(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0040815
(43) 공개일자 2021년04월14일

(51) 국제특허분류(Int. Cl.)
H04N 19/12 (2014.01) *H04N 19/176* (2014.01)
H04N 19/186 (2014.01) *H04N 19/60* (2014.01)
H04N 19/61 (2014.01) *H04N 19/625* (2014.01)
H04N 19/70 (2014.01)
(52) CPC특허분류
H04N 19/12 (2015.01)
H04N 19/176 (2015.01)
(21) 출원번호 10-2020-0127884
(22) 출원일자 2020년10월05일
심사청구일자 없음
(30) 우선권주장
1020190123489 2019년10월06일 대한민국(KR)
1020190123683 2019년10월07일 대한민국(KR)

(71) 출원인
현대자동차주식회사
서울특별시 서초구 헌릉로 12 (양재동)
이화여자대학교 산학협력단
서울특별시 서대문구 이화여대길 52 (대현동, 이화여자대학교)
기아 주식회사
서울특별시 서초구 헌릉로 12 (양재동)
(72) 발명자
강제원
서울특별시 마포구 서강로 53 서강쌍용예가아파트 104-904호
박승욱
경기도 용인시 수지구 태봉로 17, 403동 302호
임화평
경기도 화성시 남양읍 역골로 49번길 29 202호
(74) 대리인
이철희

전체 청구항 수 : 총 10 항

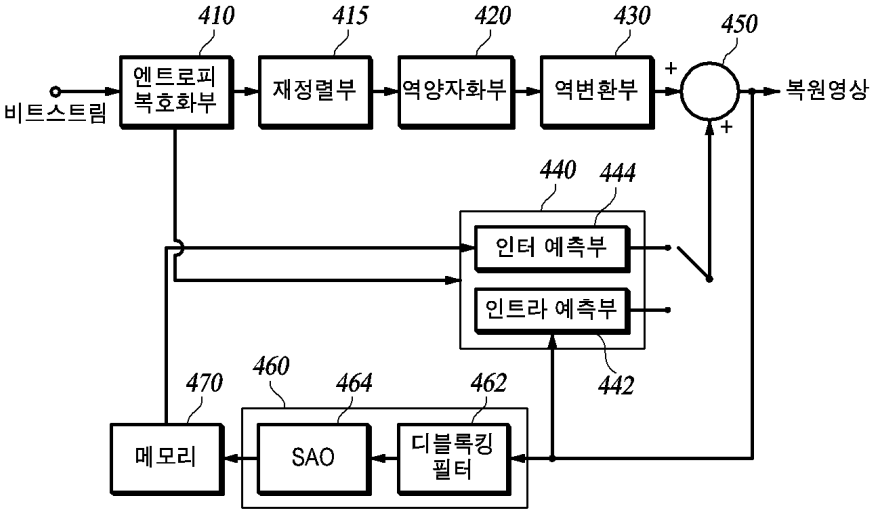
(54) 발명의 명칭 **현재블록의 변환계수들을 역변환하는 방법 및 장치**

(57) 요 약

현재블록의 변환계수들을 역변환하는 방법 및 장치를 개시한다.

본 발명의 일 실시예에 의하면, 현재블록의 변환계수들을 역변환하는 방법으로서, 비트스트림의 SPS(sequence parameter set) 레벨로부터, 인트라 예측 모드의 MTS(multiple transform selection)를 제어하는 하나 이상의 인트라 MTS 신택스 요소들 및 인터 예측 모드의 MTS를 제어하는 하나 이상의 인터 MTS 신택스 요소들을 복호화하는 단계; 상기 현재블록의 예측 모드, 상기 하나 이상의 인트라 MTS 신택스 요소들 및 상기 하나 이상의 인터 MTS 신택스 요소들에 기반하여, 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 결정하는 단계; 및 상기 결정된 하나 이상의 변환 커널들을 이용하여 상기 변환계수들을 역변환하는 단계를 포함하는 것을 특징으로 하는 방법을 제공한다.

대 표 도 - 도4



(52) CPC특허분류
H04N 19/186 (2015.01)
H04N 19/60 (2015.01)
H04N 19/61 (2015.01)
H04N 19/625 (2015.01)
H04N 19/70 (2015.01)

명 세 서

청구범위

청구항 1

현재블록의 변환계수들을 역변환하는 방법으로서,

비트스트림의 SPS(sequence parameter set) 레벨로부터, 인트라 예측 모드의 MTS(multiple transform selection)를 제어하는 하나 이상의 인트라 MTS 신택스 요소들 및 인터 예측 모드의 MTS를 제어하는 하나 이상의 인터 MTS 신택스 요소들을 복호화하는 단계;

상기 현재블록의 예측 모드, 상기 하나 이상의 인트라 MTS 신택스 요소들 및 상기 하나 이상의 인터 MTS 신택스 요소들에 기반하여, 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 결정하는 단계; 및

상기 결정된 하나 이상의 변환 커널들을 이용하여 상기 변환계수들을 역변환하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 2

제1항에 있어서,

상기 인트라 MTS 신택스 요소들은,

상기 인트라 예측 모드의 MTS가 인에이블되는지 여부를 지시하는 인트라 MTS 인에이블 플래그; 및

상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를 지시하는 인트라 MTS 선택 신택스 요소를 포함하고,

상기 인트라 MTS 선택 신택스 요소는,

상기 인트라 MTS 인에이블 플래그에 의해 상기 인트라 예측 모드의 MTS가 인에이블됨이 지시되는 경우에, 상기 비트스트림의 SPS 레벨로부터 복호화되는 것을 특징으로 하는 방법.

청구항 3

제1항에 있어서,

상기 인터 MTS 신택스 요소들은,

상기 인터 예측 모드의 MTS가 인에이블되는지 여부를 지시하는 인터 MTS 인에이블 플래그; 및

상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를 지시하는 인터 MTS 선택 신택스 요소를 포함하고,

상기 인터 MTS 선택 신택스 요소는,

상기 인터 MTS 인에이블 플래그에 의해 상기 인터 예측 모드의 MTS가 인에이블됨이 지시되는 경우에, 상기 비트스트림의 SPS 레벨로부터 복호화되는 것을 특징으로 하는 방법.

청구항 4

제1항에 있어서,

상기 인트라 MTS 신택스 요소들은,

서로 다른 세 개의 값을 나타내는 인트라 MTS 선택 신택스 요소를 포함하고,

상기 인트라 MTS 선택 신택스 요소는,

상기 인트라 예측 모드의 MTS가 인에이블되는지 여부 및, 상기 인트라 예측 모드의 MTS가 인에이블되는 경우에 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에

포함되는지 여부를, 상기 서로 다른 세 개의 값으로 나타내는 것을 특징으로 하는 방법.

청구항 5

제1항에 있어서,

상기 인터 MTS 신택스 요소들은,

서로 다른 세 개의 값을 나타내는 인터 MTS 선택 신택스 요소를 포함하고,

상기 인터 MTS 선택 신택스 요소는,

상기 인터 예측 모드의 MTS가 인에이블되는지 여부 및, 상기 인터 예측 모드의 MTS가 인에이블되는 경우에 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를, 상기 서로 다른 세 개의 값으로 나타내는 것을 특징으로 하는 방법.

청구항 6

복호화 장치로서,

인트라 예측 모드의 MTS(multiple transform selection)를 제어하는 하나 이상의 인트라 MTS 신택스 요소들 및 인터 예측 모드의 MTS를 제어하는 하나 이상의 인터 MTS 신택스 요소들을 비트스트림의 SPS(sequence parameter set) 레벨로부터 복호화하는 복호화부; 및

상기 현재블록의 예측 모드, 상기 하나 이상의 인트라 MTS 신택스 요소들 및 상기 하나 이상의 인터 MTS 신택스 요소들에 기반하여, 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 결정하고, 상기 결정된 하나 이상의 변환 커널들을 이용하여 상기 변환계수들을 역변환하는 역변환부를 포함하는 것을 특징으로 하는 장치.

청구항 7

제6항에 있어서,

상기 인트라 MTS 신택스 요소들은,

상기 인트라 예측 모드의 MTS가 인에이블되는지 여부를 지시하는 인트라 MTS 인에이블 플래그; 및

상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를 지시하는 인트라 MTS 선택 신택스 요소를 포함하고,

상기 인트라 MTS 선택 신택스 요소는,

상기 인트라 MTS 인에이블 플래그에 의해 상기 인트라 예측 모드의 MTS가 인에이블됨이 지시되는 경우에, 상기 비트스트림의 SPS 레벨로부터 복호화되는 것을 특징으로 하는 장치.

청구항 8

제6항에 있어서,

상기 인터 MTS 신택스 요소들은,

상기 인터 예측 모드의 MTS가 인에이블되는지 여부를 지시하는 인터 MTS 인에이블 플래그; 및

상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를 지시하는 인터 MTS 선택 신택스 요소를 포함하고,

상기 인터 MTS 선택 신택스 요소는,

상기 인터 MTS 인에이블 플래그에 의해 상기 인터 예측 모드의 MTS가 인에이블됨이 지시되는 경우에, 상기 비트스트림의 SPS 레벨로부터 복호화되는 것을 특징으로 하는 장치.

청구항 9

제6항에 있어서,

상기 인트라 MTS 신택스 요소들은,

서로 다른 세 개의 값을 나타내는 인트라 MTS 선택 신택스 요소를 포함하고,

상기 인트라 MTS 선택 신택스 요소는,

상기 인트라 예측 모드의 MTS가 인에이블되는지 여부 및, 상기 인트라 예측 모드의 MTS가 인에이블되는 경우에 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를, 상기 서로 다른 세 개의 값으로 나타내는 것을 특징으로 하는 장치.

청구항 10

제6항에 있어서,

상기 인터 MTS 신택스 요소들은,

서로 다른 세 개의 값을 나타내는 인터 MTS 선택 신택스 요소를 포함하고,

상기 인터 MTS 선택 신택스 요소는,

상기 인터 예측 모드의 MTS가 인에이블되는지 여부 및, 상기 인터 예측 모드의 MTS가 인에이블되는 경우에 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 지시하기 위한 MTS 인덱스가 상기 비트스트림에 포함되는지 여부를, 상기 서로 다른 세 개의 값으로 나타내는 것을 특징으로 하는 장치.

발명의 설명

기 술 분 야

[0001] 본 발명은 동영상의 부호화 및 복호화에 관한 것으로서, 더욱 구체적으로는 변환에 관련된 코딩 툴들을 효율적으로 제어함으로써 부호화 및 복호화의 효율성을 더욱 향상시키는 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] 동영상 데이터는 음성 데이터나 정지 영상 데이터 등에 비하여 많은 데이터량을 가지기 때문에, 압축을 위한 처리 없이 그 자체를 저장하거나 전송하기 위해서는 메모리를 포함하여 많은 하드웨어 자원을 필요로 한다.

[0003] 따라서, 통상적으로 동영상 데이터를 저장하거나 전송할 때에는 부호화기를 사용하여 동영상 데이터를 압축하여 저장하거나 전송하며, 복호화기에서는 압축된 동영상 데이터를 수신하여 압축을 해제하고 재생한다. 이러한 동영상 압축 기술로는 H.264/AVC를 비롯하여, H.264/AVC에 비해 약 40% 정도의 부호화 효율을 향상시킨 HEVC(High Efficiency Video Coding)가 존재한다.

[0004] 그러나, 영상의 크기 및 해상도, 프레임율이 점차 증가하고 있고, 이에 따라 부호화해야 하는 데이터량도 증가하고 있으므로 기존의 압축 기술보다 더 부호화 효율이 좋고 화질 개선 효과도 높은 새로운 압축 기술이 요구된다.

발명의 내용

해결하려는 과제

[0005] 이러한 요구에 부응하기 위해 본 발명은 개선된 부호화 및 복호화 기술을 제공하는 것을 목적으로 하며, 특히, 본 발명의 일 측면은 상위 레벨에서 정의되는 신택스 요소를 통해 다중 변환 선택(MTS)을 제어함으로써 부호화 및 복호화의 효율을 향상시키는 기술과 관련된다.

과제의 해결 수단

[0006] 본 발명의 일 측면은, 현재블록의 변환계수들을 역변환하는 방법으로서, 비트스트림의 SPS(sequence parameter set) 레벨로부터, 인트라 예측 모드의 MTS(multiple transform selection)를 제어하는 하나 이상의 인트라 MTS 신택스 요소들 및 인터 예측 모드의 MTS를 제어하는 하나 이상의 인터 MTS 신택스 요소들을 복호화하는 단계; 상기 현재블록의 예측 모드, 상기 하나 이상의 인트라 MTS 신택스 요소들 및 상기 하나 이상의 인터 MTS 신택스 요소들에 기반하여, 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 결정하는 단계; 및 상기 결

정된 하나 이상의 변환 커널들을 이용하여 상기 변환계수들을 역변환하는 단계를 포함하는 것을 특징으로 하는 방법을 제공한다.

[0007] 본 발명의 다른 일 측면은, 복호화 장치로서, 인트라 예측 모드의 MTS(multiple transform selection)를 제어하는 하나 이상의 인트라 MTS 신택스 요소들 및 인터 예측 모드의 MTS를 제어하는 하나 이상의 인터 MTS 신택스 요소들을 비트스트림의 SPS(sequence parameter set) 레벨로부터 복호화하는 복호화부; 및 상기 현재블록의 예측 모드, 상기 하나 이상의 인트라 MTS 신택스 요소들 및 상기 하나 이상의 인터 MTS 신택스 요소들에 기반하여, 상기 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 결정하고, 상기 결정된 하나 이상의 변환 커널들을 이용하여 상기 변환계수들을 역변환하는 역변환부를 포함하는 것을 특징으로 하는 장치를 제공한다.

발명의 효과

[0008] 이상에서 설명한 바와 같이 본 발명의 일 실시예에 의하면, MTS를 인트라 예측, 인터 예측, ISP, SBT 등에 개별적으로 적용할 수 있게 되므로, 부호화 및 복호화에 대한 효율성을 향상시킬 수 있다.

[0009] 또한, 본 발명의 다른 일 실시예에 의하면, 저주파 비분리 변환(LFNST)의 적용 여부를 종래 방법에 비해 더욱 신속하게 결정할 수 있으므로, 부호화 및 복호화의 지연 문제를 해소할 수 있다.

도면의 간단한 설명

[0010] 도 1은 본 개시의 기술들을 구현할 수 있는 영상 부호화 장치에 대한 예시적인 블록도이다.

도 2는 QTBTTT 구조를 이용하여 블록을 분할하는 방법을 설명하기 위한 도면이다.

도 3a는 복수의 인트라 예측 모드들을 나타낸 도면이다.

도 3b는 광각 인트라 예측모드들을 포함한 복수의 인트라 예측 모드들을 나타낸 도면이다.

도 4는 본 개시의 기술들을 구현할 수 있는 영상 복호화 장치의 예시적인 블록도이다.

도 5는 MTS를 상위 레벨에서 제어하는 본 발명의 일 예를 설명하기 위한 순서도이다.

