밴드 오프셋 타입을 포함할 수 있다.
- [0071] 예를 들어, 오프셋 타입이 에지 오프셋 타입일 경우, 각 화소가 속하는 에지 타입을 결정하여 이에 대응하는 오프셋을 적용하며, 상기 에지 타입은 현재 화소와 인접하는 2개의 화소값의 분포를 기준으로 결정될 수 있다.
- [0072] 적응적 루프 필터링(ALF) 과정은 디블록킹 필터링 과정 또는 적응적 오프셋 적용 과정을 거친 복원된 영상과 원본 영상을 비교한 값을 기초로 필터링을 수행할 수 있다.
- [0073] 픽처 저장부(180)는 후처리된 영상 데이터를 후처리부(170)로부터 입력받아 픽처(picture) 단위로 영상을 복원하여 저장하며, 픽처는 프레임 단위의 영상이거나 필드 단위의 영상일 수 있다.
- [0074] 인터 예측부(160)는 픽처 저장부(180)에 저장된 적어도 하나 이상의 참조 픽처를 이용하여 움직임 추정을 수행하고, 참조 픽처를 나타내는 참조 픽처 인덱스 및 움직임 벡터를 결정할 수 있다.
- [0075] 이 경우, 결정된 참조 픽처 인덱스 및 움직임 벡터에 따라, 픽처 저장부(180)에 저장된 다수의 참조 픽처들 중 움직임 추정에 이용된 참조 픽처로부터, 부호화하고자 하는 예측 유닛에 대응하는 예측 블록이 추출될 수 있다.
- [0076] 인트라 예측부(150)는 현재 예측 유닛이 포함되는 픽처 내부의 재구성된 화소값을 이용하여 인트라 예측 부호화를 수행할 수 있다.
- [0077] 인트라 예측부(150)는 예측 부호화할 현재 예측 유닛을 입력받아 현재 블록의 크기에 따라 미리 설정된 개수의 인트라 예측 모드 중에 하나를 선택하여 인트라 예측을 수행할 수 있다.
- [0078] 인트라 예측부(150)는 인트라 예측 블록을 생성하기 위해 참조 화소를 적응적으로 필터링하며, 참조 화소가 이용 가능하지 않은 경우 이용 가능한 참조 화소들을 이용하여 참조 화소들을 생성할 수 있다.
- [0079] 엔트로피 부호화부(140)는 양자화부(130)에 의해 양자화된 양자화 계수, 인트라 예측부(150)로부터 수신된 인트라 예측 정보, 인터 예측부(160)로부터 수신된 움직임 정보 등을 엔트로피 부호화할 수 있다.
- [0080] 도 6은 부호화 장치(10)에서 인터 예측을 수행하는 구성에 대한 일실시예를 블록도로 도시한 것으로, 도시된 인터 예측 부호화기는 움직임 정보 결정부(161), 움직임 정보 부호화 모드 결정부(162), 움직임 정보 부호화부(163), 예측 블록 생성부(164), 잔차 블록 생성부(165), 잔차 블록 부호화부(166) 및 멀티플렉서(167)를 포함하여 구성될 수 있다.
- [0081] 도 6을 참조하면, 움직임 정보 결정부(161)는 현재 블록의 움직임 정보를 결정하며, 움직임 정보는 참조 픽처 인덱스와 움직임 벡터를 포함하고, 참조 픽처 인덱스는 이전에 부호화되어 복원된 픽처 중 어느 하나를 나타낼 수 있다.
- [0082] 현재 블록이 단방향 인터 예측 부호화되는 경우에는 리스트 0(L0)에 속하는 참조 픽처들 중의 어느 하나를 나타내며, 현재 블록이 양방향 예측 부호화되는 경우에는 리스트 0(L0)의 참조 픽처들 중 하나를 나타내는 참조 픽처 인덱스와 리스트 1(L1)의 참조 픽처들 중의 하나를 나타내는 참조 픽처 인덱스를 포함할 수 있다.
- [0083] 또한, 현재 블록이 양방향 예측 부호화되는 경우에는 리스트 0과 리스트 1을 결합하여 생성된 복합 리스트(LC)의 참조 픽처들 중의 1개 또는 2개의 픽처를 나타내는 인덱스를 포함할 수 있다.
- [0084] 움직임 벡터는 각각의 참조픽처 인덱스가 나타내는 픽처 내의 예측 블록의 위치를 나타내며, 상기 움직임 벡터는 화소 단위(정수 단위) 또는 서브 화소 단위일 수 있다.
- [0085] 예를 들어, 상기 움직임 벡터는 1/2, 1/4, 1/8 또는 1/16 화소의 해상도를 가질 수 있으며, 움직임 벡터가 정수

단위가 아닐 경우 예측 블록은 정수 단위의 화소들로부터 생성될 수 있다.

- [0086] 움직임 정보 부호화 모드 결정부(162)는 현재 블록의 움직임 정보에 대한 부호화 모드를 스킵 모드, 머지 모드 및 AMVP 모드 중 어느 하나로 결정할 수 있다.
- [0087] 스킵 모드는 현재 블록의 움직임 정보와 동일한 움직임 정보를 갖는 스킵 후보자가 존재하고, 잔차 신호가 0인 경우에 적용되며, 상기 스킵 모드는 예측 유닛(PU)인 현재 블록이 코딩 유닛(CU)과 크기가 같을 때 적용될 수 있다.
- [0088] 머지 모드는 현재 블록의 움직임 정보와 동일한 움직임 정보를 갖는 머지 후보자가 존재할 때 적용되며, 상기 머지 모드는 현재 블록이 코딩 유닛(CU)과 크기가 다르거나, 크기가 같을 경우에는 잔차 신호가 존재하는 경우에 적용된다. 한편, 머지 후보자와 스킵 후보자는 동일할 수 있다.
- [0089] AMVP 모드는 스킵 모드 및 머지 모드가 적용되지 않을 때 적용되며, 현재 블록의 움직임 벡터와 가장 유사한 움직임 벡터를 갖는 AMVP 후보자를 AMVP 예측자로 선택할 수 있다.
- [0090] 움직임 정보 부호화부(163)는 움직임 정보 부호화 모드 결정부(162)에 의해 결정된 방식에 따라 움직임 정보를 부호화할 수 있다.
- [0091] 예를 들어, 움직임 정보 부호화부(163)는 움직임 정보 부호화 모드가 스킵 모드 또는 머지 모드일 경우에는 머지 움직임 벡터 부호화 과정을 수행하며, AMVP 모드일 경우에는 AMVP 부호화 과정을 수행할 수 있다.
- [0092] 예측 블록 생성부(164)는 현재 블록의 움직임 정보를 이용하여 예측 블록을 생성하며, 움직임 벡터가 정수 단위일 경우 참조 픽처 인덱스가 나타내는 픽처 내의 움직임 벡터가 나타내는 위치에 대응하는 블록을 복사하여 현재 블록의 예측 블록을 생성한다.
- [0093] 한편, 움직임 벡터가 정수 단위가 아닌 경우, 예측 블록 생성부(164)는 참조 픽처 인덱스가 나타내는 픽처 내의 정수 단위 화소들로부터 예측 블록의 화소들을 생성할 수 있다.
- [0094] 이 경우, 휘도 화소에 대해 8탭의 보간 필터를 사용하여 예측 화소가 생성되며, 색차 화소에 대해서는 4탭 보간 필터를 사용하여 예측 화소가 생성될 수 있다.
- [0095] 잔차 블록 생성부(165)는 현재 블록과 현재 블록의 예측 블록을 이용하여 잔차 블록을 생성하며, 현재 블록의 크기가 $2N \times 2N$ 인 경우 현재 블록과 현재 블록에 대응하는 $2N \times 2N$ 크기의 예측 블록을 이용하여 잔차 블록을 생성할 수 있다.
- [0096] 한편, 예측에 이용되는 현재 블록의 크기가 $2N \times N$ 또는 $N \times 2N$ 인 경우, $2N \times 2N$ 을 구성하는 2개의 $2N \times N$ 블록 각각에 대한 예측 블록을 구한 후, 상기 2개의 $2N \times N$ 예측 블록을 이용하여 $2N \times 2N$ 크기의 최종 예측 블록이 생성될 수 있다.
- [0097] 또한, 상기 $2N \times 2N$ 크기의 예측 블록을 이용하여 $2N \times 2N$ 크기의 잔차 블록이 생성될 수도 있으며, $2N \times N$ 크기를 가지는 2개의 예측 블록들의 경계 부분의 불연속성을 해소하기 위해 경계 부분의 픽셀들에 대해 오버랩 스무딩이 적용될 수 있다.
- [0098] 잔차 블록 부호화부(166)는 상기 잔차 블록을 하나 이상의 변환 유닛(TU)들로 분할하여, 각각의 변환 유닛(TU)이 변환 부호화, 양자화 및 엔트로피 부호화될 수 있다.
- [0099] 잔차 블록 부호화부(166)는 인터 예측 방법에 의해 생성된 잔차 블록을 정수기반 변환 매트릭스를 이용하여 변환할 수 있으며, 상기 변환 매트릭스는 정수기반 DCT 매트릭스일 수 있다.
- [0100] 한편, 잔차 블록 부호화부(166)는 변환 매트릭스에 의해 변환된 잔차 블록의 계수들을 양자화하기 위해 양자화 매트릭스를 이용하며, 상기 양자화 매트릭스는 양자화 파라미터에 의해 결정될 수 있다.
- [0101] 상기 양자화 파라미터는 미리 정해진 크기 이상의 코딩 유닛(CU) 별로 결정되며, 현재 코딩 유닛(CU)이 상기 미리 정해진 크기보다 작은 경우 상기 미리 정해진 크기 내의 코딩 유닛(CU)들 중 부호화 순서상 첫번째 코딩 유닛(CU)의 양자화 파라미터만을 부호화하고 나머지 코딩 유닛(CU)의 양자화 파라미터는 상기 파라미터와 동일하므로 부호화하지 않을 수 있다.
- [0102] 또한, 상기 양자화 파라미터 및 예측 모드에 따라 결정되는 양자화 매트릭스를 이용하여 상기 변환 블록의 계수들이 양자화될 수 있다.

- [0103] 상기 미리 정해진 크기 이상의 코딩 유닛(CU) 별로 결정되는 양자화 파라미터는 현재 코딩 유닛(CU)에 인접한 코딩 유닛(CU)의 양자화 파라미터를 이용하여 예측 부호화될 수 있다.
- [0104] 현재 코딩 유닛(CU)의 좌측 코딩 유닛(CU), 상측 코딩 유닛(CU) 순서로 검색하여 유효한 1개 또는 2개의 유효한 양자화 파라미터를 이용하여 현재 코딩 유닛(CU)의 양자화 파라미터 예측자를 생성할 수 있다.
- [0105] 예를 들어, 상기 순서로 검색된 유효한 첫번째 양자화 파라미터를 양자화 파라미터 예측자로 결정할 수 있으며, 또한 좌측 코딩 유닛(CU), 부호화 순서상 바로 이전의 코딩 유닛(CU) 순으로 검색하여 유효한 첫번째 양자화 파라미터를 양자화 파라미터 예측자로 결정할 수 있다.
- [0106] 양자화된 변환 블록의 계수들은 스캐닝되어 1차원의 양자화 계수들로 변환되며, 스캐닝 방식은 엔트로피 부호화 모드에 따라 다르게 설정될 수 있다.
- [0107] 예를 들어, CABAC으로 부호화될 경우 인터 예측 부호화된 양자화 계수들은 미리 정해진 하나의 방식(지그재그, 또는 대각선 방향으로의 래스터 스캔)으로 스캐닝될 수 있으며, CAVLC으로 부호화될 경우 상기 방식과 다른 방식으로 스캐닝될 수 있다.
- [0108] 예를 들어, 스캐닝 방식이 인터의 경우에는 지그재그, 인트라의 경우에는 인트라 예측 모드에 따라 결정될 수 있으며, 계수 스캐닝 방식은 변환 단위의 크기에 따라 상이하게 결정될 수도 있다.
- [0109] 한편, 상기 스캔 패턴은 방향성 인트라 예측 모드에 따라 달라질 수 있으며, 양자화 계수들의 스캔 순서는 역방향으로 스캔될 수 있다.
- [0110] 멀티플렉서(167)는 상기 움직임 정보 부호화부(163)에 의해 부호화된 움직임 정보들과 상기 잔차 블록 부호화부(166)에 의해 부호화된 잔차 신호들을 다중한다.
- [0111] 상기 움직임 정보는 부호화 모드에 따라 달라질 수 있으며, 예를 들어 스킵 또는 머지일 경우에는 예측자를 나타내는 인덱스만을 포함하고, AMVP일 경우 현재 블록의 참조 픽처 인덱스, 차분 움직임 벡터 및 AMVP 인덱스를 포함할 수 있다.
- [0112] 이하, 도 1에 도시된 인트라 예측부(150)의 동작에 대한 일실시예를 상세히 설명하기로 한다.
- [0113] 먼저, 인트라 예측부(150)는 픽처 분할부(110)로부터 예측 모드 정보 및 예측 유닛(PU)의 크기를 수신하며, 예측 유닛(PU)의 인트라 예측 모드를 결정하기 위해 참조 화소를 픽처 저장부(180)로부터 읽어들이 수 있다.
- [0114] 인트라 예측부(150)는 이용 가능하지 않은 참조 화소가 존재하는지 여부를 검토하여 참조 화소 생성 여부를 판단하며, 상기 참조 화소들은 현재 블록의 인트라 예측 모드를 결정하는데 사용될 수 있다.
- [0115] 현재 블록이 현재 픽처의 상측 경계에 위치하는 경우에는 현재 블록의 상측에 인접한 화소들이 정의되지 않고, 현재 블록이 현재 픽처의 좌측 경계에 위치하는 경우에는 현재 블록의 좌측에 인접한 화소들이 정의되지 않으며, 상기 화소들은 이용 가능한 화소들이 아닌 것으로 판단될 수 있다.
- [0116] 또한, 현재 블록이 슬라이스 경계에 위치하여 슬라이스의 상측 또는 좌측에 인접하는 화소들이 먼저 부호화되어 복원되는 화소들이 아닌 경우에도 이용 가능한 화소들이 아닌 것으로 판단될 수 있다.
- [0117] 상기와 같이 현재 블록의 좌측 또는 상측에 인접한 화소들이 존재하지 않거나, 미리 부호화되어 복원된 화소들이 존재하지 않는 경우, 이용 가능한 화소들만을 이용하여 현재 블록의 인트라 예측 모드가 결정될 수도 있다.
- [0118] 한편, 현재 블록의 이용 가능한 참조 화소들을 이용하여 이용 가능하지 않은 위치의 참조 화소가 생성될 수도 있으며, 예를 들어 상측 블록의 화소들이 이용 가능하지 않은 경우 좌측 화소들의 일부 또는 전부를 이용하여 상측 화소들을 생성할 수 있고, 그 역으로도 가능하다.
- [0119] 즉, 이용 가능하지 않은 위치의 참조 화소로부터 미리 정해진 방향으로 가장 가까운 위치의 이용 가능한 참조 화소를 복사하여 참조 화소가 생성되거나, 미리 정해진 방향에 이용 가능한 참조 화소가 존재하지 않는 경우 반대 방향의 가장 가까운 위치의 이용 가능한 참조 화소를 복사하여 참조 화소가 생성될 수 있다.
- [0120] 한편, 현재 블록의 상측 또는 좌측 화소들이 존재하는 경우에도 상기 화소들이 속하는 블록의 부호화 모드에 따라 이용 가능하지 않은 참조 화소로 결정될 수 있다.
- [0121] 예를 들어, 현재 블록의 상측에 인접한 참조 화소가 속하는 블록이 인터 부호화되어 복원된 블록일 경우, 상기 화소들을 이용 가능하지 않은 화소들로 판단할 수 있다.