도 6 내지 도 10은 MTS를 상위 레벨에서 제어하는 본 발명의 다양한 예들을 설명하기 위한 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0011] 이하, 본 발명의 일부 실시예들을 예시적인 도면을 통해 상세하게 설명한다. 각 도면의 구성 요소들에 식별 부호를 부가함에 있어서, 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 부호를 가지도록 하고 있음에 유의해야 한다. 또한, 본 발명을 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략한다.

[0013] 도 1은 본 개시의 기술들을 구현할 수 있는 영상 부호화 장치에 대한 예시적인 블록도이다. 이하에서는 도 1을 참조하여 영상 부호화 장치와 이 장치의 하위 구성들에 대하여 설명하도록 한다.

[0014] 영상 부호화 장치는 픽처 분할부(110), 예측부(120), 감산기(130), 변환부(140), 양자화부(145), 재정렬부(150), 엔트로피 부호화부(155), 역양자화부(160), 역변환부(165), 가산기(170), 필터부(180) 및 메모리(190)를 포함하여 구성될 수 있다.

[0015] 영상 부호화 장치의 각 구성요소는 하드웨어 또는 소프트웨어로 구현되거나, 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다. 또한, 각 구성요소의 기능이 소프트웨어로 구현되고 마이크로프로세서가 각 구성요소에 대응하는 소프트웨어의 기능을 실행하도록 구현될 수도 있다.

[0016] 하나의 영상(비디오)은 복수의 픽처들로 구성된다. 각 픽처들은 복수의 영역으로 분할되고 각 영역마다 부호화가 수행된다. 예를 들어, 하나의 픽처는 하나 이상의 타일(Tile) 또는/및 슬라이스(Slice)로 분할된다. 여기서, 하나 이상의 타일은 타일 그룹(Tile Group)으로 정의될 수 있다. 각 타일 또는 슬라이스는 하나 이상의 CTU(Coding Tree Unit)로 분할된다. 그리고 각 CTU는 트리 구조에 의해 하나 이상의 CU(Coding Unit)들로 분할된다. 각 CU에 적용되는 정보들은 CU의 신택스로서 부호화되고, 하나의 CTU에 포함된 CU들에 공통적으로 적용되는 정보는 CTU의 신택스로서 부호화된다. 또한, 하나의 슬라이스 내의 모든 블록들에 공통적으로 적용되는 정보

는 슬라이스 헤더의 신택스로서 부호화되며, 하나의 픽처를 구성하는 모든 블록들에 적용되는 정보는 픽처 파라미터 셋(PPS, Picture Parameter Set) 혹은 픽처 헤더에 부호화된다. 나아가, 복수의 픽처가 공통으로 참조하는 정보들은 시퀀스 파라미터 셋(SPS, Sequence Parameter Set)에 부호화된다. 그리고, 하나 이상의 SPS가 공통으로 참조하는 정보들은 비디오 파라미터 셋(VPS, Video Parameter Set)에 부호화된다. 또한, 하나의 타일 또는 타일 그룹에 공통으로 적용되는 정보는 타일 또는 타일 그룹 헤더의 신택스로서 부호화될 수도 있다.

[0017] 픽처 분할부(110)는 CTU(Coding Tree Unit)의 크기를 결정한다. CTU의 크기에 대한 정보(CTU size)는 SPS 또는 PPS의 신택스로서 부호화되어 영상 복호화 장치로 전달된다.

[0018] 픽처 분할부(110)는 영상을 구성하는 각 픽처(picture)를 미리 결정된 크기를 가지는 복수의 CTU(Coding Tree Unit)들로 분할한 이후에, 트리 구조(tree structure)를 이용하여 CTU를 반복적으로(recursively) 분할한다. 트리 구조에서의 리프 노드(leaf node)가 부호화의 기본 단위인 CU(coding unit)가 된다.

[0019] 트리 구조는 상위 노드(혹은 부모 노드)가 동일한 크기의 네 개의 하위 노드(혹은 자식 노드)들로 분할되는 쿼드트리(QuadTree, QT), 또는 상위 노드가 두 개의 하위 노드로 분할되는 바이너리트리(BinaryTree, BT), 또는 상위 노드가 1:2:1 비율을 가지는 세 개의 하위 노드로 분할되는 터너리트리(TernaryTree, TT), 또는 이러한 QT 구조, BT 구조 및 TT 구조 중 둘 이상을 혼용한 구조일 수 있다. 예컨대, QTBT(QuadTree plus BinaryTree) 구조가 사용될 수 있고, 또는 QTBTTT(QuadTree plus BinaryTree TernaryTree) 구조가 사용될 수 있다. 여기서, BTTT를 합쳐서 MTT(Multiple-Type Tree)라 지칭할 수 있다.

[0020] 도 2는 QTBTTT 분할 트리 구조를 보인다. 도 2에서 보는 바와 같이, CTU는 먼저 QT 구조로 분할될 수 있다. 쿼드트리 분할은 분할 블록(splitting block)의 크기가 QT에서 허용되는 리프 노드의 최소 블록 크기(MinQTSize)에 도달할 때까지 반복될 수 있다. QT 구조의 각 노드가 하위 레이어의 4개의 노드들로 분할되는지 여부를 지시하는 제1플래그(QT_split_flag)는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다.

[0021] QT의 리프 노드가 BT에서 허용되는 루트 노드의 최대 블록 크기(MaxBTSize)보다 크지 않은 경우, BT 구조 또는 TT 구조 중 어느 하나 이상으로 더 분할될 수 있다. BT 구조 및/또는 TT 구조에서는 복수의 분할 방향이 존재할 수 있다. 예컨대, 해당 노드의 블록이 가로로 분할되는 방향과 세로로 분할되는 방향 두 가지가 존재할 수 있다. 도 2와 같이, MTT 분할이 시작되면, 노드들이 분할되었는지 여부를 지시하는 제2플래그(mtt_split_flag)와, 분할이 되었다면 추가적으로 분할 방향(vertical 혹은 horizontal)을 나타내는 플래그 및/또는 분할 타입(Binary 혹은 Ternary)을 나타내는 플래그가 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 시그널링된다. 대안적으로, 각 노드가 하위 레이어의 4개의 노드들로 분할되는지 여부를 지시하는 제1플래그(QT_split_flag)를 부호화하기에 앞서, 그 노드가 분할되는지 여부를 지시하는 CU 분할 플래그(split_cu_flag)가 부호화될 수도 있다. CU 분할 플래그(split_cu_flag) 값이 분할되지 않았음을 지시하는 경우, 해당 노드의 블록이 분할 트리 구조에서의 리프 노드(leaf node)가 되어 부호화의 기본 단위인 CU(coding unit)가 된다. CU 분할 플래그(split_cu_flag) 값이 분할됨을 지시하는 경우, 영상 부호화 장치는 전술한 방식으로 제1플래그부터 부호화를 시작한다.

[0022] 트리 구조의 다른 예시로서 QTBT가 사용되는 경우, 해당 노드의 블록을 동일 크기의 두 개 블록으로 가로로 분할하는 타입(즉, symmetric horizontal splitting)과 세로로 분할하는 타입(즉, symmetric vertical splitting) 두 가지가 존재할 수 있다. BT 구조의 각 노드가 하위 레이어의 블록으로 분할되는지 여부를 지시하는 분할 플래그(split_flag) 및 분할되는 타입을 지시하는 분할 타입 정보가 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 전달된다. 한편, 해당 노드의 블록을 서로 비대칭 형태의 두 개의 블록으로 분할하는 타입이 추가로 더 존재할 수도 있다. 비대칭 형태에는 해당 노드의 블록을 1:3의 크기 비율을 가지는 두 개의 직사각형 블록으로 분할하는 형태가 포함될 수 있고, 혹은 해당 노드의 블록을 대각선 방향으로 분할하는 형태가 포함될 수도 있다.

[0023] CU는 CTU로부터의 QTBT 또는 QTBTTT 분할에 따라 다양한 크기를 가질 수 있다. 이하에서는, 부호화 또는 복호화하고자 하는 CU(즉, QTBTTT의 리프 노드)에 해당하는 블록을 '현재블록'이라 칭한다. QTBTTT 분할의 채용에 따라, 현재블록의 모양은 정사각형뿐만 아니라 직사각형일 수도 있다.

[0024] 예측부(120)는 현재블록을 예측하여 예측블록을 생성한다. 예측부(120)는 인트라 예측부(122)와 인터 예측부(124)를 포함한다.

[0025] 일반적으로, 픽처 내 현재블록들은 각각 예측적으로 코딩될 수 있다. 일반적으로 현재블록의 예측은 (현재블록

을 포함하는 픽처로부터의 데이터를 사용하는) 인트라 예측 기술 또는 (현재블록을 포함하는 픽처 이전에 코딩된 픽처로부터의 데이터를 사용하는) 인터 예측 기술을 사용하여 수행될 수 있다. 인터 예측은 단방향 예측과 양방향 예측 모두를 포함한다.

[0026] 인트라 예측부(122)는 현재블록이 포함된 현재 픽처 내에서 현재블록의 주변에 위치한 픽셀(참조 픽셀)들을 이용하여 현재블록 내의 픽셀들을 예측한다. 예측 방향에 따라 복수의 인트라 예측모드가 존재한다. 예컨대, 도 3a에서 보는 바와 같이, 복수의 인트라 예측모드는 planar 모드와 DC 모드를 포함하는 2개의 비방향성 모드와 65개의 방향성 모드를 포함할 수 있다. 각 예측모드에 따라 사용할 주변 픽셀과 연산식이 다르게 정의된다.

[0027] 직사각형 모양의 현재블록에 대한 효율적인 방향성 예측을 위해, 도 3b에 점선 화살표로 도시된 방향성 모드들(67 ~ 80번, -1 ~ -14 번 인트라 예측모드들)이 추가로 사용될 수 있다. 이들은 "광각 인트라 예측모드들(wide angle intra-prediction modes)"로 지칭될 수 있다. 도 3b에서 화살표들은 예측에 사용되는 대응하는 참조샘플들을 가리키는 것이며, 예측 방향을 나타내는 것이 아니다. 예측 방향은 화살표가 가리키는 방향과 반대이다. 광각 인트라 예측모드들은 현재블록이 직사각형일 때 추가적인 비트 전송 없이 특정 방향성 모드를 반대방향으로 예측을 수행하는 모드이다. 이때 광각 인트라 예측모드들 중에서, 직사각형의 현재블록의 너비와 높이의 비율에 의해, 현재블록에 이용 가능한 일부 광각 인트라 예측모드들이 결정될 수 있다. 예컨대, 45도보다 작은 각도를 갖는 광각 인트라 예측모드들(67 ~ 80번 인트라 예측모드들)은 현재블록이 높이가 너비보다 작은 직사각형 형태일 때 이용 가능하고, -135도보다 큰 각도를 갖는 광각 인트라 예측모드들(-1 ~ -14 번 인트라 예측모드들)은 현재블록이 높이가 너비보다 큰 직사각형 형태일 때 이용 가능하다.

[0028] 인트라 예측부(122)는 현재블록을 부호화하는데 사용할 인트라 예측모드를 결정할 수 있다. 일부 예들에서, 인트라 예측부(122)는 여러 인트라 예측모드들을 사용하여 현재블록을 인코딩하고, 테스트된 모드들로부터 사용할 적절한 인트라 예측모드를 선택할 수도 있다. 예를 들어, 인트라 예측부(122)는 여러 테스트된 인트라 예측모드들에 대한 레이트 왜곡(rate-distortion) 분석을 사용하여 레이트 왜곡 값들을 계산하고, 테스트된 모드들 중 최선의 레이트 왜곡 특징들을 갖는 인트라 예측모드를 선택할 수도 있다.

[0029] 인트라 예측부(122)는 복수의 인트라 예측모드 중에서 하나의 인트라 예측모드를 선택하고, 선택된 인트라 예측모드에 따라 결정되는 주변 픽셀(참조 픽셀)과 연산식을 사용하여 현재블록을 예측한다. 선택된 인트라 예측모드에 대한 정보는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 전달된다.

[0030] 인터 예측부(124)는 움직임 보상 과정을 통해 현재블록에 대한 예측블록을 생성한다. 인터 예측부(124)는 현재 픽처보다 먼저 부호화 및 복호화된 참조픽처 내에서 현재블록과 가장 유사한 블록을 탐색하고, 그 탐색된 블록을 이용하여 현재블록에 대한 예측블록을 생성한다. 그리고, 현재 픽처 내의 현재블록과 참조픽처 내의 예측블록 간의 변위(displacement)에 해당하는 움직임벡터(motion vector)를 생성한다. 일반적으로, 움직임 추정은 루마(luma) 성분에 대해 수행되고, 루마 성분에 기초하여 계산된 모션 벡터는 루마 성분 및 크로마 성분 모두에 대해 사용된다. 현재블록을 예측하기 위해 사용된 참조픽처에 대한 정보 및 움직임벡터에 대한 정보를 포함하는 움직임 정보는 엔트로피 부호화부(155)에 의해 부호화되어 영상 복호화 장치로 전달된다.

[0031] 감산기(130)는 현재블록으로부터 인트라 예측부(122) 또는 인터 예측부(124)에 의해 생성된 예측블록을 감산하여 잔차블록을 생성한다.

[0032] 변환부(140)는 잔차블록을 하나 이상의 변환블록들로 나누고, 변환을 하나 이상의 변환 블록들에 적용하여, 변환블록들의 잔차 값들을 픽셀 도메인에서 주파수 도메인으로 변환한다. 주파수 도메인에서, 변환된 블록들은 하나 이상의 변환 계수 값들을 포함하는 계수 블록들이라고 지칭된다. 변환에는 2차원 변환 커널이 사용될 수 있으며, 수평 방향 변환과 수직 방향 변환에 각각 1차원 변환 커널이 사용될 수도 있다. 변환 커널은 이산 코사인 변환(DCT), 이산 사인 변환(DST) 등에 기반할 수 있다.

[0033] 변환부(140)는 잔차블록의 전체 크기를 변환 단위로 사용하여 잔차블록 내의 잔차 신호들을 변환할 수 있다. 또한, 변환부(140)는 잔차블록을 수평 혹은 수직 방향으로 2개의 서브블록으로 분할하고, 변환을 2개의 서브블록 중 하나에만 수행할 수 있다. 따라서, 변환 블록의 사이즈는 잔차 블록의 사이즈(따라서 예측블록의 사이즈)와 상이할 수도 있다. 변환이 수행되지 않는 서브블록에는 논-제로 잔차 샘플 값들이 존재하지 않거나 매우 희소할 수 있다. 변환이 수행되지 않는 서브블록의 잔차 샘플들은 시그널링되지 않으며, 영상 복호화 장치에 의해 모두 "0"으로 간주될 수 있다. 분할 방향과 분할 비율에 따라 여러 파티션 타입들이 존재할 수 있다. 변환부(140)는 잔차블록의 코딩 모드(혹은 변환 모드)에 관한 정보(예컨대, 잔차 블록을 변환하였는지 아니면 잔차 서브블록을 변환하였는지를 나타내는 정보, 잔차블록을 서브블록들로 분할하기 위해 선택된 파티션 타입을 나타내는 정보,

변환이 수행되는 서브블록을 식별하는 정보 등을 포함하는 정보)를 엔트로피 부호화부(155)에 제공할 수 있다. 엔트로피 부호화부(155)는 잔차블록의 코딩 모드(혹은 변환 모드)에 관한 정보를 부호화할 수 있다.