- [0122] 이 경우, 현재 블록에 인접한 블록이 인트라 부호화되어 복원된 블록에 속하는 화소들을 이용하여 이용 가능한 참조 화소들이 생성될 수 있으며, 부호화 장치(10)가 부호화 모드에 따라 이용 가능한 참조 화소를 판단한다는 정보를 복호화 장치(20)로 전송한다.
- [0123] 인트라 예측부(150)는 상기 참조 화소들을 이용하여 현재 블록의 인트라 예측 모드를 결정하며, 현재 블록에 이용 가능한 인트라 예측 모드의 수는 블록의 크기에 따라 달라질 수 있다.
- [0124] 예를 들어, 현재 블록의 크기가 8x8, 16x16, 32x32인 경우에는 34개의 인트라 예측 모드가 존재할 수 있고, 현재 블록의 크기가 4x4인 경우에는 17개의 인트라 예측 모드가 존재할 수 있다.
- [0125] 상기 34개 또는 17개의 인트라 예측 모드는 적어도 하나 이상의 비방향성 모드(non-directional mode)와 복수개의 방향성 모드들(directional modes)로 구성될 수 있다.
- [0126] 하나 이상의 비방향성 모드는 DC 모드 및/또는 플래너(planar) 모드일 수 있다. DC 모드 및 플래너모드가 비방향성 모드로 포함되는 경우에는, 현재 블록의 크기에 관계없이 35개의 인트라 예측 모드가 존재할 수도 있다.
- [0127] 이 경우, 2개의 비방향성 모드(DC 모드 및 플래너 모드)와 33개의 방향성 모드가 포함될 수 있다.
- [0128] 플래너 모드의 경우, 현재 블록의 우하측(bottom-right)에 위치하는 적어도 하나의 화소값(또는 상기 화소값의 예측값, 이하 제1 참조값이라 함)과 참조 화소들을 이용하여 현재 블록의 예측 블록이 생성된다.
- [0129] 본 발명의 일실시예에 따른 영상 복호화 장치의 구성은 도 1 내지 도 6을 참조하여 설명한 영상 부호화 장치(10)의 구성으로부터 도출될 수 있으며, 예를 들어 도 1 내지 도 6을 참조하여 설명한 바와 같은 영상 부호화 방법의 과정들을 역으로 수행함으로써 영상을 복호화할 수 있다.
- [0130] 도 7은 본 발명의 일실시예에 따른 동영상 복호화 장치의 구성을 블록도로 도시한 것으로, 복호화 장치(20)는 엔트로피 복호화부(210), 역양자화/역변환부(220), 가산기(270), 디블록킹 필터(250), 픽처 저장부(260), 인트라 예측부(230), 움직임 보상 예측부(240) 및 인트라/인터전환 스위치(280)를 구비한다.
- [0131] 엔트로피 복호화부(210)는, 영상 부호화 장치(10)에서 부호화된 비트 스트림을 입력받아 복호화하여 인트라 예측 모드 인덱스, 움직임 정보, 양자화 계수 시퀀스 등으로 분리하며, 복호화된 움직임 정보를 움직임 보상 예측부(240)로 전달한다.
- [0132] 엔트로피 복호화부(210)는 인트라 예측 모드 인덱스를 인트라 예측부(230)와 역양자화/역변환부(220)로 전달하여, 역양자화 계수 시퀀스를 역양자화/역변환부(220)로 전달한다.
- [0133] 역양자화/역변환부(220)는 상기 양자화 계수 시퀀스를 2차원 배열의 역양자화 계수로 변환하며, 상기 변환을 위해 복수의 스케닝 패턴들 중 하나를 선택할 수 있으며 예를 들어 현재 블록의 예측 모드(즉, 인트라 예측 또는 인터 예측)와 인트라 예측 모드에 기초하여 스케닝 패턴을 선택할 수 있다.
- [0134] 역양자화/역변환부(220)는 2차원 배열의 역양자화 계수에 대해 복수의 양자화 매트릭스들 중에서 선택된 양자화 매트릭스를 적용하여 양자화 계수를 복원한다.
- [0135] 한편, 복원하고자 하는 현재 블록의 크기에 따라 서로 다른 양자화 매트릭스가 적용되며, 동일 크기의 블록에 대해서도 상기 현재 블록의 예측 모드 및 인트라 예측 모드 중 적어도 하나에 기초하여 양자화 매트릭스가 선택될 수 있다.
- [0136] 역양자화/역변환부(220)는 상기 복원된 양자화 계수를 역변환하여 잔차 블록을 복원하며, 상기 역변환 과정은 변환 유닛(TU)을 기본 단위로 하여 수행될 수 있다.
- [0137] 가산기(270)는 역양자화/역변환부(220)에 의해 복원된 잔차 블록과 인트라 예측부(230) 또는 움직임 보상 예측부(240)에 의해 생성되는 예측 블록을 합하여 영상 블록을 복원한다.
- [0138] 디블록킹 필터(250)는 가산기(270)에 의해 생성된 복원 영상에 디블록킹 필터 처리를 수행하여, 양자화 과정에 따른 영상 손실에 기인하는 디블록킹 아티팩트를 감소시킬 수 있다.
- [0139] 픽처 저장부(260)는 디블록킹 필터(250)에 의해 디블록킹 필터 처리가 수행된 로컬 복호 영상을 저장하기 위한 프레임 메모리이다.
- [0140] 인트라 예측부(230)는 엔트로피 복호화부(210)로부터 수신된 인트라 예측 모드 인덱스에 기초하여 현재 블록의 인트라 예측 모드를 복원하고, 복원된 인트라 예측 모드에 따라 예측 블록을 생성한다.

- [0141] 움직임 보상 예측부(240)는 움직임 벡터 정보에 기초하여 픽처 저장부(260)에 저장된 픽처로부터 현재 블록에 대한 예측 블록을 생성하며, 소수 정밀도의 움직임 보상이 적용될 경우 선택된 보간 필터를 적용하여 예측 블록을 생성할 수 있다.
- [0142] 인트라/인터 전환 스위치(280)는 부호화 모드에 기초하여 인트라 예측부(230)와 움직임 보상 예측부(240)의 어느 하나에서 생성된 예측 블록을 가산기(270)에 제공할 수 있다.
- [0143] 도 8는 영상 복호화 장치(20)에서 인터 예측을 수행하는 구성에 대한 일실시예를 블록도로 도시한 것으로, 인터 예측 복호화기는 디멀티플렉서(241), 움직임 정보 부호화 모드 판단부(242), 머지 모드 움직임 정보 복호화부(243), AMVP 모드 움직임 정보 복호화부(244), 예측블록 생성부(245), 잔차 블록 복호화부(246) 및 복원블록 생성부(247)를 포함한다.
- [0144] 도 8을 참조하면, 디-멀티플렉서(241)는 수신된 비트스트림으로부터 현재 부호화된 움직임 정보와 부호화된 잔차 신호들을 역다중화하여, 역다중화된 움직임 정보를 움직임 정보 부호화 모드 판단부(242)로 전송하고, 역다중화된 잔차신호를 잔차블록 복호화부(246)로 전송할 수 있다.
- [0145] 움직임 정보 부호화 모드 판단부(242)는 현재 블록의 움직임 정보 부호화 모드를 판단하며, 수신된 비트스트림의 skip_flag가 1의 값을 갖는 경우 현재 블록의 움직임 정보 부호화 모드가 스킵 부호화 모드로 부호화된 것으로 판단할 수 있다.
- [0146] 움직임 정보 부호화 모드 판단부(242)는 수신된 비트스트림의 skip_flag가 0의 값을 갖고, 디-멀티플렉서(241)로부터 수신된 움직임 정보가 머지 인덱스만을 갖는 경우, 현재 블록의 움직임 정보 부호화 모드가 머지 모드로 부호화된 것으로 판단할 수 있다.
- [0147] 또한, 움직임 정보 부호화 모드 판단부(242)는 수신된 비트스트림의 skip_flag가 0의 값을 갖고, 디멀티플렉서(241)로부터 수신된 움직임 정보가 참조 픽처 인덱스와 차분 움직임 벡터와 AMVP인덱스를 갖는 경우, 현재 블록의 움직임 정보 부호화 모드가 AMVP 모드로 부호화된 것으로 판단할 수 있다.
- [0148] 머지 모드 움직임 정보 복호화부(243)는 움직임 정보 부호화 모드 판단부(242)가 현재 블록의 움직임 정보 부호화 모드를 스킵 또는 머지 모드로 판단한 경우에 활성화되며, AMVP 모드 움직임 정보 복호화부(244)는 움직임 정보 부호화 모드 판단부(242)가 현재 블록의 움직임 정보 부호화 모드를 AMVP 모드로 판단한 경우에 활성화될 수 있다.
- [0149] 예측블록 생성부(245)는 머지 모드 움직임 정보 복호화부(243) 또는 AMVP 모드 움직임 정보 복호화부(244)에 의해 복원된 움직임 정보를 이용하여 현재 블록의 예측 블록을 생성한다.
- [0150] 움직임 벡터가 정수 단위일 경우, 참조 픽처 인덱스가 나타내는 픽처 내의 움직임 벡터가 나타내는 위치에 대응하는 블록을 복사하여 현재 블록의 예측 블록이 생성될 수 있다.
- [0151] 한편, 움직임 벡터가 정수 단위가 아닐 경우, 참조 픽처 인덱스가 나타내는 픽처 내의 정수 단위 화소들로부터 예측 블록의 화소들이 생성되며, 이 경우 휘도 화소의 경우에는 8탭의 보간 필터를 사용하고 색차 화소의 경우 4탭 보간 필터를 사용하여 예측 화소가 생성될 수 있다.
- [0152] 잔차 블록 복호화부(246)는 잔차 신호를 엔트로피 복호화하고, 엔트로피 복호화된 계수들을 역스캐닝하여 2차원의 양자화된 계수 블록을 생성하며, 역스캐닝 방식은 엔트로피 복호화 방식에 따라 달라질 수 있다.
- [0153] 예를 들어, CABAC 기반으로 복호화된 경우 대각선 방향의 래스터 역스캔 방식으로, CAVLC 기반으로 복호화된 경우에는 지그재그 역스캔 방식으로 상기 역스캐닝 방식이 적용될 수 있다. 또한, 예측 블록의 크기에 따라 상기 역스캐닝 방식이 상이하게 결정될 수도 있다.
- [0154] 잔차블록 복호화부(246)는 상기과 같이 생성된 계수블록을 역양자화 매트릭스를 이용하여 역양자화하며, 상기 양자화 매트릭스를 유도하기 위해 양자화 파라미터를 복원할 수 있다. 여기서, 양자화 스텝 사이즈는 미리 정해진 크기 이상의 코딩 윗별로 복원될 수 있다.
- [0155] 잔차블록 복호화부(260)는 상기 역양자화된 계수 블록을 역변환하여 잔차블록을 복원한다.
- [0156] 복원블록 생성부(270)는 상기 예측블록 생성부(250)에 의해 생성된 예측 블록과 상기 잔차블록 복호화부(260)에 의하여 생성된 잔차 블록을 더하여 복원 블록을 생성한다.
- [0157] 이하, 현재 블록을 인트라 예측을 통해 복원하는 과정에 대한 일실시예를 도 7을 다시 참조하여 설명한다.