[0034] 양자화부(145)는 변환부(140)로부터 출력되는 변환 계수들을 양자화하고, 양자화된 변환 계수들을 엔트로피 부호화부(155)로 출력한다. 양자화부(145)는, 어떤 블록 혹은 프레임에 대해, 변환 없이, 관련된 잔차 블록을 곧바로 양자화할 수도 있다.

[0035] 재정렬부(150)는 양자화된 잔차 값에 대해 계수 값의 재정렬을 수행할 수 있다. 재정렬부(150)는 계수 스캐닝(coefficient scanning)을 통해 2차원의 계수 어레이를 1차원의 계수 시퀀스로 변경할 수 있다. 예를 들어, 재정렬부(150)에서는 지그-재그 스캔(zig-zag scan) 또는 대각선 스캔(diagonal scan)을 이용하여 DC 계수부터 고주파수 영역의 계수까지 스캔하여 1차원의 계수 시퀀스를 출력할 수 있다. 변환 단위의 크기 및 인트라 예측 모드에 따라 지그-재그 스캔 대신 2차원의 계수 어레이를 열 방향으로 스캔하는 수직 스캔, 2차원의 블록 형태 계수를 행 방향으로 스캔하는 수평 스캔이 사용될 수도 있다. 즉, 변환 단위의 크기 및 인트라 예측 모드에 따라 지그-재그 스캔, 대각선 스캔, 수직 방향 스캔 및 수평 방향 스캔 중에서 사용될 스캔 방법이 결정될 수도 있다.

[0036] 엔트로피 부호화부(155)는, CABAC(Context-based Adaptive Binary Arithmetic Code), 지수 골롬(Exponential Golomb) 등의 다양한 부호화 방식을 사용하여, 재정렬부(150)로부터 출력된 1차원의 양자화된 변환 계수들의 시퀀스를 부호화함으로써 비트스트림을 생성한다.

[0037] 또한, 엔트로피 부호화부(155)는 블록 분할과 관련된 CTU size, CU 분할 플래그, QT 분할 플래그, MTT 분할 타입, MTT 분할 방향 등의 정보를 부호화하여, 영상 복호화 장치가 영상 부호화 장치와 동일하게 블록을 분할할 수 있도록 한다. 또한, 엔트로피 부호화부(155)는 현재블록이 인트라 예측에 의해 부호화되었는지 아니면 인터 예측에 의해 부호화되었는지 여부를 지시하는 예측 타입에 대한 정보를 부호화하고, 예측 타입에 따라 인트라 예측정보(즉, 인트라 예측모드에 대한 정보) 또는 인터 예측정보(참조픽처 및 움직임벡터에 대한 정보)를 부호화한다.

[0038] 역양자화부(160)는 양자화부(145)로부터 출력되는 양자화된 변환 계수들을 역양자화하여 변환 계수들을 생성한다. 역변환부(165)는 역양자화부(160)로부터 출력되는 변환 계수들을 주파수 도메인으로부터 공간 도메인으로 변환하여 잔차블록을 복원한다.

[0039] 가산부(170)는 복원된 잔차블록과 예측부(120)에 의해 생성된 예측블록을 가산하여 현재블록을 복원한다. 복원된 현재블록 내의 픽셀들은 다음 순서의 블록을 인트라 예측할 때 참조 픽셀로서 사용된다.

[0040] 필터부(180)는 블록 기반의 예측 및 변환/양자화로 인해 발생하는 블록킹 아티팩트(blocking artifacts), 링잉 아티팩트(ringing artifacts), 블러링 아티팩트(blurring artifacts) 등을 줄이기 위해 복원된 픽셀들에 대한 필터링을 수행한다. 필터부(180)는 디블록킹 필터(182)와 SAO(Sample Adaptive Offset) 필터(184)를 포함할 수 있다.

[0041] 디블록킹 필터(180)는 블록 단위의 부호화/복호화로 인해 발생하는 블록킹 현상(blocking artifact)을 제거하기 위해 복원된 블록 간의 경계를 필터링하고, SAO 필터(184)는 디블록킹 필터링된 영상에 대해 추가적인 필터링을 수행한다. SAO 필터(184)는 손실 부호화(lossy coding)로 인해 발생하는 복원된 픽셀과 원본 픽셀 간의 차이를 보상하기 위해 사용되는 필터이다.

[0042] 디블록킹 필터(182) 및 SAO 필터(184)를 통해 필터링된 복원블록은 메모리(190)에 저장된다. 한 픽처 내의 모든 블록들이 복원되면, 복원된 픽처는 이후에 부호화하고자 하는 픽처 내의 블록을 인터 예측하기 위한 참조 픽처로 사용될 수 있다.

[0044] 도 4는 본 개시의 기술들을 구현할 수 있는 영상 복호화 장치의 예시적인 블록도이다. 이하에서는 도 4를 참조하여 영상 복호화 장치와 이 장치의 하위 구성들에 대하여 설명하도록 한다.

[0045] 영상 복호화 장치는 엔트로피 복호화부(410), 재정렬부(415), 역양자화부(420), 역변환부(430), 예측부(440), 가산기(450), 필터부(460) 및 메모리(470)를 포함하여 구성될 수 있다.

[0046] 도 1의 영상 부호화 장치와 마찬가지로, 영상 복호화 장치의 각 구성요소는 하드웨어 또는 소프트웨어로 구현되거나, 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다. 또한, 각 구성요소의 기능이 소프트웨어로 구현되고

마이크로프로세서가 각 구성요소에 대응하는 소프트웨어의 기능을 실행하도록 구현될 수도 있다.

[0047] 엔트로피 복호화부(410)는 영상 부호화 장치에 의해 생성된 비트스트림을 복호화하여 블록 분할과 관련된 정보를 추출함으로써 복호화하고자 하는 현재블록을 결정하고, 현재블록을 복원하기 위해 필요한 예측정보와 잔차신호에 대한 정보 등을 추출한다.

[0048] 엔트로피 복호화부(410)는 SPS(Sequence Parameter Set) 또는 PPS(Picture Parameter Set)로부터 CTU size에 대한 정보를 추출하여 CTU의 크기를 결정하고, 픽처를 결정된 크기의 CTU로 분할한다. 그리고, CTU를 트리 구조의 최상위 레이어, 즉, 루트 노드로 결정하고, CTU에 대한 분할정보를 추출함으로써 트리 구조를 이용하여 CTU를 분할한다.

[0049] 예컨대, QTBTTT 구조를 사용하여 CTU를 분할하는 경우, 먼저 QT의 분할과 관련된 제1 플래그(QT_split_flag)를 추출하여 각 노드를 하위 레이어의 네 개의 노드로 분할한다. 그리고, QT의 리프 노드에 해당하는 노드에 대해서는 MTT의 분할과 관련된 제2 플래그(MTT_split_flag) 및 분할 방향(vertical / horizontal) 및/또는 분할 타입(binary / ternary) 정보를 추출하여 해당 리프 노드를 MTT 구조로 분할한다. 이를 통해 QT의 리프 노드 이하의 각 노드들을 BT 또는 TT 구조로 반복적으로(recursively) 분할한다.

[0050] 또 다른 예로서, QTBTTT 구조를 사용하여 CTU를 분할하는 경우, 먼저 CU의 분할 여부를 지시하는 CU 분할 플래그(split_cu_flag)를 추출하고, 해당 블록이 분할된 경우, 제1 플래그(QT_split_flag)를 추출할 수도 있다. 분할 과정에서 각 노드는 0번 이상의 반복적인 QT 분할 후에 0번 이상의 반복적인 MTT 분할이 발생할 수 있다. 예컨대, CTU는 바로 MTT 분할이 발생하거나, 반대로 다수 번의 QT 분할만 발생할 수도 있다.

[0051] 다른 예로서, QTBT 구조를 사용하여 CTU를 분할하는 경우, QT의 분할과 관련된 제1 플래그(QT_split_flag)를 추출하여 각 노드를 하위 레이어의 네 개의 노드로 분할한다. 그리고, QT의 리프 노드에 해당하는 노드에 대해서는 BT로 더 분할되는지 여부를 지시하는 분할 플래그(split_flag) 및 분할 방향 정보를 추출한다.

[0052] 한편, 엔트로피 복호화부(410)는 트리 구조의 분할을 통해 복호화하고자 하는 현재블록을 결정하게 되면, 현재블록이 인트라 예측되었는지 아니면 인터 예측되었는지를 지시하는 예측 타입에 대한 정보를 추출한다. 예측 타입 정보가 인트라 예측을 지시하는 경우, 엔트로피 복호화부(410)는 현재블록의 인트라 예측정보(인트라 예측모드)에 대한 신택스 엘리먼트를 추출한다. 예측 타입 정보가 인터 예측을 지시하는 경우, 엔트로피 복호화부(410)는 인터 예측정보에 대한 신택스 엘리먼트, 즉, 움직임벡터 및 그 움직임벡터가 참조하는 참조픽처를 나타내는 정보를 추출한다.

[0053] 한편, 엔트로피 복호화부(410)는 잔차블록의 코딩 모드에 관한 정보(예컨대, 잔차블록이 부호화되었는지 잔차블록의 서브블록만이 부호화되었는지 여부에 관한 정보, 잔차블록을 서브블록들로 분할하기 위해 선택된 파티션 타입을 나타내는 정보, 부호화된 잔차 서브블록을 식별하는 정보, 양자화 파라미터 등)를 비트스트림으로부터 추출한다. 또한, 엔트로피 복호화부(410)는 잔차신호에 대한 정보로서 현재블록의 양자화된 변환계수들에 대한 정보를 추출한다.

[0054] 재정렬부(415)는, 영상 부호화 장치에 의해 수행된 계수 스캐닝 순서의 역순으로, 엔트로피 복호화부(410)에서 엔트로피 복호화된 1차원의 양자화된 변환계수들의 시퀀스를 다시 2차원의 계수 어레이(즉, 블록)로 변경할 수 있다.

[0055] 역양자화부(420)는 양자화된 변환계수들을 역양자화하고, 역변환부(430)는, 잔차블록의 코딩 모드에 관한 정보를 기초로, 역양자화된 변환계수들을 주파수 도메인으로부터 공간 도메인으로 역변환하여 잔차신호들을 복원함으로써 현재블록에 대한 복원된 잔차블록을 생성한다.

[0056] 역변환부(430)는, 잔차블록의 코딩 모드에 관한 정보가 영상 부호화 장치에서 현재블록의 잔차블록이 부호화되었다고 지시하는 경우에, 역양자화된 변환계수들에 대해 현재블록의 사이즈(따라서, 복원될 잔차블록의 사이즈)를 변환 단위로 사용하여 역변환을 수행하여 현재블록에 대한 복원된 잔차블록을 생성한다.

[0057] 또한, 역변환부(430)는, 잔차블록의 코딩 모드에 관한 정보가 영상 부호화 장치에서 잔차블록의 하나의 서브블록만이 부호화되었다고 지시하는 경우에, 역양자화된 변환계수들에 대해, 변환된 서브블록의 사이즈를 변환 단위로 사용하여, 역변환을 수행하여 변환된 서브블록에 대한 잔차신호들을 복원하고, 비변환된 서브블록에 대한 잔차신호들을 "0" 값으로 채움으로써, 현재블록에 대한 복원된 잔차블록을 생성한다.

[0058] 예측부(440)는 인트라 예측부(442) 및 인터 예측부(444)를 포함할 수 있다. 인트라 예측부(442)는 현재블록의 예측 타입이 인트라 예측일 때 활성화되고, 인터 예측부(444)는 현재블록의 예측 타입이 인터 예측일 때 활성화

된다.

[0059] 인트라 예측부(442)는 엔트로피 복호화부(410)로부터 추출된 인트라 예측모드에 대한 신택스 엘리먼트로부터 복수의 인트라 예측모드 중 현재블록의 인트라 예측모드를 결정하고, 인트라 예측모드에 따라 현재블록 주변의 참조 픽셀들을 이용하여 현재블록을 예측한다.

[0060] 인터 예측부(444)는 엔트로피 복호화부(410)로부터 추출된 인트라 예측모드에 대한 신택스 엘리먼트를 이용하여 현재블록의 움직임벡터와 그 움직임벡터가 참조하는 참조픽처를 결정하고, 움직임벡터와 참조픽처를 이용하여 현재블록을 예측한다.

[0061] 가산기(450)는 역변환부(430)로부터 출력되는 잔차블록과 인터 예측부(444) 또는 인트라 예측부(442)로부터 출력되는 예측블록을 가산하여 현재블록을 복원한다. 복원된 현재블록 내의 픽셀들은 이후에 복호화할 블록을 인트라 예측할 때의 참조픽셀로서 활용된다.

[0062] 필터부(460)는 디블록킹 필터(462) 및 SAO 필터(464)를 포함할 수 있다. 디블록킹 필터(462)는 블록 단위의 복호화로 인해 발생하는 블록킹 현상(blocking artifact)를 제거하기 위해, 복원된 블록 간의 경계를 디블록킹 필터링한다. SAO 필터(464)는 손실 부호화(lossy coding)으로 인해 발생하는 복원된 픽셀과 원본 픽셀 간의 차이를 보상하기 위해, 디블록킹 필터링 이후의 복원된 블록에 대해 추가적인 필터링을 수행한다. 디블록킹 필터(462) 및 SAO 필터(464)를 통해 필터링된 복원블록은 메모리(470)에 저장된다. 한 픽처 내의 모든 블록들이 복원되면, 복원된 픽처는 이후에 부호화하고자 하는 픽처 내의 블록을 인터 예측하기 위한 참조 픽처로 사용된다.

[0064] 효율적인 비디오 압축을 위하여, 다양한 예측 모드를 통해 예측하고 남은 잔차신호들(잔차샘플들)에 대해 양자화 과정 또는 스케일링(scaling) 과정(이하, '스케일링 과정'으로 지칭하도록 한다)을 추가로 진행할 수 있다.

[0065] 잔차샘플들의 변환은 효율적인 영상 압축과 시각적 인지의 중요성을 고려하여, 변환(transform) 기법을 통해 잔차샘플들을 픽셀 도메인으로부터 주파수 도메인으로 변환하는 기술이다. 잔차샘플들의 역변환은 변환 기법(역변환 기법)을 통해 잔차샘플들을 주파수 도메인으로부터 픽셀 도메인으로 변환하는 기술이다.

[0066] 그러나, 스크린 콘텐츠와 같은 비자연 영상의 경우에는 이러한 변환/역변환 기법이 비효율적일 수 있으므로, 이러한 경우에는 변환/역변환 기법이 생략(변환 스킵)될 수 있다. 잔차샘플들에 대한 변환/역변환이 생략되는 경우에는 잔차샘플들에 대한 스케일링 과정만이 수행되거나, 또는 스케일링도 수행되지 않고 엔트로피 부호화/복호화 과정만이 수행될 수도 있다.