- [0158] 먼저, 수신된 비트스트림으로부터 현재 블록의 인트라 예측 모드가 복호화되며, 그를 위해 엔트로피 복호화부(210)는 복수의 인트라 예측 모드 테이블들 중 하나를 참조하여 현재 블록의 제1 인트라 예측 모드 인덱스를 복원할 수 있다.
- [0159] 상기 복수의 인트라 예측 모드 테이블들 부호화 장치(10)와 복호화 장치(20)가 공유하는 테이블로서, 현재 블록에 인접한 복수 블록들에 대한 인트라 예측 모드의 분포에 따라 선택된 어느 하나의 테이블이 적용될 수 있다.
- [0160] 예를 들어, 현재 블록의 좌측 블록의 인트라 예측 모드와 현재 블록의 상측 블록의 인트라 예측 모드가 동일하면 제1 인트라 예측 모드 테이블을 적용하여 현재 블록의 제 1 인트라 예측 모드 인덱스를 복원하고, 동일하지 않으면 제2 인트라 예측 모드 테이블을 적용하여 현재 블록의 제 1 인트라 예측 모드 인덱스를 복원할 수 있다.
- [0161] 또 다른 예로써, 현재 블록의 상측 블록과 좌측 블록의 인트라 예측 모드가 모두 방향성 예측 모드(directional intra prediction mode)일 경우, 상기 상측 블록의 인트라 예측 모드의 방향과 상기 좌측 블록의 인트라 예측 모드의 방향이 소정 각도 이내이면 제 1 인트라 예측 모드 테이블을 적용하여 현재 블록의 제 1 인트라 예측 모드 인덱스를 복원하고, 소정 각도를 벗어나면 제2 인트라 예측 모드 테이블을 적용하여 현재 블록의 제 1 인트라 예측 모드 인덱스를 복원할 수도 있다.
- [0162] 엔트로피 복호화부(210)는 복원된 현재 블록의 제1 인트라 예측 모드 인덱스를 인트라 예측부(230)로 전송한다.
- [0163] 제1 인트라 예측 모드를 인덱스를 수신한 인트라 예측부(230)는, 상기 인덱스가 최소값을 가질 경우(즉, 0일 경우), 현재 블록의 최대 가능 모드를 현재 블록의 인트라 예측 모드로 결정할 수 있다.
- [0164] 한편, 인트라 예측부(230)는, 상기 인덱스가 0 이외의 값을 가질 경우, 현재 블록의 최대 가능 모드가 나타내는 인덱스와 상기 제1 인트라 예측 모드 인덱스를 비교하고, 비교 결과 상기 제1 인트라 예측 모드 인덱스가 상기 현재 블록의 최대 가능 모드가 나타내는 인덱스보다 작지 않으면 상기 제1 인트라 예측 모드 인덱스에 1을 더한 제2 인트라 예측 모드 인덱스에 대응하는 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 결정하고, 그렇지 않으면 상기 제1 인트라 예측 모드 인덱스에 대응하는 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 결정할 수 있다.
- [0165] 현재 블록에 허용 가능한 인트라 예측 모드는 적어도 하나 이상의 비방향성 모드(non-directional mode)와 복수개의 방향성 모드들(directional modes)로 구성될 수 있다.
- [0166] 하나 이상의 비방향성 모드는 DC 모드 및/또는 플래너(planar) 모드일수 있다. 또한, DC 모드와 플래너 모드 중 어느 하나가 적응적으로 상기 허용 가능한 인트라 예측 모드 셋에 포함될 수 있다.
- [0167] 이를 위해, 픽처 헤더 또는 슬라이스 헤더에 상기 허용 가능한 인트라 예측 모드 셋에 포함되는 비방향성 모드를 특정하는 정보가 포함될 수 있다.
- [0168] 다음으로, 인트라 예측부(230)는 인트라 예측 블록을 생성하기 위해, 참조 화소들을 픽처 저장부(260)로부터 읽어들이고, 이용 가능하지 않은 참조 화소가 존재하는지 여부를 판단한다.
- [0169] 상기 판단은 현재 블록의 복호된 인트라 예측 모드를 적용하여 인트라 예측 블록을 생성하는데 이용되는 참조 화소들의 존재 여부에 따라 행해질 수도 있다.
- [0170] 다음으로, 인트라 예측부(230)는 참조 화소를 생성할 필요가 있을 경우에는 미리 복원된 이용 가능한 참조 화소들을 이용하여 이용 가능하지 않은 위치의 참조 화소들을 생성할 수 있다.
- [0171] 이용 가능하지 않은 참조 화소에 대한 정의 및 참조 화소의 생성 방법은 도 1에 따른 인트라 예측부(150)에서의 동작과 동일할 수 있으나, 현재 블록의 복호화된 인트라 예측 모드에 따라 인트라 예측 블록을 생성하는데 이용되는 참조 화소들이 선택적으로 복원될 수도 있다.
- [0172] 또한, 인트라 예측부(230)는 예측 블록을 생성하기 위해 참조 화소들에 필터를 적용할지 여부를 판단하며, 즉 현재 블록의 인트라 예측 블록을 생성하기 위하여 참조 화소들에 대해 필터링을 적용할지 여부를 상기 복호된 인트라 예측 모드 및 현재 예측 블록의 크기에 기초하여 결정할 수 있다.
- [0173] 블록킹 아티팩트의 문제는 블록의 크기가 커질수록 커지므로, 블록의 크기가 커질수록 참조 화소를 필터링하는 예측모드의 수를 증가시킬 수 있으나, 블록이 소정 크기보다 커지는 경우 평탄한 영역으로 볼 수 있으므로 복잡도 감소를 위해 참조 화소를 필터링하지 않을 수도 있다.
- [0174] 상기 참조 화소에 필터 적용이 필요하다고 판단된 경우, 인트라 예측부(230)는 필터를 이용하여 상기 참조 화소

들을 필터링한다.

- [0175] 상기한 참조 화소들 간의 단차의 차이 정도에 따라 적어도 2개 이상의 필터를 적응적으로 적용할 수도 있다. 상기 필터의 필터계수는 대칭적인 것이 바람직하다.
- [0176] 또한, 상기한 2개 이상의 필터가 현재 블록의 크기에 따라 적응적으로 적용될 수도 있으며, 필터를 적용할 경우 크기가 작은 블록에 대해서는 대역폭이 좁은 필터가, 크기가 큰 블록들에 대해서는 대역폭이 넓은 필터가 적용될 수도 있다.
- [0177] DC 모드인 경우에는 참조 화소들의 평균값으로 예측 블록이 생성되므로 필터를 적용할 필요가 없으며, 상이 수직 방향으로 연관성(correlation)이 있는 수직 모드에서는 참조 화소에 필터를 적용할 필요가 없고, 영상이 수평 방향으로 연관성이 있는 수평 모드에서도 참조 화소에 필터를 적용할 필요가 없을 수 있다.
- [0178] 이와 같이, 필터링의 적용 여부는 현재 블록의 인트라 예측 모드와도 연관성이 있으므로, 현재 블록의 인트라 예측 모드 및 예측 블록의 크기에 기초하여 참조 화소를 적응적으로 필터링할 수 있다.
- [0179] 다음으로, 인트라 예측부(230)는 상기 복원된 인트라 예측 모드에 따라 참조 화소 또는 필터링된 참조 화소들을 이용하여 예측 블록을 생성하며, 상기 예측 블록의 생성은 부호화 장치(10)에서의 동작과 동일할 수 있으므로, 그에 대한 상세한 설명은 생략하기로 한다.
- [0180] 인트라 예측부(230)는 상기 생성된 예측 블록을 필터링할지 여부를 판단하며, 상기 필터링 여부는 슬라이스 헤더 또는 부호화 유닛 헤더에 포함된 정보를 이용하거나 또는 현재 블록의 인트라 예측 모드에 따라 결정될 수 있다.
- [0181] 상기 생성된 예측 블록을 필터링할 것으로 판단할 경우, 인트라 예측부(230)는 현재 블록에 인접한 이용 가능한 참조 화소들을 이용하여 생성된 예측 블록의 특정 위치의 화소를 필터링하여 새로운 화소를 생성할 수 있다.
- [0182] 예를 들어, DC 모드에서는 예측 화소들 중 참조 화소들과 접하는 예측 화소는 상기 예측 화소와 접하는 참조 화소를 이용하여 필터링될 수 있다.
- [0183] 따라서, 예측 화소의 위치에 따라 1개 또는 2개의 참조 화소를 이용하여 예측 화소가 필터링되며, DC 모드에서의 예측 화소의 필터링은 모든 크기의 예측 블록에 적용할 수 있다.
- [0184] 한편, 수직 모드에서는 예측 블록의 예측 화소들 중 좌측 참조 화소와 접하는 예측 화소들은 상기 예측블록을 생성하는데 이용되는 상측 화소 이외의 참조 화소들을 이용하여 변경될 수 있다.
- [0185] 마찬가지로, 수평 모드에서는 생성된 예측 화소들 중 상측 참조 화소와 접하는 예측 화소들은 상기 예측블록을 생성하는데 이용되는 좌측 화소 이외의 참조 화소들을 이용하여 변경될 수 있다.
- [0186] 이와 같은 방식으로 복원된 현재 블록의 예측 블록과 복호화한 현재 블록의 잔차 블록을 이용하여 현재 블록이 복원될 수 있다.
- [0187] 도 9는 영상을 블록 단위로 분할하여 처리하는 방법에 대한 제2 실시예를 설명하기 위한 도식인 것이다.
- [0188] 도 9를 참조하면, 최대 256x256 픽셀 크기를 가지는 코딩 트리 유닛(CTU)은 먼저 쿼드 트리(quad tree) 구조로 나뉘어, 정사각형의 형태를 가지는 4개의 코딩 유닛(CU)들로 분할될 수 있다.
- [0189] 여기서, 상기 쿼드 트리 구조로 분할된 코딩 유닛들 중 적어도 하나는 이진 트리(binary tree) 구조로 나뉘어, 직사각형의 형태를 가지는 2개의 코딩 유닛(CU)들로 재 분할될 수 있다.
- [0190] 한편, 상기 쿼드 트리 구조로 분할된 코딩 유닛들 중 적어도 하나는 쿼드 트리 구조로 나뉘어, 정사각형의 형태를 가지는 4개의 코딩 유닛(CU)들로 재 분할될 수도 있다.
- [0191] 그리고 상기 이진 트리 구조로 재 분할된 코딩 유닛들 중 적어도 하나는 이진 트리 구조로 다시 나뉘어, 정사각형 또는 직사각형의 형태를 가지는 2개의 코딩 유닛(CU)들로 분할될 수 있다.
- [0192] 한편, 상기 쿼드 트리 구조로 재 분할된 코딩 유닛들 중 적어도 하나는 쿼드 트리 구조 또는 이진 트리 구조로 다시 나뉘어, 정사각형 또는 직사각형의 형태를 가지는 코딩 유닛(CU)들로 분할될 수도 있다.
- [0193] 상기와 같이 이진 트리 구조로 분할되어 구성된 코딩 블록(Coding Block : CB)들은, 더 이상 분할되지 않고, 예측 및 변환에 이용될 수 있다. 즉, 도 9에 도시된 바와 같은 코딩 블록(CB)에 속하는 예측 유닛(PU)과 변환 유닛(TU)의 크기는, 해당 코딩 블록(CB)의 크기와 동일할 수 있다.