[0067] 종래 부호화/복호화 방법의 경우에, 변환블록의 크기가 4x4, 8x8, 16x16, 32x32로 정해져 있으며, 이 변환블록들에 대해 변환 또는 변환 스킵이 적용될 수 있다. 변환블록에 변환이 적용된 경우에, 영상 복호화 장치는 양자화된 변환계수(TransCoeffLevel[x][y])들을 역양자화하고, 역양자화된 변환계수(d[x][y])들을 주파수 도메인으로부터 공간 도메인으로 역변환하여 잔차샘플(r[x][y])들을 복원할 수 있다. 또한, 영상 복호화 장치는 복원된 잔차샘플들을 영상의 비트 뎁스에 따라 수정(shift)하여 수정된 잔차샘플들을 유도할 수 있다.

[0068] 종래 부호화/복호화 방법의 경우에, 크기가 4x4인 변환블록에 대해 변환 스킵이 적용되거나, 추가적인 신택스 요소에 따라 다른 크기의 변환블록에 대해서도 변환 스킵이 적용될 수 있다. 변환블록에 변환 스킵이 적용된 경우에, 영상 복호화 장치는 양자화된 변환계수(TransCoeffLevel[x][y])들을 역양자화하고, 역양자화된 변환계수(d[x][y])들에 shift 연산을 적용하여 잔차샘플(r[x][y])들을 복원할 수 있다. 또한, 영상 복호화 장치는 복원된 잔차샘플들을 영상의 비트 뎁스에 따라 수정(shift)하여 수정된 잔차샘플들을 유도할 수 있다. 여기서, 역양자화된 변환계수들에 적용된 shift 연산은 변환 기법을 대신하여 적용된 것이다.

[0069] 변환 스킵된 잔차샘플들에 대해 회전 기법이 적용되었는지 여부를 지시하는 플래그(예: transform_skip_rotation_enabled_flag)가 회전 기법이 적용되었음을 지시하는 경우에(transform_skip_rotation_enabled_flag==1)는, 변환 스킵된 잔차샘플들이 180도 회전될 수 있다. 따라서, 영상 복호화 장치는 대칭성(회전)을 고려하여 잔차샘플들을 반대 방향 또는 반대 순서로 스캔할 수 있다.

[0070] **다중 변환 선택(multiple transform selection, MTS)**

[0071] 잔차샘플들에 대해 변환 기법이 적용되는 경우에는 DCT-Ⅱ 변환 커널(변환 타입)이 잔차샘플들에 대해 일반적으로 적용된다. 다만, 잔차샘플들의 다양한 특성에 따라 보다 적절한 변환 기법을 적용하기 위하여, 여러 개의 변환 커널들 중에서 최적의 하나 또는 두 개의 변환 커널들이 선택적으로 잔차샘플들에 적용될 수도 있다.

[0072] 여러 개의 변환 커널들 중에서 최적의 하나 또는 두 개의 변환 커널들을 선택하여 이를 잔차샘플들에 적용하는 기법을 다중 변환 선택(MTS)라 지칭할 수 있다.

[0073] MTS는 4K 영상, 360도 동영상, 드론 영상 등과 같은 다양한 자연 영상에 대한 비트율 감소를 통해 네트워크의 부담을 경감시킬 수 있다. 또한, MTS는 다양한 자연 영상을 복호화하는 장치에 대한 에너지 소비를 절감시킬 뿐만 아니라, 빠른 복호화를 가능하도록 하기 위해 이용될 수 있다.

[0074] MTS에 이용될 수 있는 변환 커널들이 표 1에 나타나 있다.

표 1

[0075]

Transform Type	Basis function $T_i(j)$, $i, j = 0, 1, \ldots, N-1$
DCT-II	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{\pi \cdot i \cdot (2j+1)}{2N}\right)$ where, $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i = 0 \\ 1 & i \neq 0 \end{cases}$
DCT-VIII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \cos\left(\frac{\pi \cdot (2i+1) \cdot (2j+1)}{4N+2}\right)$
DST-VII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \sin\left(\frac{\pi \cdot (2i+1) \cdot (j+1)}{2N+1}\right)$

[0076] MTS의 사용 여부를 결정하는 신택스 요소들이 부호화되어 영상 부호화 장치로부터 영상 복호화 장치로 시그널링될 수 있다. MTS의 사용 여부를 결정하는 신택스 요소(mts_cu_flag)를 통해 MTS의 제어가 블록 단위(블록 레벨)에서 수행될 수 있으나, 보다 상위 레벨인 SPS 레벨에서 MTS의 활성화 여부를 결정하는 신택스 요소(sps_mts_enabled_flag)를 통해서도 MTS가 제어될 수 있다. 이 경우, mts_cu_flag는 SPS 레벨에서 MTS가 활성화되는 경우에(sps_mts_enabled_flag==1) 시그널링 및 복호화될 수 있다. MTS는 루마 성분에 대해서만 적용될 수 있으며, 현재블록의 폭(width, 가로 방향 길이)과 높이(height, 세로 방향 길이)가 모두 32 이하이면서, cbf 플래그가 1인 경우에 적용될 수 있다.

[0077] MTS가 적용되지 않는 경우에는 가로 방향(horizontal)의 변환 커널과 세로(vertical) 방향의 변환 커널이 모두 DCT-Ⅱ 변환 커널로 결정될 수 있다. 이와 달리, MTS가 적용되는 경우에는 명시적 MTS(explicit MTS) 및 묵시적 MTS(implicit MTS) 중에서 어느 하나의 방식이 역변환을 위해 적용될 수 있다.

[0078] 명시적 MTS는 변환블록(변환계수들)에 사용할 변환 커널을 명시적으로 전송하는 방법이다. 변환블록에 사용할 변환 커널은 영상 부호화 장치로부터 시그널링되는 인덱스를 통해 지시된다. 이때, 가로 방향의 변환 커널과 세로 방향의 변환 커널을 지시하기 위한 신택스 요소들(mts_hor_flag 및 mts_ver_flag)이 시그널링될 수 있다. mts_hor_flag 및 mts_ver_flag를 통해 가로 방향에 적용되는 변환 커널과 세로 방향에 적용되는 변환 커널이 서로 다르게 선택될 수 있다. mts_cu_flag, mts_hor_flag 및 mts_ver_flag 간의 맵핑 테이블이 표 2에 나타나 있다.

표 2

mts_cu_flag	mts_hor_flag	mts_ver_flag	Intra/Inter	
			Horizontal	Vertical
0			DCT-II	
1	0	0	DST-VII	DST-VII
	0	1	DCT-VIII	DST-VII
	1	0	DST-VII	DCT-VIII
	1	1	DST-VII	DCT-VIII

[0079]

[0080] 위와 같이 변환 커널을 명시적으로 시그널링하는 명시적 MTS 외에, 변환 커널을 묵시적으로 지시하는 묵시적 MTS가 적용될 수도 있다.

[0081] 묵시적 MTS에서, 가로 방향의 변환 커널과 세로 방향의 변환 커널을 지시하기 위한 변환 타입 쌍(trTypeHor 및 trTypeVer)은 아래 수학식 1을 통해 유도될 수 있다.

수학식 1

[0082]
$$trTypeHor = (nTbW >= 4 \ \&\& \ nTbW <= 16) \ ? \ DST\text{-}VII : DCT\text{-}II$$

$$trTypeVer = (nTbH >= 4 \ \&\& \ nTbH <= 16) \ ? \ DST\text{-}VII : DCT\text{-}II$$

[0083] 수학식 1에서, nTbW와 nTbH 각각은 변환블록의 가로 방향 길이(너비)와 세로 방향 길이(높이)를 나타낸다.

[0084] 변환 타입 쌍(trTypeHor 및 trTypeVer)은 (DST7, DST7), (DST7, DCT2), (DCT2, DST7), (DCT2, DCT2)로 정의될 수 있다.

[0085] 묵시적 MTS는 특정 부호화/복호화 기술이 적용되는 경우에 적용될 수 있다. 예를 들어, 현재블록이 ISP(intra sub-partition)로 부호화/복호화되는 경우에는 DCT-II 및 DST-VII이 적용되며, 현재블록이 LFNST(low-frequence non-separable transform) 및 MIP(Matrix-weighted intra prediction)로 부호화/복호화되는 경우에는 MTS가 적용되지 않고 변환 커널이 DCT-II로 결정될 수 있다.

[0086] 한편, 본 명세서에서, 'MTS가 적용되지 않는 경우'란 명시적 MTS 및 묵시적 MTS가 적용되지 않고 DCT-II를 변환 커널로 결정하는 방식을 의미한다. 또한, '명시적 MTS'란 영상 부호화 장치가 변환 커널을 지시하는 인덱스를 시그널링하고 영상 복호화 장치가 시그널링된 인덱스가 지시하는 변환 커널을 역변환에 적용하는 방식을 의미한다. 나아가, '묵시적 MTS'란 영상 부호화 장치로부터 변환 커널을 지시하는 인덱스가 시그널링되지 않고, 미리 설정된 조건에 따라 변환 커널을 유도하여 사용하는 방식을 의미한다.

[0087] **인트라 서브 파티션(intra sub-partition, ISP)**

[0088] ISP는 현재블록을 그 크기에 따라 가로 방향 또는 세로 방향으로 둘 또는 넷의 하위 영역(직사각형)으로 분할하고, 분할된 하위 영역들에 대해 인트라 예측을 수행하는 기법을 의미한다. ISP는 휘도 성분을 포함하는 블록(luma intra block)에 대해 적용될 수 있다.

[0089] ISP가 적용될 수 있는 블록의 최소 크기는 4x8 또는 8x4이며, 해당 블록의 크기가 최소 크기와 같다면 해당 블록이 2개의 영역으로 분할될 수 있다. 여기서, 분할된 각 영역들은 최소 16개의 샘플을 가질 수 있다. 해당 블록의 크기가 최소 크기를 초과하면, 해당 블록이 4개의 영역으로 분할될 수 있다. 분할된 하위 영역들에 대해서는 서로 같은 인트라 예측 모드가 적용될 수 있다.

[0090] ISP와 다른 부호화/복호화 기법 간의 관계는 아래와 같다.

[0091] - 다중 참조 라인(Multiple reference line, MRL): MRL의 인덱스가 0이 아니면(즉, 예측블록에 바로 인접한 라인 내 샘플을 참조하지 않는 경우), ISP가 적용되지 않는 것으로 추론되어 ISP에 관련된 신택스 요소들이 시그널링되지 않는다.

[0092] - 엔트로피 코딩의 계수 그룹(transform coefficient group): ISP가 적용되는 경우에는 엔트로피 코딩의 서브블록이 아래 표 3과 같이 가능한 모든 경우에 16개의 샘플을 가진다.

표 3

[0093]

Block Size	Coefficient group Size
$1 \times N, N \geq 16$	1×16
$N \times 1, N \geq 16$	16×1
$2 \times N, N \geq 8$	2×8
$N \times 2, N \geq 8$	8×2
All other possible $M \times N$ cases	4×4

[0094] - CBF 코딩: ISP가 적용되는 경우, 하위 영역들 중에서 적어도 하나가 non-zero CBF를 갖는 것으로 추론될 수 있다. 따라서, 하위 영역들의 총 개수가 n이고 앞선 n-1개의 하위 영역들이 zero CBF라면, 가장 마지막의 하위 영역(n번째 하위 영역)은 non-zero CBF로 추론될 수 있다.

[0095] - MPM: MPM 리스트는 DC 모드를 제외하고, ISP 수평 분할의 경우에는 수평 방향의 인트라 예측이 우선 순위를 가지며, ISP 수직 분할의 경우에는 수직 방향의 인트라 예측이 우선 순위를 가지도록 설정될 수 있다.

[0096] - 변환 크기 제한: ISP에서 하위 영역들에 적용되는 16보다 큰 모든 변환 커널은 DCT-II로 결정될 수 있다.

[0097] - PDPC: 현재블록에 ISP가 적용되는 경우, PDPC 필터가 하위 영역에 적용되지 않을 수 있다.

[0098] - MTS: 현재블록에 ISP가 적용되는 경우, mts_cu_flag가 묵시적으로 0으로 설정될 수 있다. 대신, ISP에 대한 변환 커널은 인트라 예측 모드와 블록의 크기에 따라 고정되어 선택될 수 있다. wxh 크기의 하위 영역에 대해 선택되는 변환 커널은 아래와 같다.

[0099] w==1 또는 h==1인 경우에는 가로 방향 변환 커널과 세로 방향 변환 커널이 설정되지 않을 수 있다. w==2 또는 w>32이면, 가로 방향 변환 커널이 DCT-II로 결정될 수 있다. h==2 또는 h>32이면, 세로 방향 변환 커널이 DCT-II로 설정될 수 있다. 위 예들에 해당하지 않는 경우에는 아래 표 4를 통해 변환 커널들이 설정될 수 있다.

표 4

[0100]

Intra mode	t_H	t_V
Planar Ang. 31,32,34,36,37	DST-VII	DST-VII
DC Ang. 33, 35	DCT-II	DCT-II
Ang. 2, 4, 6…28,30 Ang. 39,41,43…63,65	DST-VII	DCT-II
Ang. 3,5,7…27,29 Ang. 38,40,42…64,66	DCT-II	DST-VII

[0101] 표 4에서, t_H는 가로 방향 변환 커널을 나타내며, t_V는 세로 방향 변환 커널을 나타낸다.

[0102] **서브블록 변환(sub-block transform, SBT)**

[0103] SBT는 인터 예측에서 현재블록을 보다 작은 블록(서브블록)들로 분할하여 서브블록 단위에서 변환을 수행하는 기법이다. SBT의 타입정보 및 SBT의 위치정보가 영상 부호화 장치로부터 영상 복호화 장치로 시그널링된다.

[0104] SBT-V(세로 방향 분할)의 경우에는 변환블록의 가로 방향 길이(너비 또는 폭)가 현재블록의 가로 방향 길이(너비 또는 폭)의 1/2 또는 1/4와 같을 수 있다. 또한, SBT-H(가로 방향 분할)의 경우에는 변환블록의 세로 방향 길이(높이)의 1/2 또는 1/4와 같을 수 있다. 따라서, SBT에서는 2:2 분할, 1:3 및 3:1 분할이 발생할 수 있다.

[0105] SBT의 타입(타입정보)에 따라 가로 방향 변환 및 세로 방향 변환을 묵시적으로 다르게 설정할 수 있다. 예를 들어, SBT-V의 위치 0(좌측 서브블록)의 가로 방향 변환 및 세로 방향 변환 각각은 DCT-VIII 및 DST-VII일 수 있다. 만약, 어느 서브블록의 크기가 32보다 큰 경우에는 가로 방향 변환 및 세로 방향 변환 모두 DCT-II로 설정될 수 있다.

[0106] **저주파 비-분리 변환(low-frequence non-separable transform, LFNST)**

[0107] LFNST는 앞서 설명된 변환 과정을 통해 변환된 변환계수들에 대해 추가적인 변환을 수행하여 부호화 및 복호화의 효율성을 향상시키는 기법이다. 앞서 설명된 변환 과정을 1차 변환(forward primary transform)으로 보면, LFNST는 2차 변환에 해당할 수 있다.

[0108] LFNST는 영상 부호화 장치에서 1차 변환과 양자화 과정 사이에 적용되며 1차 변환된 계수들에 대해 적용될 수 있다. 또한, LFNST는 영상 복호화 장치에서 역양자화와 1차 역변환 사이에 적용되며 역양자화된 변환계수들에 적용될 수 있다.