- [0194] 상기와 같이 쿼드 트리 구조로 분할된 코딩 유닛은 도 3 및 도 4를 참조하여 설명한 바와 같은 방법을 이용하여 하나 또는 2 이상의 예측 유닛(PU)들로 분할될 수 있다.
- [0195] 또한, 상기와 같이 쿼드 트리 구조로 분할된 코딩 유닛은 도 5를 참조하여 설명한 바와 같은 방법을 이용하여 하나 또는 2 이상의 변환 유닛(TU)들로 분할될 수 있으며, 상기 분할된 변환 유닛(TU)은 최대 64x64 픽셀 크기를 가질 수 있다.
- [0196] 도 10은 영상을 블록 단위로 분할하여 처리하기 위해 사용되는 신택스(syntax) 구조에 대한 일실시예를 나타내는 것이다.
- [0197] 도 10을 참조하면, 도 9를 참조하여 설명한 바와 같은 코딩 유닛(CU)의 분할 여부는 `split_cu_flag`를 이용하여 나타내어지며, 이진 트리를 이용하여 분할되는 코딩 유닛(CU)의 깊이는 `binary_depth`를 이용하여 나타내어질 수 있다.
- [0198] 또한, 코딩 유닛(CU)이 이진 트리 구조로 분할되는지 여부는 별도의 `binary_split_flag`로 나타내어 질 수도 있다.
- [0199] 도 9 및 도 10을 참조하여 설명한 바와 같은 방법에 의해 분할된 블록들(예를 들어, 코딩 유닛(CU), 예측 유닛(PU) 및 변환 유닛(TU))에 대해 도 1 내지 도 8을 참조하여 설명한 바와 같은 방법들이 적용됨으로써, 영상에 대한 부호화 및 복호화가 수행될 수 있다.
- [0200] 이하에서는, 도 11 내지 도 16을 참조하여, 코딩 유닛(CU)을 하나 또는 2 이상의 변환 유닛(TU)들로 분할하는 방법에 대한 또 다른 실시예들을 설명하기로 한다.
- [0201] 본 발명의 실시예에 따르면, 코딩 유닛(CU)은 이진 트리 구조로 나뉘어 잔차 블록에 대한 변환이 수행되는 기본 단위인 변환 유닛(TU)들로 분할될 수 있다.
- [0202] 도 11을 참조하면, 이진 트리 구조로 분할되어 $N \times 2N$ 또는 $2N \times N$ 의 크기를 가지는 직사각형의 코딩 블록들(CB_0 , CB_1) 중 적어도 하나는, 다시 이진 트리 구조로 나뉘어, $N \times N$ 의 크기를 가지는 정사각형의 변환 유닛들(TU_0 , TU_1)로 분할될 수 있다.
- [0203] 상기한 바와 같이, 블록 기반의 영상 부호화 방법은, 예측, 변환, 양자화 및 엔트로피 부호화 단계들을 수행할 수 있다.
- [0204] 상기 예측 단계에서는, 현재 부호화를 수행하는 블록과 기존의 부호화된 영상 또는 주변 영상을 참조하여 예측 신호를 생성하며, 이를 통해 현재 블록과의 차분 신호를 계산할 수 있다.
- [0205] 한편, 변환 단계에서는, 상기 차분 신호를 입력으로 하여 다양한 변환 함수를 이용하여 변환을 수행하며, 상기 변환된 신호는 DC 계수와 AC 계수들로 분류된 에너지 집중(Energy compaction)되어 부호화 효율이 향상될 수 있다.
- [0206] 또한, 양자화 단계에서는 변환 계수(Transform coefficient)들을 입력으로 양자화가 수행하며, 이후 양자화된 신호에 대해 엔트로피 부호화가 수행됨으로써 영상이 부호화될 수 있다.
- [0207] 한편, 영상 복호화 방법은 상기와 같은 부호화 과정의 역순으로 진행되며, 영상의 화질 왜곡 현상이 양자화 단계에서 발생할 수 있다.
- [0208] 부호화 효율을 향상시키면서 화질 왜곡 현상을 줄이기 위한 방법으로서, 변환 단계에서 입력으로 들어오는 차분 신호의 분포 및 영상의 특징에 따라 변환 유닛(TU)의 크기 또는 모양과 적용되는 변환 함수 종류를 다양하게 할 수 있다.
- [0209] 예를 들어, 예측 단계에서 블록 기반 움직임 추정 과정을 통해 현재 블록과 유사한 블록을 찾는 경우, SAD(Sum of Absolute Difference) 또는 MSE(Mean Square error) 등과 같은 코스트(cost) 측정 방법을 이용하여, 차분 신호의 분포는 영상의 특성에 따라 다양한 형태로 생길 수 있다.
- [0210] 그에 따라, 다양한 차분 신호의 분포에 기초해 선택적으로 변환 유닛(CU)의 크기 또는 모양을 결정하여 변환을 수행함으로써, 효과적인 부호화가 수행될 수 있다.
- [0211] 도 12를 참조하면, 임의의 코딩 블록(CB_x)에서 (a)에 도시된 바와 같이 차분 신호가 발생하는 경우, (b)에 도시된 바와 같이 해당 코딩 블록(CB_x)을 이진 트리 구조로 나누어 2개의 변환 유닛(TU)들로 분할함으로써 효과적인

인 변환이 수행되도록 할 수 있다.

- [0212] 예를 들어, DC 값은 일반적으로 입력 신호의 평균값을 나타낸다고 말할 수 있으므로, 도 12의 (a)에 도시된 바와 같은 차분 신호가 변환 과정의 입력으로 수신되는 경우, 코딩 블록(CBx)을 2개의 변환 유닛(TU)들로 분할함으로써 효과적으로 DC 값을 나타낼 수 있다.
- [0213] 도 13을 참조하면, $2N \times 2N$ 의 크기를 가지는 정사각형의 코딩 유닛(CU_0)이 이진 트리 구조로 나뉘어, $N \times 2N$ 또는 $2N \times N$ 의 크기를 가지는 직사각형의 변환 유닛들(TU_0 , TU_1)로 분할될 수 있다.
- [0214] 본 발명의 또 다른 실시예에 따르면, 상기한 바와 같이 코딩 유닛(CU)을 이진 트리 구조로 분할하는 단계를 2회 이상 반복해 수행하여, 복수의 변환 유닛(TU)들로 분할할 수 있다.
- [0215] 도 14를 참조하면, $N \times 2N$ 의 크기를 가지는 직사각형의 코딩 블록(CB_1)을 이진 트리 구조로 분할하고, 상기 분할된 $N \times N$ 의 크기를 가지는 블록을 다시 이진 트리 구조로 분할하여 $N/2 \times N$ 또는 $N \times N/2$ 의 크기를 가지는 직사각형의 블록을 구성한 후, 상기 $N/2 \times N$ 또는 $N \times N/2$ 의 크기를 가지는 블록을 다시 이진 트리 구조로 분할하여 $N/2 \times N/2$ 의 크기를 가지는 정사각형의 변환 유닛들(TU_1 , TU_2 , TU_4 , TU_5)로 분할할 수 있다.
- [0216] 도 15를 참조하면, $2N \times 2N$ 의 크기를 가지는 정사각형의 코딩 블록(CB_0)을 이진 트리 구조로 분할하고, 상기 분할된 $N \times 2N$ 의 크기를 가지는 블록을 다시 이진 트리 구조로 분할하여 $N \times N$ 의 크기를 가지는 정사각형의 블록을 구성한 후, 상기 $N \times N$ 의 크기를 가지는 블록을 다시 이진 트리 구조로 분할하여 $N/2 \times N$ 의 크기를 가지는 직사각형의 변환 유닛들(TU_1 , TU_2)로 분할할 수 있다.
- [0217] 도 16을 참조하면, $2N \times N$ 의 크기를 가지는 직사각형의 코딩 블록(CB_0)을 이진 트리 구조로 분할하고, 상기 분할된 $N \times N$ 의 크기를 가지는 블록을 다시 쿼드 트리 구조로 분할하여 $N/2 \times N/2$ 의 크기를 가지는 정사각형의 변환 유닛들(TU_1 , TU_2 , TU_3 , TU_4)로 분할할 수 있다.
- [0218] 도 11 내지 도 16을 참조하여 설명한 바와 같은 방법에 의해 분할된 블록들(예를 들어, 코딩 유닛(CU), 예측 유닛(PU) 및 변환 유닛(TU))에 대해 도 1 내지 도 8을 참조하여 설명한 바와 같은 방법들이 적용됨으로써, 영상에 대한 부호화 및 복호화가 수행될 수 있다.
- [0219] 이하에서는, 본 발명에 따른 부호화 장치(10)가 블록 분할 구조를 결정하는 방법에 대한 실시예들에 대해 설명한다.
- [0220] 영상 부호화 장치(10)에 구비된 픽처 분할부(110)는 미리 설정된 순서에 따라 RDO(Rate distortion Optimization)를 수행하여, 상기한 바와 같이 분할 가능한 코딩 유닛(CU), 예측 유닛(PU) 및 변환 유닛(TU)의 분할 구조를 결정할 수 있다.
- [0221] 예를 들어, 블록 분할 구조를 결정하기 위해, 픽처 분할부(110)는 RDO-Q(Rate distortion Optimization-Quantization)를 수행하면서 비트레이트(bitrate)와 디스토션(distortion) 측면에서 최적의 블록 분할 구조를 결정할 수 있다.
- [0222] 도 17을 참조하면, 코딩 유닛(CU)이 $2N \times 2N$ 픽셀 크기의 형태를 가지는 경우, (a)에 도시된 $2N \times 2N$ 픽셀 크기, (b)에 도시된 $N \times N$ 픽셀 크기, (c)에 도시된 $N \times 2N$ 픽셀 크기, (d)에 도시된 $2N \times N$ 픽셀 크기의 변환 유닛(PU) 분할 구조 순서로 RDO를 수행하여 변환 유닛(PU)의 최적 분할 구조를 결정할 수 있다.
- [0223] 도 18을 참조하면, 코딩 유닛(CU)이 $N \times 2N$ 또는 $2N \times N$ 픽셀 크기의 형태를 가지는 경우, (a)에 도시된 $N \times 2N$ (또는, $2N \times N$)의 픽셀 크기, (b)에 도시된 $N \times N$ 의 픽셀 크기, (c)에 도시된 $N/2 \times N$ (또는, $N \times N/2$)와 $N \times N$ 의 픽셀 크기, (d)에 도시된 $N/2 \times N/2$, $N/2 \times N$ 및 $N \times N$ 의 픽셀 크기, (e)에 도시된 $N/2 \times N$ 의 픽셀 크기의 변환 유닛(PU) 분할 구조 순서로 RDO를 수행하여 변환 유닛(PU)의 최적 분할 구조를 결정할 수 있다.
- [0224] 상기에서는 RDO(Rate distortion Optimization)를 수행하여 블록 분할 구조가 결정되는 것을 예로 들어 본 발명의 블록 분할 방법을 설명하였으나, 픽처 분할부(110)는 SAD(Sum of Absolute difference) 또는 MSE(Mean Square Error)를 이용하여 블록 분할 구조를 결정함으로써 복잡도를 저감시키면서도 적절한 효율을 유지할 수 있다.
- [0225] 상술한 본 발명에 따른 방법은 컴퓨터에서 실행되기 위한 프로그램으로 제작되어 컴퓨터가 읽을 수 있는 기록 매체에 저장될 수 있으며, 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피

디스크, 광 데이터 저장장치 등이 있으며, 또한 캐리어 웨이브(예를 들어 인터넷을 통한 전송)의 형태로 구현되는 것도 포함한다.

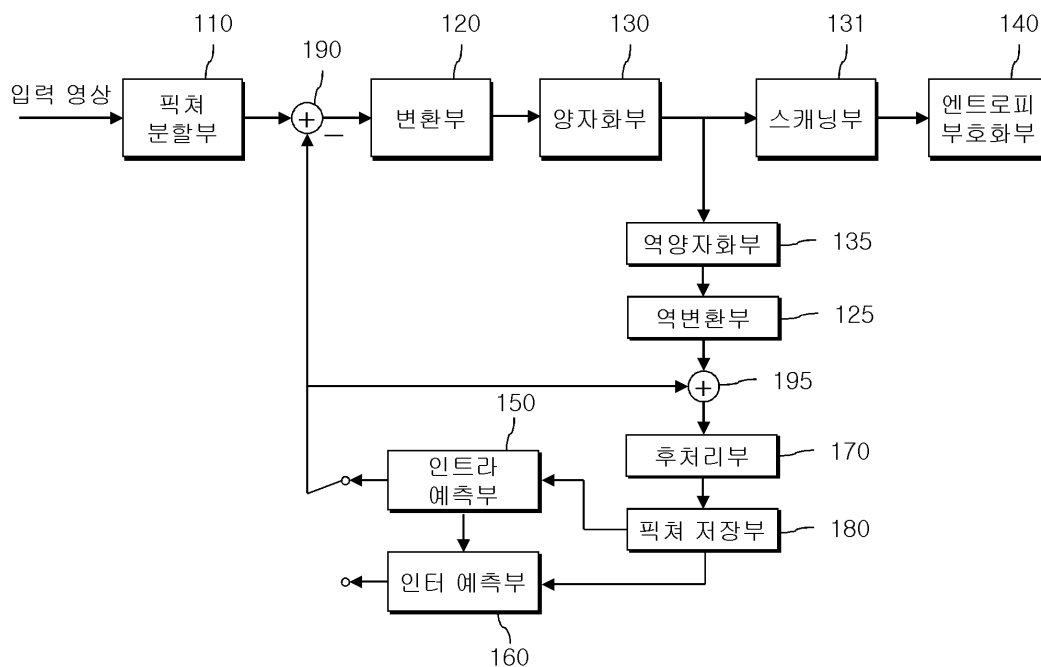
[0226] 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다. 그리고, 상기 방법을 구현하기 위한 기능적인(function) 프로그램, 코드 및 코드 세그먼트들은 본 발명이 속하는 기술분야의 프로그래머들에 의해 용이하게 추론될 수 있다.

[0227] 또한, 이상에서는 본 발명의 바람직한 실시예에 대하여 도시하고 설명하였지만, 본 발명은 상술한 특정의 실시예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 의해 다양한 변형 실시가 가능한 것은 물론이고, 이러한 변형 실시들은 본 발명의 기술적 사상이나 전망으로부터 개별적으로 이해되어서는 안될 것이다.