[0109] LFNST에서는, 블록의 크기에 따라, 4x4 크기의 비분리 변환(4x4 LFNST) 또는 8x8 크기의 비분리 변환(8x8 LFNST)이 적용될 수 있다. 예를 들어, 4x4 LFNST는 블록의 가로 방향 크기와 세로 방향 크기 중에서 작은 값이 8 미만인 작은 블록에 대해 적용되며, 8x8 LFNST는 블록의 가로 방향 크기와 세로 방향 크기 중에서 작은 값이 4를 초과하는 큰 블록에 대해 적용될 수 있다.

[0110] 4x4 LFNST를 적용하기 위한 4x4 블록 X(입력 블록)를 행렬로 표현하면 수학식 2와 같다.

수학식 2

[0111]
$$X = \begin{bmatrix} X_{00} & X_{01} & X_{02} & X_{03} \\ X_{10} & X_{11} & X_{12} & X_{13} \\ X_{20} & X_{21} & X_{22} & X_{23} \\ X_{30} & X_{31} & X_{32} & X_{33} \end{bmatrix}$$

[0112] 행렬로 표현된 X를 벡터로 변환한 $\vec{X}$ 는 아래 수학식 3과 같다.

수학식 3

[0113]
$$\vec{X} = [X_{00} \quad X_{01} \quad X_{02} \quad X_{03} \quad X_{10} \quad X_{11} \quad X_{12} \quad X_{13} \quad X_{20} \quad X_{21} \quad X_{22} \quad X_{23} \quad X_{30} \quad X_{31} \quad X_{32} \quad X_{33}]^T$$

[0114] 아래 수학식 4를 통해 LFNST 변환 계수 벡터가 계산될 수 있다. $\vec{F}$ 는 LFNST 변환 계수 벡터를 나타내며, T는 16x16의 LFNST 변환 행렬(LFNST 변환 커널)을 나타낸다.

수학식 4

[0115]
$$\vec{F} = T \cdot \vec{X}$$

[0116] 16x1의 계수 벡터 $\vec{F}$ 는 4x4 블록으로 새롭게 배치되며(재구성되며), 인트라 예측 모드에 따라 가로/세로/대각선 방향의 스캔 순서를 사용하여 재구성될 수 있다.

[0117] LFNST에서는 총 4개의 변환 집합(transform set)이 존재하며, 변환 집합 각각에 대해 2개의 LFNST 변환 행렬(변환 커널)이 LFNST를 위해 사용될 수 있다. 변환 집합들 중에서 LFNST를 위해 사용될 변환 집합은 아래 표 5와 같이 인트라 예측 모드와 일대일(1:1)로 맵핑되는 사전 정의된 테이블에 따라 결정될 수 있다. 예를 들어, CCLM 모드가 사용된 경우에는 변환 집합을 지시하는 인덱스(Tr. set index)가 0으로 설정될 수 있다.

표 5

[0118]

IntraPredMode	Tr. set index
IntraPredMode < 0	1
0 <= IntraPredMode <= 1	0
2 <= IntraPredMode <= 12	1
13 <= IntraPredMode <= 23	2
24 <= IntraPredMode <= 44	3
45 <= IntraPredMode <= 55	2
56 <= IntraPredMode<= 80	1
81 <= IntraPredMode<= 83	0

[0119] LFNST는 적용 가능한 경우가 제한될 수 있다.

[0120] 예를 들어, 블록의 첫 번째 계수 하위 그룹(sub-group)을 제외한 나머지 그룹에서 모든 계수가 0인 경우에 LFNST가 적용될 수 있다. 따라서, LFNST 인덱스의 코딩은 1차 변환 계수 중에서 스캔 순서 상 마지막 0이 아닌 변환 계수(마지막 유효 계수, last significant coefficient)의 위치에 의존하게 된다.

[0121] 다른 예로, LFNST는 인트라 예측된 블록에 적용 가능하며, 루마 블록과 크로마 블록 모두에 적용 가능하다. 따라서, dual tree 구조의 경우에는, LFNST가 루마 블록과 크로마 블록에 분리되어(구분되어) 적용될 수 있도록, LFNST 인덱스들도 분리되어 시그널링될 수 있다. 다만, P(predictive)/B(bi-predictive) 프레임의 경우에는 LFNST 인덱스가 루마 블록과 크로마 블록 모두에 대해 하나만 시그널링될 수 있다.

[0122] 또 다른 예로, ISP가 적용된 블록에 대해서는 LFNST가 자동적으로 적용 불가능한 것으로 설정될 수 있으며, MIP 모드가 적용된 블록에 대해서도 LFNST가 적용 불가능한 것으로 설정될 수 있다.

[0123] 또 다른 예로, 큰 사이즈를 가지는 현재블록(예를 들어, 64x64 초과하는 블록)의 경우에는 변환블록의 크기가 분할되는 것으로 가정되므로, LFNST도 해당 큰 블록에 대해 적용되지 않을 수 있다. 또한, LFNST에서는 DCT-II 변환 커널만 적용되는 것으로 설정될 수도 있다.

[0124] LFNST의 복잡도를 개선하기 위해, 블록 전체에 대해 마지막 유효 계수를 검사하지 않고, 블록 내 일부의 영역에 대해서만 LFNST를 적용할 수도 있다. 이 경우, 마지막 유효 계수의 존재 여부를 검사하는 과정은 블록의 좌상단에 위치하는 4x4 블록에 대해서만 적용될 수 있다.

[0125] 예를 들어, 4x16 크기의 블록의 경우에, 가로 방향 길이 및 세로 방향 길이 중에서 짧은 길이가 4이고 8보다 작기 때문에, 가장 저주파에 해당하는 좌측의 4x4 블록에 대해 4x4 LFNST가 적용될 수 있다. 이를 위하여, 수학식 4에서, 4x4 블록의 변환계수들이 16x1의 벡터 형태로 재배열되며, 16x16의 LFNST 변환 커널 T가 적용되고, LFNST 계수 F가 4x4 형태(4x4 블록)로 재배열된다.

[0126] 다른 예로, 16x16 크기의 블록의 경우, 가로 방향 길이 및 세로 방향 길이 중에서 짧은 길이가 8보다 크므로,

가장 저주파에 해당하는 좌상단의 48개의 변환 계수들이 48x1의 벡터 형태로 재배열된 후에, 16x48의 LFNST 변환 커널이 적용된다. LFNST 계수 F는 4x4 형태(4x4 블록)로 재배열된다.

[0127] LFNST는 DCT-II 변환 커널을 이용한 변환/역변환에 보다 효과적인 반면에, 묵시적 MTS와 같이 DCT-II 변환 커널을 이용하지 않는 경우에는 덜 효과적일 수 있다. 따라서, 현재블록에 LFNST가 적용되거나 MIP가 적용되는 경우에는, 묵시적 MTS를 위한 가로 방향 변환 타입과 세로 방향 변환 타입이 DCT-II 변환 커널로 고정되어 설정될 수도 있다.

[0129] **실시예 1**

[0130] 실시예 1은 MTS를 효율적으로 제어하는 방법이다.

[0131] 앞서 설명된 바와 같이, 종래 방법에서, sps_mts_enabled_flag는 MTS의 활성화 여부(사용 여부)를 SPS 레벨에서 결정한다. sps_mts_enabled_flag==0인 경우에는 (MTS가 적용되지 않고) DCT-II 변환 커널이 적용된다. 이와 달리, sps_mts_enabled_flag==1인 경우에는, 아래 표 6과 같이, sps_explicit_mts_intra_enabled_flag와 sps_explicit_mts_inter_enabled_flag가 시그널링되고 복호화된다.

표 6

[0132]

seq_parameter_set_rbsp( ) {	Descriptor
.....	
sps_mts_enabled_flag	u(1)
if(sps_mts_enabled_flag) {	
sps_explicit_mts_intra_enabled_flag	u(1)
sps_explicit_mts_inter_enabled_flag	u(1)
}	
.....	
}	

[0133] 인트라 예측 모드로 예측된 블록(인트라 코딩된 블록)과 인터 예측 모드로 예측된 블록(인터 코딩된 블록)에 묵시적 MTS 또는 명시적 MTS를 적용하지 여부가 sps_explicit_mts_intra_enabled_flag와 sps_explicit_mts_inter_enabled_flag의 값에 따라 결정된다.

[0134] sps_explicit_mts_intra_enabled_flag는 MTS 인덱스(mts_idx)가 인트라 코딩 유닛 신택스에서 시그널링되는지 여부를 지시하는 신택스 요소로서, sps_explicit_mts_intra_enabled_flag==0은 인트라 코딩 블록에 대해 묵시적 MTS가 적용됨을 나타내며, sps_explicit_mts_intra_enabled_flag==1은 인트라 코딩 블록에 대해 명시적 MTS가 적용됨을 나타낸다.

[0135] sps_explicit_mts_inter_enabled_flag는 mts_idx가 인터 코딩 유닛 신택스에서 시그널링되는지 여부를 지시하는 신택스 요소로서, sps_explicit_mts_inter_enabled_flag==0이면 인터 코딩 블록에 대해 묵시적 MTS가 적용됨을 나타내며, sps_explicit_mts_inter_enabled_flag==1이면 인터 코딩 블록에 대해 명시적 MTS가 적용됨을 나타낸다.

[0136] sps_mts_enabled_flag의 값, sps_explicit_mts_intra_enabled_flag의 값 및 sps_explicit_mts_inter_enabled_flag의 값에 따른 MTS 적용 방식을 나타내면, 아래 표 7과 같다.

표 7

Tool	Enabling condition
Intra implicit MTS	sps_mts_enabled_flag == 1 sps_explicit_mts_intra_enabled_flag == 0
Intra explicit MTS	sps_mts_enabled_flag == 1 sps_explicit_mts_intra_enabled_flag == 1
Inter explicit MTS	sps_mts_enabled_flag == 1 sps_explicit_mts_inter_enabled_flag == 1

[0137]

[0138] 그러나, MTS 관련 신택스 요소들에 대한 위와 같은 시그널링 방식은 MTS를 효과적으로 제어하기 어렵다. 예를 들어, 인터 코딩 블록에 대해 명시적 MTS를 사용하고자 하여 sps_mts_enabled_flag==1로 설정한 경우에, sps_explicit_mts_inter_enabled_flag가 시그널링될 뿐만 아니라, sps_explicit_mts_intra_enabled_flag도 시그널링된다. 따라서, sps_explicit_mts_intra_enabled_flag에 대한 복호화 과정을 통해 sps_explicit_mts_intra_enabled_flag의 값을 파악하는 과정이 추가적으로 수행되므로, 부호화/복호화 과정의 효율이 저하될 수 있다.

[0139] 또한, 인터 코딩 블록에 대해 명시적 MTS를 사용하고 인트라 코딩된 모든 블록들에 대해 DCT-II 변환 커널을 사용하고자 하는 경우에, 복호화된 sps_explicit_mts_intra_enabled_flag의 값에 따라 묵시적 MTS(sps_explicit_mts_intra_enabled_flag==0) 또는 명시적 MTS(sps_explicit_mts_intra_enabled_flag==1) 과정이 수행되게 된다. 즉, 인트라 코딩 블록에 대해 DCT-II 변환 커널을 고정적으로 사용할 수 없는 문제가 발생하게 된다.

[0140] 이러한 문제를 해결하기 위해, 인트라 코딩 블록에 대해 MTS가 적용되지 않도록 sps_mts_enabled_flag==0으로 설정하게 되면, 인터 코딩 블록에 대해서도 무조건적으로 DCT-II 변환 커널을 사용할 수 밖에 없으므로(MTS disable), 같은 결과가 발생할 수 있다.

[0141] 또한, 아래 표 8과 같이, 인트라 코딩 블록에 ISP가 적용된 경우에는 해당 블록에 묵시적 MTS가 적용되며, 인트라 코딩 블록에 ISP가 적용되지 않은 경우에는 해당 블록에 묵시적 MTS 또는 명시적 MTS가 적용될 수 있다. 따라서, 인트라 예측 모드와 인터 예측 모드를 개별적으로 제어할 수 없는 종래 방법에 따르면, ISP 또는 SBT 등 다른 부호화/복호화 기술에 대해 MTS를 개별적으로 적용할 수 없는 문제도 발생하게 된다.

표 8

MTS 적용 부호화 기술	Intra Coding Block	Inter Coding Block
Implicit MTS	ISP, Nominal intra prediction	SBT
Explicit MTS	Nominal Intra prediction	Inter prediction

[0142]

[0143] 실시예 1에서는 MTS를 부호화/복호화 모드에 따라 개별적으로 제어함으로써, 위와 같은 종래 방법의 문제점을 해결하고자 한다.

[0144] 먼저, 영상 부호화 장치(엔트로피 부호화부)는 하나 이상의 인트라 MTS 신택스 요소들 및 하나 이상의 인터 MTS 신택스 요소들을 부호화하여 영상 복호화 장치로 시그널링할 수 있다. 즉, 본 발명에서는, 인트라 코딩 블록에 대한 MTS 제어하는 신택스 요소와 인터 코딩 블록에 대한 MTS를 제어하는 신택스 요소가 서로 구분되어 시그널링된다.

[0145] 인트라 MTS 신택스 요소들은 인트라 코딩 블록의 MTS를 제어하는 신택스 요소이고, 인터 MTS 신택스 요소들은 인터 코딩 블록의 MTS를 제어하는 신택스 요소이다. 인트라 MTS 신택스 요소들 및 인터 MTS 신택스 요소들은 비트스트림의 SPS 레벨에서 정의될 수 있다. 즉, 인트라 MTS 신택스 요소들 및 인터 MTS 신택스 요소들은 블록 레벨보다 상위의 레벨에서 정의될 수 있다.

[0146] 영상 부호화 장치는(엔트로피 부호화부)는 현재블록의 (양자화된) 변환계수들을 부호화하여 영상 복호화 장치로 시그널링할 수 있다.

[0147] 영상 복호화 장치(엔트로피 복호화부)는 하나 이상의 인트라 MTS 신택스 요소들 및 하나 이상의 인터 MTS 신택스 요소들을 비트스트림의 SPS 레벨로부터 복호화할 수 있다(S510).

[0148] 영상 복호화 장치(엔트로피 복호화부)는 (양자화된) 변환계수들을 비트스트림으로부터 복호화할 수 있다(S520). 또한, 영상 복호화 장치(역양자화부)는 복호화된 변환계수들을 역양자화하여 현재블록에 대한 변환계수들을 도출할 수 있다.

[0149] 영상 복호화 장치(역변환부)는 도출된 변환계수들의 역변환에 이용될 하나 이상의 변환 커널들을 결정할 수 있다(S530). 역변환에 이용될 변환 커널들은 현재블록의 예측 모드(인트라 예측 모드, 인터 예측 모드, ISP 모드, SBT 모드 등), 인트라 MTS 신택스 요소들 및 인터 MTS 신택스 요소들에 기반하여 결정될 수 있다.

[0150] 영상 복호화 장치(역변환부)는 결정된 변환 커널들을 이용하여 변환계수들을 역변환함으로써 현재블록에 대한 잔차블록(잔차샘플들 또는 잔차신호)을 유도할 수 있다(S540).

[0151] 이와 같이, 실시예 1에서는 MTS를 제어하는 신택스 요소들이 현재블록의 예측 모드에 따라 구분되어 시그널링되므로, MTS의 제어가 현재블록의 예측 모드 별로 구분되어 구현될 수 있다. 따라서, 실시예 1은 어느 하나의 예측 모드에 대한 MTS가 다른 예측 모드에 영향을 미치는 종래 방법의 문제점을 해결할 수 있다.

[0152] 한편, 인트라 MTS 신택스 요소들과 인터 MTS 신택스 요소들은 다양한 형태로 구현될 수 있다. 이하에서는, 실시예를 구분하여 두 신택스 요소들에 대한 다양한 형태에 대해 설명하도록 한다.

[0153] 실시예 1-1

[0154] 인트라 MTS 신택스 요소들은 인트라 MTS 인에이블 플래그(sps_mts_intra_enabled_flag) 및 인트라 MTS 선택 신택스 요소(sps_intra_mts_selection)를 포함하여 구성될 수 있다. 인터 MTS 신택스 요소들도 인터 MTS 인에이블 플래그(sps_mts_inter_enabled_flag) 및 인터 MTS 선택 신택스 요소(sps_inter_mts_selection)를 포함하여 구성될 수 있다.

[0155] sps_mts_intra_enabled_flag는 인트라 예측 모드의 MTS가 인에이블되는지 여부를 나타내는 신택스 요소에 해당하며, sps_mts_inter_enabled_flag는 인터 예측 모드의 MTS가 인에이블되는지 여부를 나타내는 신택스 요소에 해당한다. 영상 복호화 장치는 비트스트림의 SPS 레벨로부터 sps_mts_intra_enabled_flag 및 sps_mts_inter_enabled_flag를 복호화할 수 있다(S610).

[0156] sps_intra_mts_selection은 mts_idx가 비트스트림에 포함되는지 여부(변환 유닛 신택스에서 인트라 코딩 유닛 신택스에 포함되는지 여부)를 나타내는 신택스 요소이다. sps_inter_mts_selection는 mts_idx가 비트스트림에 포함되는지 여부(변환 유닛 신택스에서 인터 코딩 유닛 신택스에 포함되는지 여부)를 나타내는 신택스 요소이다. 즉, sps_intra_mts_selection 및 sps_inter_mts_selection은 묵시적 MTS와 명시적 MTS 중에서 어느 것이 적용되는지 여부를 나타내는 신택스 요소이다.

[0157] 영상 복호화 장치는 sps_mts_intra_enabled_flag 및 sps_mts_inter_enabled_flag의 값을 판단할 수 있다(S620, S650).

[0158] sps_mts_intra_enabled_flag==0인 경우에는 인트라 코딩된 모든 블록에 대해 DCT-II 변환 커널이 적용된다. 이와 달리, sps_mts_intra_enabled_flag==1인 경우에는 sps_intra_mts_selection이 비트스트림으로부터 복호화되고(S630), sps_intra_mts_selection==0인 경우에(S640)는 묵시적 MTS가 적용되며, sps_intra_mts_selection==1인 경우에(S640)는 mts_idx가 지시하는 변환 커널이 적용된다(명시적 MTS).

[0159] sps_mts_inter_enabled_flag==0인 경우에는 인터 코딩된 모든 블록에 대해 DCT-II 변환 커널이 적용된다. 이와 달리, sps_mts_inter_enabled_flag==1인 경우에는 sps_inter_mts_selection이 비트스트림으로부터 복호화되고(S660), sps_inter_mts_selection==0인 경우에(S670)는 묵시적 MTS가 적용되며, sps_inter_mts_selection==1인

경우에(S670)는 mts_idx가 지시하는 변환 커널이 적용된다(명시적 MTS).

[0160] 실시예 1-1에 대한 신택스 구조가 표 9에 나타나 있다.

표 9

[0161]

seq_parameter_set_rbsp( ) {
sps_mts_intra_enabled_flag
sps_mts_inter_enabled_flag
if(sps_mts_intra_enabled_flag) {
sps_intra_mts_selection
}
if(sps_mts_inter_enabled_flag) {
sps_inter_mts_selection
}
...
}

[0162] sps_intra_mts_selection 및 sps_inter_mts_selection를 사용하면, 아래 표 10과 같이 인트라 코딩 블록 및 인터 코딩 블록 각각에 대해 MTS를 구분하여 적용 가능하다.

표 10

[0163]

MTS 적용 부호화 기술	Intra	Inter
Implicit MTS	ISP, Nominal intra prediction	SBT
Explicit MTS	Nominal Intra prediction	Inter prediction

[0164] 실시예 1-2

[0165] 인트라 MTS 신택스 요소들은 인트라 MTS 선택 신택스 요소(sps_intra_mts_selection)를 포함하여 구성될 수 있다. 인터 MTS 신택스 요소들도 인터 MTS 선택 신택스 요소(sps_inter_mts_selection)를 포함하여 구성될 수 있다.

[0166] sps_intra_mts_selection은 인트라 예측 모드의 MTS가 인에이블되는지 여부와 mts_idx가 비트스트림에 포함되는지 여부(변환 유닛 신택스에서 인트라 코딩 유닛 신택스에 포함되는지 여부)를 나타내는 신택스 요소이다. sps_inter_mts_selection는 인터 예측 모드의 MTS가 인에이블되는지 여부와 MTS 인덱스가 비트스트림에 포함되는지 여부(변환 유닛 신택스에서 인터 코딩 유닛 신택스에 포함되는지 여부)를 나타내는 신택스 요소이다. 즉, sps_intra_mts_selection 및 sps_inter_mts_selection은 MTS의 인에이블 여부 및 묵시적 MTS와 명시적 MTS 중에서 어느 것이 적용되는지 여부를 하나의 신택스 요소로 나타낼 수 있다.

[0167] sps_intra_mts_selection==제1값(예를 들어, 0)은 인트라 코딩 블록에 대해 MTS가 적용되지 않음을 나타내며, sps_intra_mts_selection==제2값(예를 들어 1)은 인트라 코딩 블록에 대해 묵시적 MTS가 적용됨을 나타내고, sps_intra_mts_selection==제3값(예를 들어 2)은 인트라 코딩 블록에 대해 명시적 MTS가 적용됨을 나타낼 수 있다.

[0168] sps_inter_mts_selection==제1값(예를 들어, 0)은 인터 코딩 블록에 대해 MTS가 적용되지 않음을 나타내며, sps_inter_mts_selection==제2값(예를 들어 1)은 인터 코딩 블록에 대해 묵시적 MTS가 적용됨을 나타내고, sps_inter_mts_selection==제3값(예를 들어 2)은 인터 코딩 블록에 대해 명시적 MTS가 적용됨을 나타낼 수

있다.

[0169] 영상 복호화 장치는 sps_intra_mts_selection 및 sps_inter_mts_selection를 비트스트림의 SPS 레벨로부터 복호화하고(S710), sps_intra_mts_selection 및 sps_inter_mts_selection의 값을 판단할 수 있다(S720, S730).

[0170] sps_intra_mts_selection==0인 경우에는 인트라 코딩된 모든 블록에 대해 DCT-II 변환 커널이 적용된다. sps_intra_mts_selection==1인 경우에는 묵시적 MTS가 적용된다. 즉, sps_intra_mts_selection==0 또는 1인 경우에는 mts_idx가 시그널링 및 복호화되지 않는다. sps_intra_mts_selection==2인 경우에는 mts_idx가 지시하는 변환 커널이 적용된다(명시적 MTS).

[0171] sps_inter_mts_selection==0인 경우에는 인터 코딩된 모든 블록에 대해 DCT-II 변환 커널이 적용된다. sps_inter_mts_selection==1인 경우에는 묵시적 MTS가 적용된다. 즉, sps_inter_mts_selection==0 또는 1인 경우에는 mts_idx가 시그널링 및 복호화되지 않는다. sps_inter_mts_selection==2인 경우에는 mts_idx가 지시하는 변환 커널이 적용된다(명시적 MTS).

[0172] 실시예 1-2에 대한 신택스 구조가 표 11에 나타나 있다.

표 11

[0173]

seq_parameter_set_rbsp( ) {	Descriptor
.....	
sps_intra_mts_selection	u(1)
sps_inter_mts_selection	u(1)
if(sps_mts_enabled_flag) {	
...	
}	

[0174] 한편, 실시예 1-1과 실시예 1-2는 서로 혼용될 수 있다. 예를 들어, 인트라 MTS 신택스 요소들이 실시예 1-1과 같이 sps_mts_intra_enabled_flag 및 sps_intra_mts_selection를 포함하여 구성되고, 인터 MTS 신택스 요소들이 실시예 1-2와 같이 하나의 sps_inter_mts_selection를 포함하여 구성될 수 있다.

[0175] 다른 예로, 인트라 MTS 신택스 요소들이 실시예 1-2와 같이 하나의 sps_intra_mts_selection를 포함하여 구성되고, 인터 MTS 신택스 요소들이 실시예 1-1과 같이 sps_mts_inter_enabled_flag 및 sps_inter_mts_selection를 포함하여 구성될 수도 있다.

[0176] 실시예 1-3

[0177] 인트라 MTS 신택스 요소들은 ISP MTS 인에이블 플래그(sps_isp_non_dct2_enabled_flag)를 포함하여 구성되며, 인터 MTS 신택스 요소들은 SBT MTS 인에이블 플래그(sps_sbt_non_dct2_enabled_flag)를 포함하여 구성될 수 있다.

[0178] sps_isp_non_dct2_enabled_flag는 ISP가 적용된 인트라 코딩 블록에 DCT-II 변환 커널이 적용되는지 여부를 나타내는 신택스 요소이다. sps_isp_non_dct2_enabled_flag는 비트스트림의 SPS 레벨에서 정의되어 영상 부호화 장치로부터 영상 복호화 장치로 시그널링될 수 있다. sps_isp_non_dct2_enabled_flag는 sps_mts_enabled_flag에 독립적으로 사용될 수 있다.

[0179] sps_isp_non_dct2_enabled_flag==0은 ISP가 적용된 인트라 코딩 블록에 DCT-II 변환 커널이 적용됨을 나타내며, sps_isp_non_dct2_enabled_flag==1은 ISP가 적용된 인트라 코딩 블록에 DCT-II 변환 커널이 적용되지 않음(묵시적 MTS가 적용됨)을 나타낼 수 있다.

[0180] 영상 복호화 장치는 ISP의 활성화 여부를 지시하는 sps_isp_enabled_flag를 비트스트림으로부터 복호화하고(S810), sps_isp_enabled_flag의 값을 판단할 수 있다(S820).

[0181] sps_isp_enabled_flag==0인 경우에는 ISP 모드가 비활성화되므로, sps_isp_non_dct2_enabled_flag이 시그널링 및 복호화되지 않는다. 이와 달리, sps_isp_enabled_flag==1인 경우에, 영상 복호화 장치는 sps_isp_non_dct2_enabled_flag를 비트스트림으로부터 복호화하고(S830), sps_isp_non_dct2_enabled_flag의 값

을 판단할 수 있다(S840).

[0182] sps_isp_non_dct2_enabled_flag==0인 경우에는 ISP가 적용된 인트라 코딩 블록에 DCT-II 변환 커널이 적용되며, sps_isp_non_dct2_enabled_flag==1인 경우에는 묵시적 MTS(예를 들어, DST-VII)가 적용될 수 있다.

[0183] sps_sbt_non_dct2_enabled_flag는 SBT가 적용된 인터 코딩 블록에 DCT-II 변환 커널이 적용되는지 여부를 나타내는 신택스 요소이다. sps_sbt_non_dct2_enabled_flag는 비트스트림의 SPS 레벨에서 정의되어 영상 부호화 장치로부터 영상 복호화 장치로 시그널링될 수 있다. sps_sbt_non_dct2_enabled_flag는 sps_mts_enabled_flag에 독립적으로 사용될 수 있다.

[0184] sps_sbt_non_dct2_enabled_flag==0은 SBT가 적용된 인터 코딩 블록에 DCT-II 변환 커널이 적용됨을 나타내며, sps_sbt_non_dct2_enabled_flag==1은 SBT가 적용된 인터 코딩 블록에 DCT-II 변환 커널이 적용되지 않음(묵시적 MTS가 적용됨)을 나타낼 수 있다.

[0185] 영상 복호화 장치는 SBT의 활성화 여부를 지시하는 sps_sbt_enabled_flag를 비트스트림으로부터 복호화하고(S910), sps_sbt_enabled_flag의 값을 판단할 수 있다(S920).

[0186] sps_sbt_enabled_flag==0인 경우에는 SBT 모드가 비활성화되므로, sps_sbt_non_dct2_enabled_flag이 시그널링 및 복호화되지 않는다. 이와 달리, sps_sbt_enabled_flag==1인 경우에, 영상 복호화 장치는 sps_sbt_non_dct2_enabled_flag를 비트스트림으로부터 복호화하고(S930), sps_sbt_non_dct2_enabled_flag의 값을 판단할 수 있다(S940).

[0187] sps_sbt_non_dct2_enabled_flag==0인 경우에는 SBT가 적용된 인터 코딩 블록에 DCT-II 변환 커널이 적용되며, sps_sbt_non_dct2_enabled_flag==1인 경우에는 묵시적 MTS(예를 들어, DST-VII 또는 DCT-VIII)가 적용될 수 있다.

[0188] 실시예 1-3에 대한 신택스 구조를 나타내면 표 12와 같다.

표 12

[0189]

seq_parameter_set_rbsp( ) {	Descriptor
...	
sps_isp_enabled_flag	u(1)
if (sps_isp_enabled_flag)	
sps_isp_non_dct2_enabled_flag	u(1)
...	
sps_sbt_enabled_flag	u(1)
if(sps_sbt_enabled_flag) {	
sps_sbt_max_size_64_flag	u(1)
sps_sbt_non_dct2_enabled_flag	u(1)
}	
...	
}	

[0190] 실시예 1-4

[0191] 실시예 1-4에서는 인트라 코딩 블록 및 인터 코딩 블록 모두에 대해 DCT-II 변환 커널의 적용 여부를 지시하는 신택스 요소(sps_implicit_dct2_flag)가 더 활용될 수 있다.

[0192] 영상 부호화 장치는 인트라 코딩 블록 및 인터 코딩 블록 모두에 대해 DCT-II 변환 커널의 적용 여부를 판단하고, 판단 결과를 sps_implicit_dct2_flag의 값으로 설정할 수 있다. 또한, 영상 부호화 장치는 sps_implicit_dct2_flag를 부호화하여 영상 복호화 장치로 시그널링할 수 있다.

[0193] 영상 복호화 장치는 sps_implicit_dct2_flag를 비트스트림의 SPS 레벨로부터 복호화하고(S1010), sps_implicit_dct2_flag의 값을 판단할 수 있다(S1020). sps_implicit_dct2_flag==1이면 인트라 코딩 블록 및 인터 코딩 블록 모두에 DCT-II 변환 커널이 적용될 수 있다. 이와 달리, sps_implicit_dct2_flag==0이면, 영상

복호화 장치는 아래에서 설명되는 다른 신택스 요소들을 복호화하고(S1030, S1050), 복호화된 다른 신택스 요소들(인트라 MTS 신택스 요소들 및 인터 MTS 신택스 요소들)의 값에 따라 인트라 코딩 블록 및 인터 코딩 블록 각각에 대한 MTS를 개별적으로 제어할 수 있다.