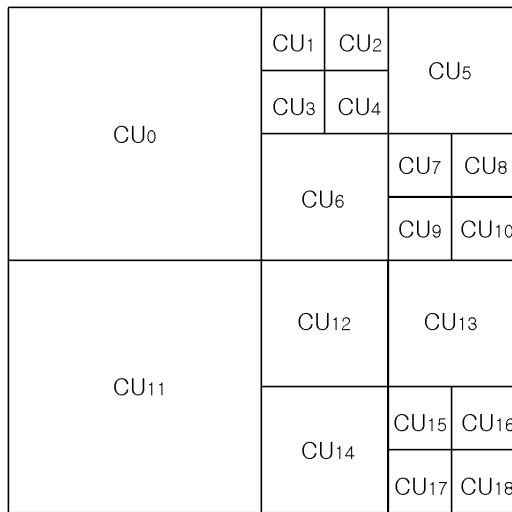
도면

도면1

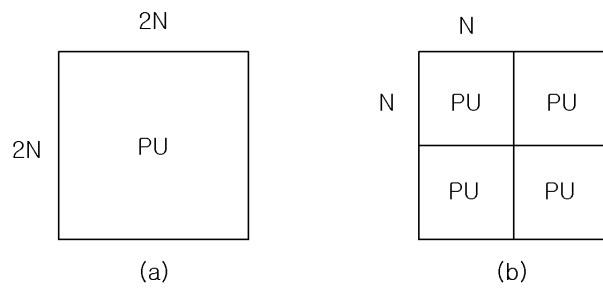
10



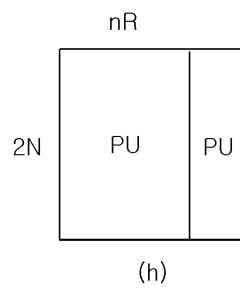
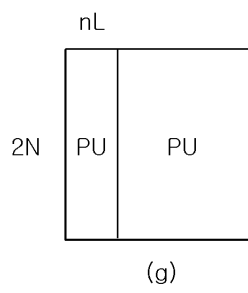
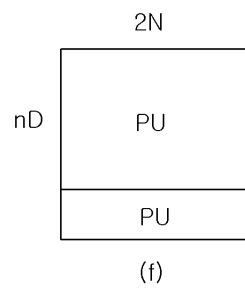
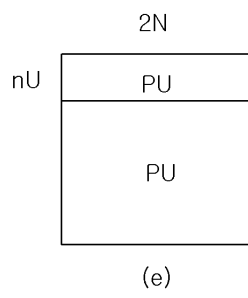
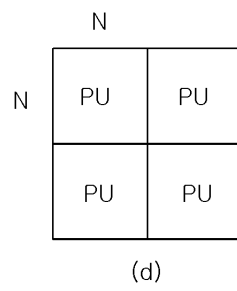
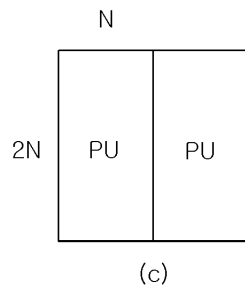
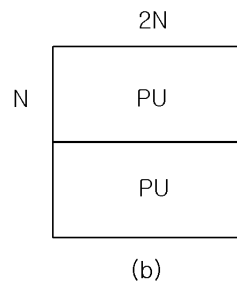
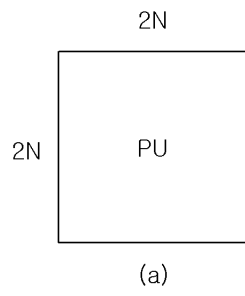
도면2