[0194] 다른 신택스 요소들

[0195] 1. 인트라 MTS 신택스 요소들은 sps_explicit_mts_intra_enabled_flag 및 sps_implicit_intra_mts_enabled_flag를 포함하여 구성되고, 인터 MTS 신택스 요소들은 sps_explicit_mts_inter_enabled_flag를 포함하여 구성될 수 있다.

[0196] 인트라 MTS 신택스 요소들이 sps_explicit_mts_intra_enabled_flag 및 sps_implicit_intra_mts_enabled_flag를 포함하여 구성되고, 인터 MTS 신택스 요소들이 sps_explicit_mts_inter_enabled_flag를 포함하여 구성되는 예에 대한 신택스 구조가 표 13에 나타나 있다.

표 13

[0197]

...	u(1)
sps_implicit_dct2_flag	u(1)
if (sps_implicit_dct2_flag = = 0) {	
sps_explicit_mts_intra_enabled_flag	u(1)
sps_explicit_mts_inter_enabled_flag	u(1)
}	
if (sps_explicit_mts_intra_enabled_flag = = 0)	
sps_implicit_intra_mts_enabled_flag	
}	
sps_sbt_enabled_flag	u(1)

[0198] sps_explicit_mts_intra_enabled_flag는 인트라 코딩 블록에 명시적 MTS가 적용되는지 여부를 나타내는 신택스 요소로서, sps_explicit_mts_intra_enabled_flag==0은 명시적 MTS가 적용되지 않음을 나타내고, sps_explicit_mts_intra_enabled_flag==1은 명시적 MTS가 적용됨을 나타낼 수 있다.

[0199] sps_implicit_intra_mts_enabled_flag는 ISP가 적용되지 않은 인트라 코딩 블록에 묵시적 MTS가 적용되는지 여부를 나타내는 신택스 요소로서, sps_explicit_mts_intra_enabled_flag==0인 경우에(S1040) 비트스트림으로부터 복호화될 수 있다(S1050). sps_implicit_intra_mts_enabled_flag==0(S1060)은 묵시적 MTS가 적용되지 않음(DCT-II 변환 커널이 적용됨)을 나타내고, sps_implicit_intra_mts_enabled_flag==1(S1060)은 묵시적 MTS가 적용됨을 나타낼 수 있다.

[0200] sps_explicit_mts_inter_enabled_flag는 인터 코딩 블록에 명시적 MTS가 적용되는지 여부를 나타내는 신택스 요소로서, sps_explicit_mts_inter_enabled_flag==0(S1070)은 명시적 MTS가 적용되지 않음을 나타내고, sps_explicit_mts_inter_enabled_flag==1(S1070)은 명시적 MTS가 적용됨을 나타낼 수 있다.

[0201] 2. 인트라 MTS 신택스 요소들은 인트라 MTS 선택 신택스 요소(sps_intra_mts_selection)를 포함하여 구성되고, 인터 MTS 신택스 요소들도 인터 MTS 선택 신택스 요소(sps_inter_mts_selection)를 포함하여 구성될 수 있다.

[0202] sps_intra_mts_selection 및 sps_inter_mts_selection를 이용하여 인트라 코딩 블록과 인터 코딩 블록에 대한 MTS를 개별적으로 제어하는 방법은 실시예 1-2와 동일하며, 해당 방법에 대한 신택스 구조가 표 14에 나타나 있다.

표 14

[0203]

…	u(1)
sps_implicit_dct2_flag	u(1)
if (sps_implicit_dct2_flag = = 0) {	
sps_intra_mts_selection	u(v)
sps_inter_mts_selection	u(v)
}	
…	

[0204] 3. 인트라 MTS 신택스 요소들은 인트라 MTS 인에이블 플래그(sps_mts_intra_enabled_flag) 및 인트라 MTS 선택 신택스 요소(sps_intra_mts_selection)를 포함하여 구성될 수 있다. 인터 MTS 신택스 요소들도 인터 MTS 인에이블 플래그(sps_mts_inter_enabled_flag) 및 인터 MTS 선택 신택스 요소(sps_inter_mts_selection)를 포함하여 구성될 수 있다.

[0205] sps_mts_intra_enabled_flag, sps_intra_mts_selection, sps_mts_inter_enabled_flag 및 sps_inter_mts_selection를 이용하여 인트라 코딩 블록과 인터 코딩 블록에 대한 MTS를 개별적으로 제어하는 방법은 실시예 1-1과 동일하며, 해당 방법에 대한 신택스 구조가 표 15에 나타나 있다.

표 15

[0206]

seq_parameter_set_rbsp( ) {
sps_mts_intra_enabled_flag
sps_mts_inter_enabled_flag
if(sps_mts_intra_enabled_flag) {
sps_intra_mts_selection
}
if(sps_mts_inter_enabled_flag) {
sps_inter_mts_selection
}
…
}

[0207] 4. 표 14와 표 15의 예에서, 인트라 MTS 신택스 요소들에 ISP MTS 인에이블 플래그(sps_isp_non_dct2_enabled_flag)가 더 포함될 수 있으며, 인터 MTS 신택스 요소들에 SBT MTS 인에이블 플래그(sps_sbt_non_dct2_enabled_flag)가 더 포함될 수도 있다.

[0209] **실시예 2**

[0210] 실시예 2는 LFNST를 효율적으로 제어하는 방법이다.

[0211] 실시예 2-1

[0212] LFNST의 적용 여부를 블록 레벨(CU 레벨)에서 시그널링되는(코딩 유닛 신택스 구문에 정의된) LFNST 인덱스(lfnst_idx)에 의해 결정된다. lfnst_idx가 시그널링되는 신택스 구조가 표 16에 나타나 있다.

표 16

coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {
…
LfnstDcOnly = 1
LfnstZeroOutSigCoeffFlag = 1
transform_tree(x0, y0, cbWidth, cbHeight, treeType)
lfnstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC : cbWidth
lfnstHeight = (treeType == DUAL_TREE_CHROMA) ? cbHeight / SubHeightC : cbHeight

if(Min(lfnstWidth, lfnstHeight) >= 4 && sps_lfnst_enabled_flag == 1 &&
CuPredMode[chType][x0][y0] == MODE_INTRA &&
IntraSubPartitionsSplitType == ISP_NO_SPLIT &&
(!intra_mip_flag[x0][y0] || Min(lfnstWidth, lfnstHeight) >= 16) &&
tu_mts_idx[x0][y0] == 0 && Max(cbWidth, cbHeight) <= MaxTbSizeY) {

if(LfnstDcOnly == 0 && LfnstZeroOutSigCoeffFlag == 1)
lfnst_idx[x0][y0]
}
}

[0213]

[0214] lfnst_idx는 LFNST를 코딩 블록에 적용하지 여부와, 적용한다면 선택된 변환 집합 내 어떠한 변환 커널을 사용할지 여부를 지시할 수 있다. lfnst_idx==0이면 해당 코딩 블록에 LFNST가 적용되지 않으며, lfnst_idx==1이면 선택된 변환 집합 내 첫 번째 변환 커널이 LFNST를 위해 사용되고, lfnst_idx==2이면 선택된 변환 집합 내 두 번째 변환 커널이 LFNST를 위해 사용된다. LFNST를 위해 사용될 변환 세트는 코딩 블록의 인트라 예측 방향에 의해 표 17과 같이 결정될 선택될 수 있다.

표 17

Intra prediction mode (predModeIntra)	Transform set
predModeIntra < 0	1
0 <= predModeIntra <= 1	0
2 <= predModeIntra <= 12	1
13 <= predModeIntra <= 23	2
24 <= predModeIntra <= 44	3
45 <= predModeIntra <= 55	2
56 <= predModeIntra <= 80	1
81 <= predModeIntra <= 83	0

[0215]

[0216] 표 16에 나타내 바와 같이, lfnst_idx는 transform_tree 구문 이후에 시그널링 및 복호화된다. transform_tree 구문 내에서 transform_unit 구문이 호출되며, 이 transform_unit 구문 내에서 residual_coding 구문이 호출된다. 마지막 유효 계수의 위치(lastScanPos)는 표 18 및 표 19와 같이 residual_coding 구문에서 정의되므로, 결국 transform_tree 과정과 residual_coding 과정이 모두 완료된 후에야 lfnst_idx가 시그널링 및 복호화될 수 있다. 따라서, 종래 방법의 경우에는, LFNST의 적용 여부를 결정하는 과정에서 부호화 및 복호화의 지연이 발생할 수 있다.

표 18

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	Descriptor
if((tu_mts_idx[x0][y0] > 0 || (cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6)) && cIdx = = 0 && log2TbWidth > 4)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	
MaxCcbs = 2 * (1 << log2TbWidth) * (1<< log2TbHeight)	
if(tu_mts_idx[x0][y0] > 0 || (cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6)) && cIdx = = 0 && log2TbHeight > 4)	
log2ZoTbHeight = 4	
else	
log2ZoTbHeight = Min(log2TbHeight, 5)	
if(log2TbWidth > 0)	
last_sig_coeff_x_prefix	ae(v)
if(log2TbHeight > 0)	
last_sig_coeff_y_prefix	ae(v)
if(last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ae(v)
if(last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ae(v)
log2TbWidth = log2ZoTbWidth	
log2TbHeight = log2ZoTbHeight	
remBinsPass1 = ((1 << (log2TbWidth + log2TbHeight)) * 7) >> 2	
log2SbW = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)	
log2SbH = log2SbW	
if(log2TbWidth + log2TbHeight > 3) {	
if(log2TbWidth < 2) {	
log2SbW = log2TbWidth	
log2SbH = 4 − log2SbW	
} else if(log2TbHeight < 2) {	
log2SbH = log2TbHeight	
log2SbW = 4 − log2SbH	
}	
}	
numSbCoeff = 1 << (log2SbW + log2SbH)	
lastScanPos = numSbCoeff	

[0217]

표 19

lastSubBlock = (1 << (log2TbWidth + log2TbHeight − (log2SbW + log2SbH))) − 1	
do {	
if(lastScanPos == 0) {	
lastScanPos = numSbCoeff	
lastSubBlock− −	
}	
lastScanPos− −	
xS = DiagScanOrder[log2TbWidth − log2SbW][log2TbHeight − log2SbH] [lastSubBlock][0]	
yS = DiagScanOrder[log2TbWidth − log2SbW][log2TbHeight − log2SbH] [lastSubBlock][1]	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][1]	
} while((xC != LastSignificantCoeffX) || (yC != LastSignificantCoeffY))	
if(lastSubBlock == 0 && log2TbWidth >= 2 && log2TbHeight >= 2 && !transform_skip_flag[x0][y0] && lastScanPos > 0)	
LfnstDcOnly = 0	
if((lastSubBlock > 0 && log2TbWidth >= 2 && log2TbHeight >= 2) || (lastScanPos > 7 && (log2TbWidth == 2 || log2TbWidth == 3) && log2TbWidth == log2TbHeight))	
LfnstZeroOutSigCoeffFlag = 0	

[0218]

[0219] 특히, 루마 블록과 크로마 블록이 서로 트리 구조를 공유하는 경우에는, 크로마 블록의 마지막 유효 계수의 위치(lastScanPos)까지 파악한 후에 LFNST의 적용 여부를 결정할 수 있으므로, 지연 문제가 보다 더 악화될 수 있다.

[0220] 이러한 문제를 해결하기 위해, 실시예 2-1에서는 lfnst_idx를 시그널링 및 복호화하기 위한 과정을 coding_unit 구문(CU 레벨)으로부터 residual_coding 구문(TU 레벨)으로 이동시킨다. 구체적으로, 영상 부호화 장치는 마지막 유효 계수의 위치를 나타내는 lastScanPos를 계산하고, lastScanPos의 값에 따라 lfnst_idx를 부호화하여 영상 복호화 장치로 시그널링한다. 영상 복호화 장치는 lastScanPos를 계산하고, lastScanPos의 값에 따라 lfnst_idx를 비트스트림으로부터 복호화할 수 있다.

[0221] 따라서, 표 16에 나타낸 lfnst_idx를 시그널링 및 복호화하기 위한 과정들은 coding_unit 구문으로부터 삭제될 수 있으며, 아래 표 20에 나타낸 바와 같이 residual_coding 구문에서 정의될 수 있다.

표 20

[0222]

```
residual_coding( x0, y0, log2TbWidth, log2TbHeight, cIdx ) {
...
    if( Min( cbWidth, cbHeight ) >= 2 && sps_lfnst_enabled_flag == 1 &&
      CuPredMode[ chType ][ x0 ][ y0 ] == MODE_INTRA &&
      IntraSubPartitionsSplitType == ISP_NO_SPLIT &&
      ( !intra_mip_flag[ x0 ][ y0 ] || Min( cbWidth, cbHeight ) >= 16 ) &&
      Max( cbWidth, cbHeight ) <= MaxTbSizeY &&
      ( cIdx == 0 || (treeType == DUAL_TREE_CHROMA && (cIdx == 1 ||
      tu_cbf_cb[ x0 ][ y0 ] == 0) ) ) {
      if( lastScanPos > 0 && lastScanPos < 16 &&
        !( lastScanPos > 7 && (log2TbWidth == 2 || log2TbWidth == 3 )
          && log2TbWidth == log2TbHeight ) )
        lfnst_idx[ x0 ][ y0 ]
    }
...
```

[0223] 실시예 2-2

[0224] 1. 실시예 2-1에서, 마지막 유효 계수의 위치는 루마 블록에 대해서만 판단하고, 그 결과(LFNST의 적용 여부)를 모든 루마 블록과 크로마 블록에 대해 동일하게 적용할 수 있다. 이 예는 루마 블록과 크로마 블록이 서로 다른 블록 구조를 가지는 경우(dual tree)인 경우에도 동일하게 적용될 수 있다.

[0225] 2. 다른 예로, 루마 블록 및 크로마 블록 모두에 대해 마지막 유효 계수의 위치를 판단하고, 그 결과를 이용하여 결정된 LFNST의 적용 여부를 루마 블록과 크로마 블록에 대해 동일하게 적용할 수도 있다. 이 예 또한, 루마 블록과 크로마 블록이 서로 다른 블록 구조를 가지는 경우(dual tree)인 경우에도 동일하게 적용될 수 있다.

[0226] 실시예 2-3

[0227] 종래 방법에 의하면, LFNST와 ISP가 하나의 블록에 동시에 적용될 수 없으므로, ISP가 적용되는 블록에 대해서는 lfnst_idx가 시그널링 및 복호화되지 않는다. 그러나, 실시예 2-3에서는, 이러한 제한을 없애고, LFNST와 ISP가 하나의 블록에 동시에 적용될 수 있는 예를 제안한다. 실시예 2-3에 의하면, 0이 아닌 CBF를 가진 모든 TU에 동일한 LFNST가 적용되어, lfnst_idx를 전체 CU에 대해 한 번만 전송하면 되므로, 비트 효율성이 향상될 수 있다.

[0228] ISP와 LFNST가 동시에 적용될 수 있도록 하는 coding_unit 구문이 표 21에 나타나 있다.