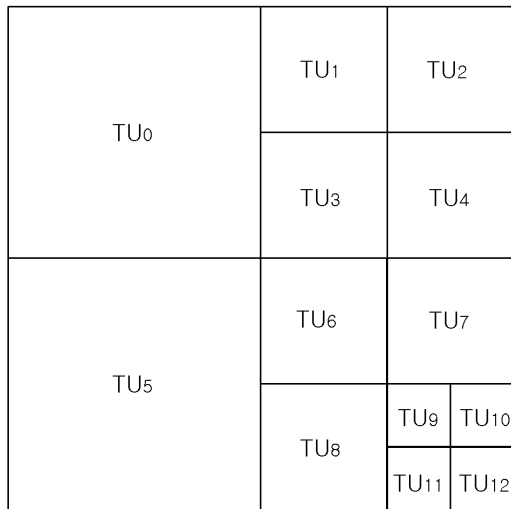
도면3



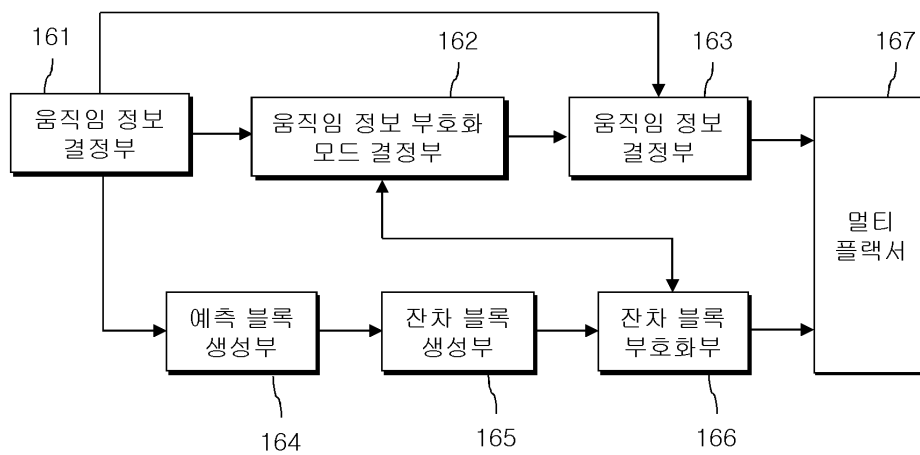
도면4



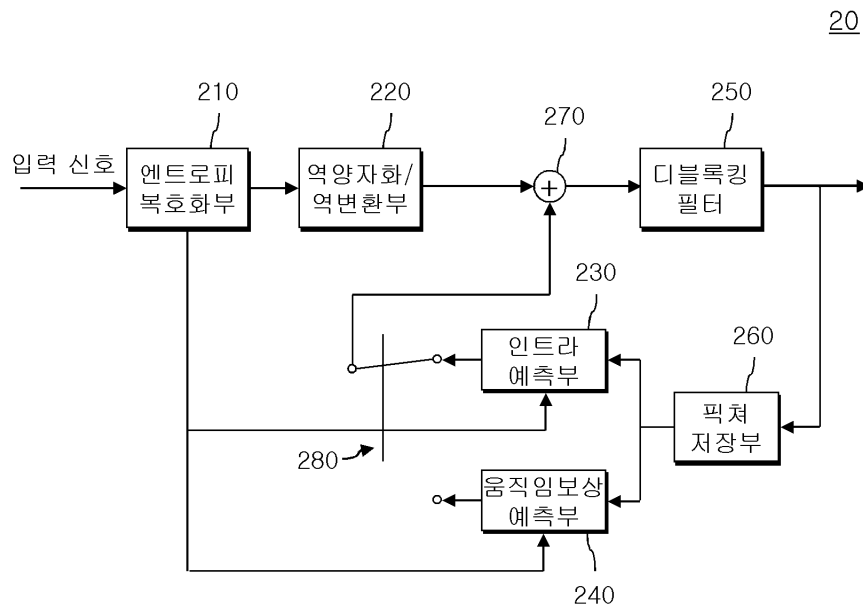
도면5



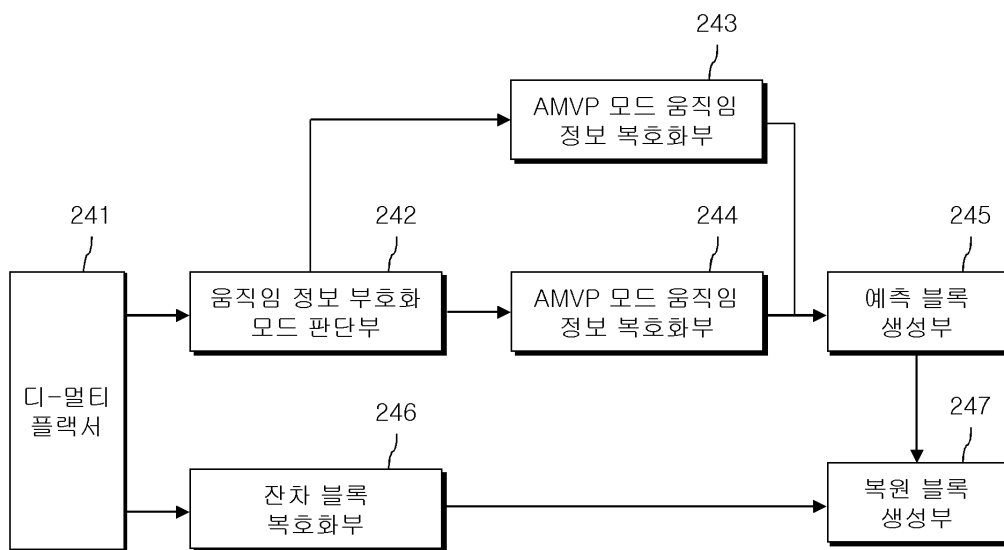
도면6



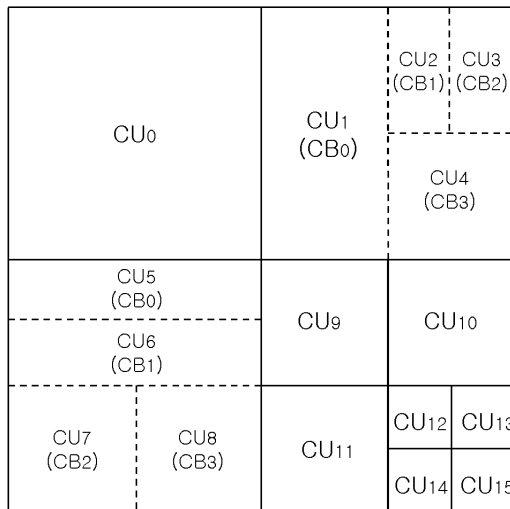
도면7



도면8



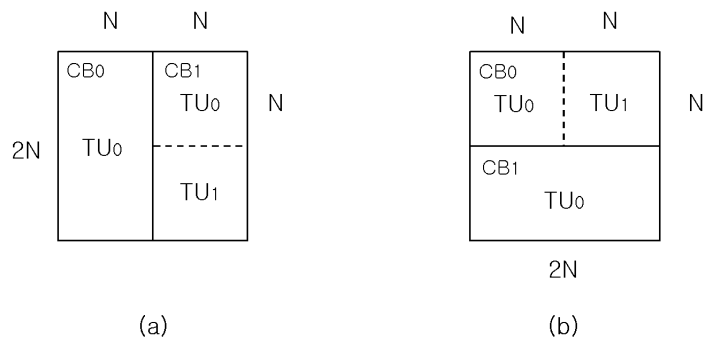
도면9



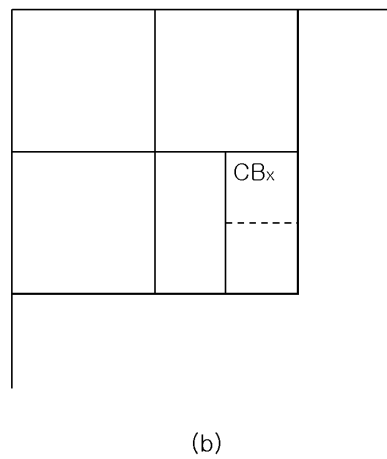
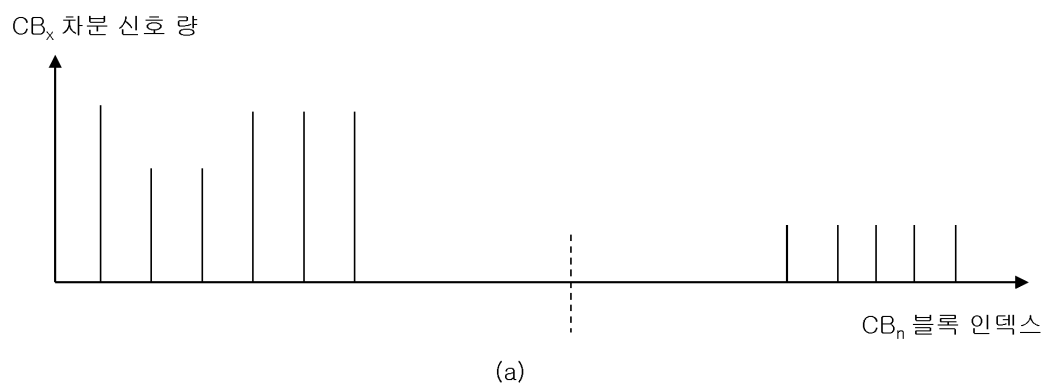
도면10

coding_quadtree (x0, y0, log2CbSize, cqtDepth){	Descriptor
if(x0 + (1 << log2CbSize) <= pic_width_in_luma_samples && y0 + (1 << log2CbSize) <= pic_height_in_luma_samples && log2CbSize > MinCbLog2SizeY){	
split_cu_flag [x0][y0]	ae(v)
binary_depth [x0][y0] // or Binary_split_flag	
}	
...	
if(split_cy_flag[x0][y0]){	
x1 = x0 + (1 << (log2CbSize - 1))	
y1 = y0 + (1 << (log2CbSize - 1))	
<i>coding_quadtree</i> (x0, y0, log2CbSize - 1, cqtDepth+1)	
if(x1 < pic_width_in_luma_samples)	
<i>coding_quadtree</i> (x1, y0, log2CbSize - 1, cqtDepth + 1)	
if(y1 < pic_height_in_luma_samples)	
<i>coding_quadtree</i> (x0, y1, log2CbSize - 1, cqtDepth + 1)	
if(x1 < pic_width_in_luma_samples && y1 < pic_height_in_luma_sampels)	
<i>coding_quadtree</i> (x1, y1, log2CbSize - 1, cqtDepth + 1)	
} else if (binary_depth == 0){	
<i>coding_unit</i> (x0, y0, log2CbSize)	
} else	
<i>coding_binarytree</i> (x0, y0, log2CbSize, cqtDepth, binary_depth /* Binary_split_flag*/)	
}	

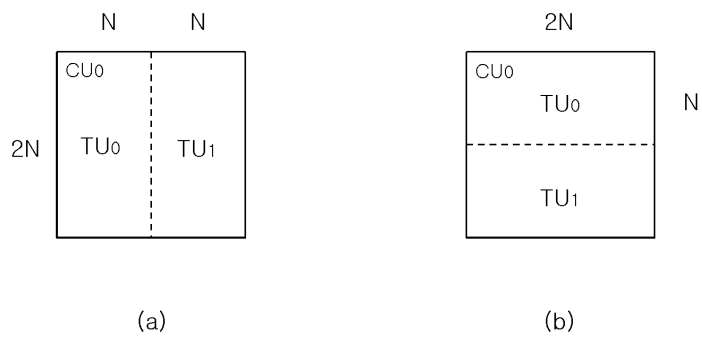
도면11



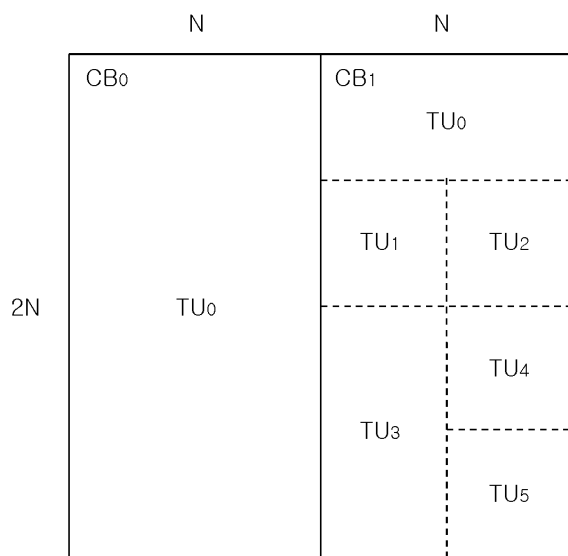
도면12



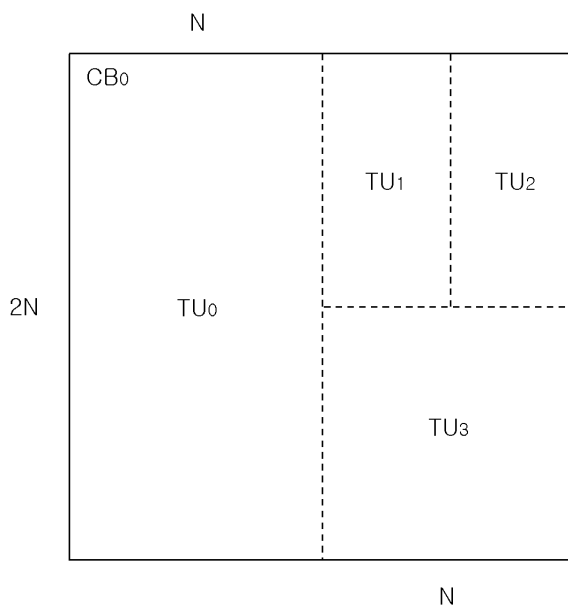
도면13



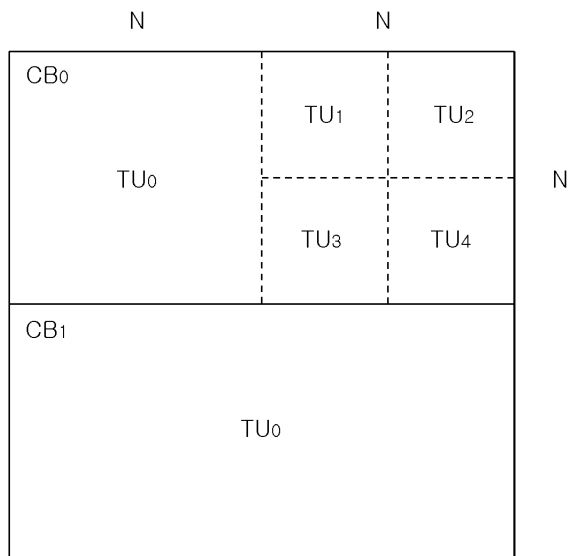
도면14



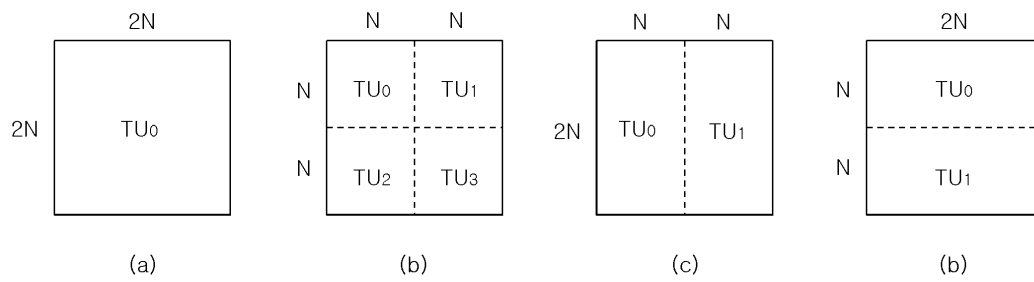
도면15



도면16



도면17



도면18