표 21

LfnstDcOnly = 1
LfnstZeroOutSigCoeffFlag = 1
transform_tree(x0, y0, cbWidth, cbHeight, treeType)
lfnstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC : (IntraSubPartitionsSplitType == ISP_VER_SPLIT) ? cbWidth / NumIntraSubPartitions : cbWidth
lfnstHeight = (treeType == DUAL_TREE_CHROMA) ? cbHeight / SubHeightC : (IntraSubPartitionsSplitType == ISP_HOR_SPLIT) ? cbHeight / NumIntraSubPartitions : cbHeight

if(Min(lfnstWidth, lfnstHeight) >= 4 && sps_lfnst_enabled_flag == 1 &&
CuPredMode[chType][x0][y0] == MODE_INTRA &&
(!intra_mip_flag[x0][y0] || Min(lfnstWidth, lfnstHeight) >= 16) &&
tu_mts_idx[x0][y0] == 0 && Max(cbWidth, cbHeight) <= MaxTbSizeY) {

if((IntraSubPartitionsSplitType != ISP_NO_SPLIT || LfnstDcOnly) == 0 && LfnstZeroOutSigCoeffFlag == 1)
lfnst_idx[x0][y0]
}

[0229]

[0230] 이 경우에, LFNST의 적용 여부와 ISP의 적용 여부를 비트스트림 내 같은 레벨에서 한 번만 시그널링하여, ISP가 적용된 모든 블록에 대해 같은 변환 커널이 적용되도록 구성될 수도 있다. 즉, ISP가 적용되는 모든 블록에 대해서는 하나의 인트라 예측 모드가 적용되므로, 이 하나의 인트라 예측 모드에 의해 결정되는 변환 집합 내 하나의 변환 커널을 이용하도록, lfnst_idx가 시그널링될 수도 있다.

[0231] ISP가 적용될 수 있는 블록의 최소 크기는 LFNST 변환 커널의 최소 크기에 맞추어 4x4로 제한될 수도 있다. 즉, 블록의 가로 방향 길이와 세로 방향 길이 모두는 4보다 작을 수 없도록 제한될 수 있다.

[0232] 실시예 2-4

[0233] 실시예 2-4는, 실시예 2-1과 실시예 2-3을 결합하여, LFNST의 적용 여부를 결정하기 위한 과정의 지연 문제를 해소하고 LFNST와 ISP를 하나의 블록에 동시에 적용할 수 있는 방법이다.

[0234] 즉, 실시예 2-4에서는, lfnst_idx를 시그널링 및 복호화하기 위한 과정을 coding_unit 구문으로부터 residual_coding 구문으로 이동시키며, residual_coding 구문 내에서 ISP에 해당하지 않음을 나타내는 조건(IntraSubPartitionsSplitType==ISP_NO_SPLIT)을 삭제하는 방법이다.

[0235] 실시예 2-4에 대한 신택스 구조가 표 22에 나타나 있다.

표 22

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {
…
if(Min(cbWidth, cbHeight) >= 2 && sps_lfnst_enabled_flag == 1 && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] || Min(cbWidth, cbHeight) >= 16) && Max(cbWidth, cbHeight) <= MaxTbSizeY && (cIdx == 0 || (treeType == DUAL_TREE_CHROMA && (cIdx == 1 || tu_cbf_cb[x0][y0] == 0))) {
if(lastScanPos > 0 && lastScanPos < 16 && !(lastScanPos > 7 && (log2TbWidth == 2 || log2TbWidth == 3) && log2TbWidth == log2TbHeight))
lfnst_idx[x0][y0]
}
…

[0236]

[0237] 실시형태에 따라, 4x4 이상의 블록에 대해 ISP에도 실시예 2-4가 적용될 수 있도록, 표 22의 신택스 구조에 아래와 같이 'ISP가 적용될 수 있는 블록의 최소 크기를 4x4로 제한'하는 구문(Min(lfnstWidth, lfnstHeight)>=4)이 추가될 수도 있으며, 추가된 결과가 표 23에 나타나 있다.

[0238] lfnstWidth=:(IntraSubPartitionsSplitType==ISP_VER_SPLIT)?cbWidth/NumIntraSubPartitions:cbWidth

[0239] lfnstHeight=:(IntraSubPartitionsSplitType==ISP_HOR_SPLIT)?cbHeight/NumIntraSubPartitions:cbHeight

표 23

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {
…
if(Min(lfnstWidth, lfnstHeight) >= 4 && sps_lfnst_enabled_flag == 1 && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] || Min(cbWidth, cbHeight) >= 16) && Max(cbWidth, cbHeight) <= MaxTbSizeY && (cIdx == 0 || (treeType == DUAL_TREE_CHROMA && (cIdx == 1 || tu_cbf_cb[x0][y0] == 0))) {
if(lastScanPos > 0 && lastScanPos < 16 && !(lastScanPos > 7 && (log2TbWidth == 2 || log2TbWidth == 3) && log2TbWidth == log2TbHeight))
lfnst_idx[x0][y0]

[0240]

[0241] 이상의 설명은 본 실시예의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 실시예가 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 실시예의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 실시예들은 본 실시예의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 실시예의 기술 사상의 범위가 한정되는 것은 아니다. 본 실시예의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 실시예의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

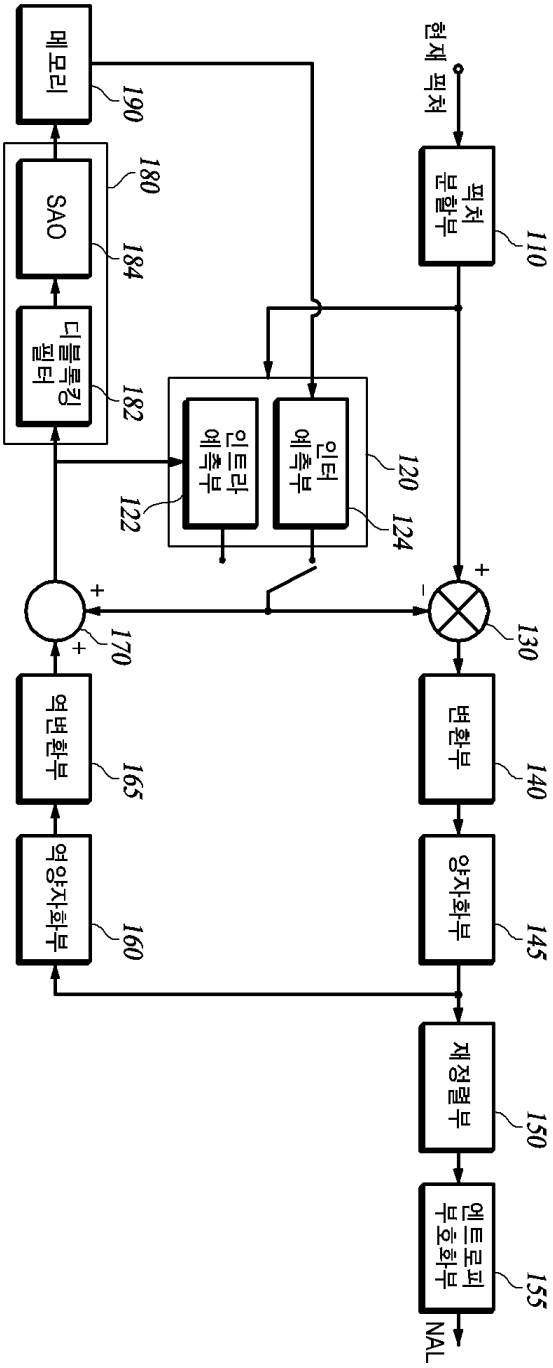
부호의 설명

[0242] 120, 440: 예측부 130: 감산기

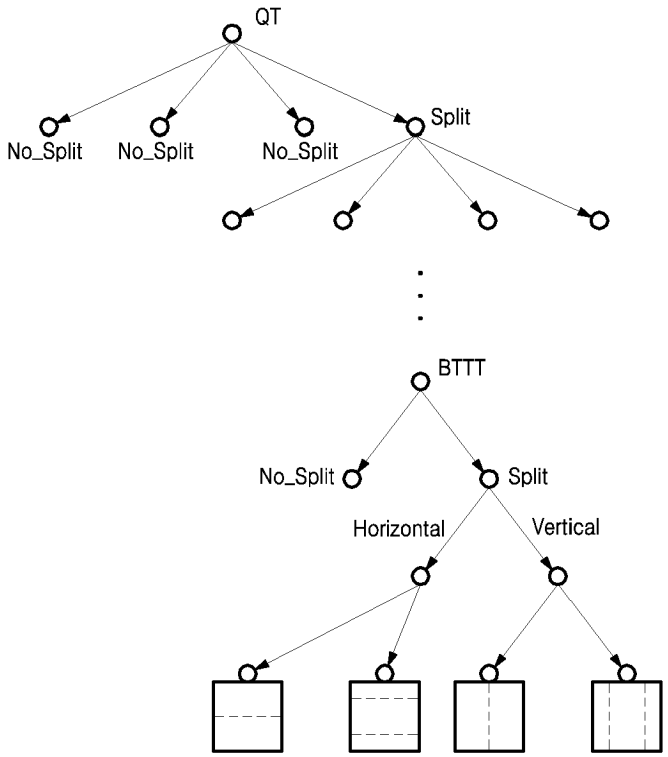
170, 450: 가산기 180, 460: 필터부

도면

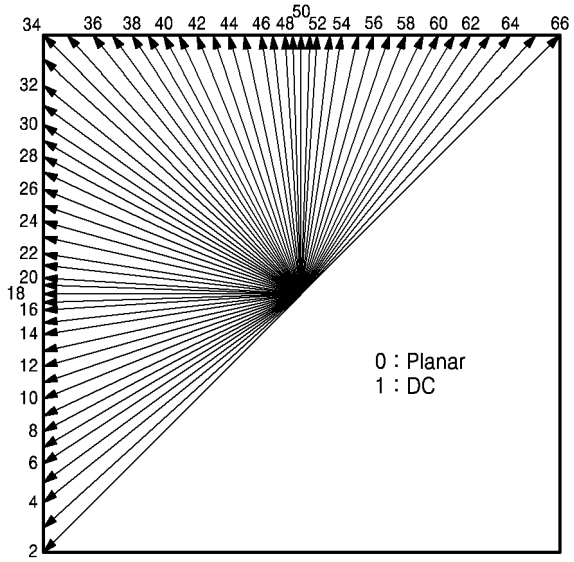
도면1


현재 픽처
픽처 분할부
110
메모리
190
SAO
180
184
디블록킹 필터
182
인트라 예측부
122
인터 예측부
120
124
+
-
130
변환부
140
양자화부
145
제정렬부
150
엔트로피 부호화부
155
NAL
+
+
170
역변환부
165
역양자화부
160

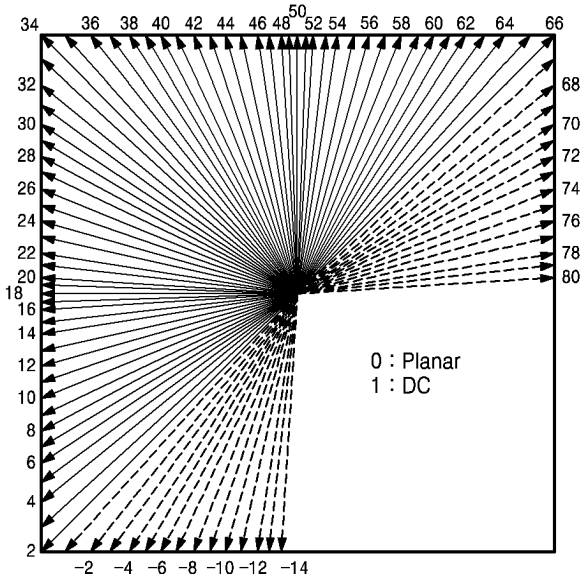
도면2


QT
No_Split
No_Split
No_Split
Split
BTTT
No_Split
Split
Horizontal
Vertical

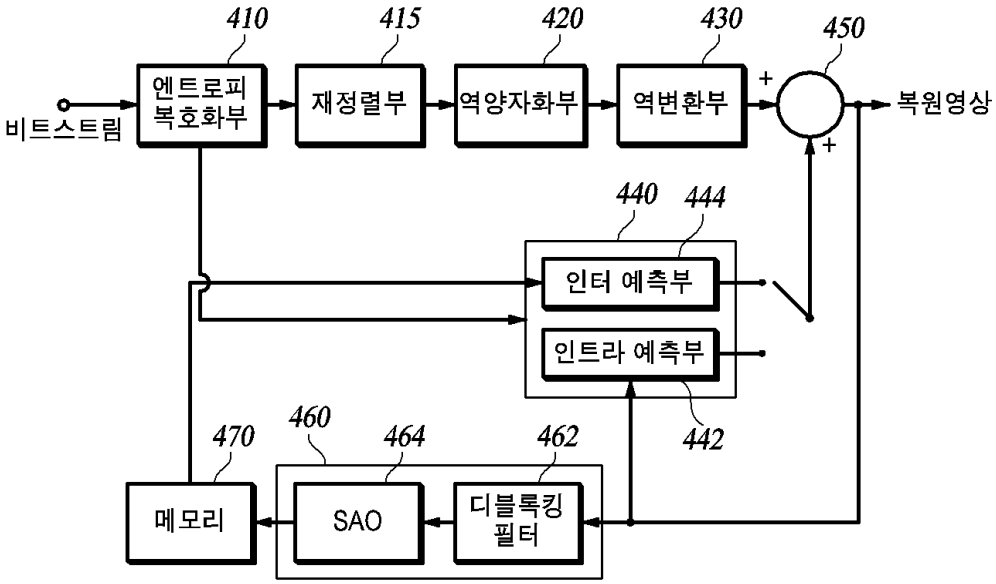
도면3a


34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
32
30
28
26
24
22
20
18
16
14
12
10
8
6
4
2
0 : Planar
1 : DC

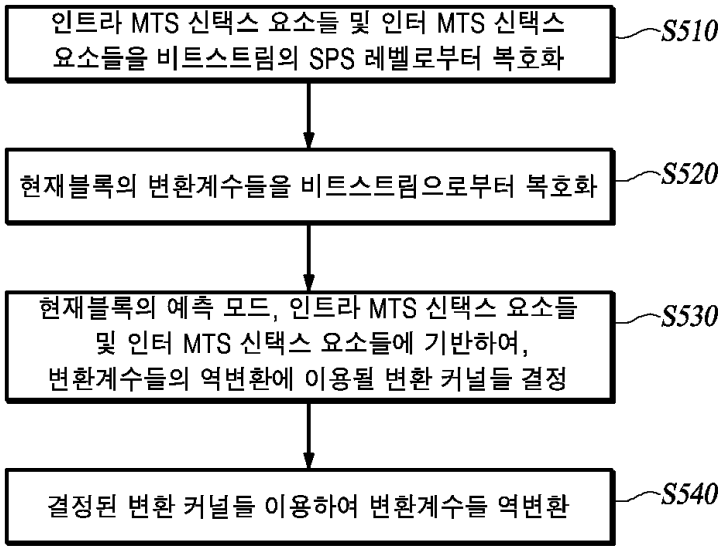
도면3b


0 : Planar
1 : DC

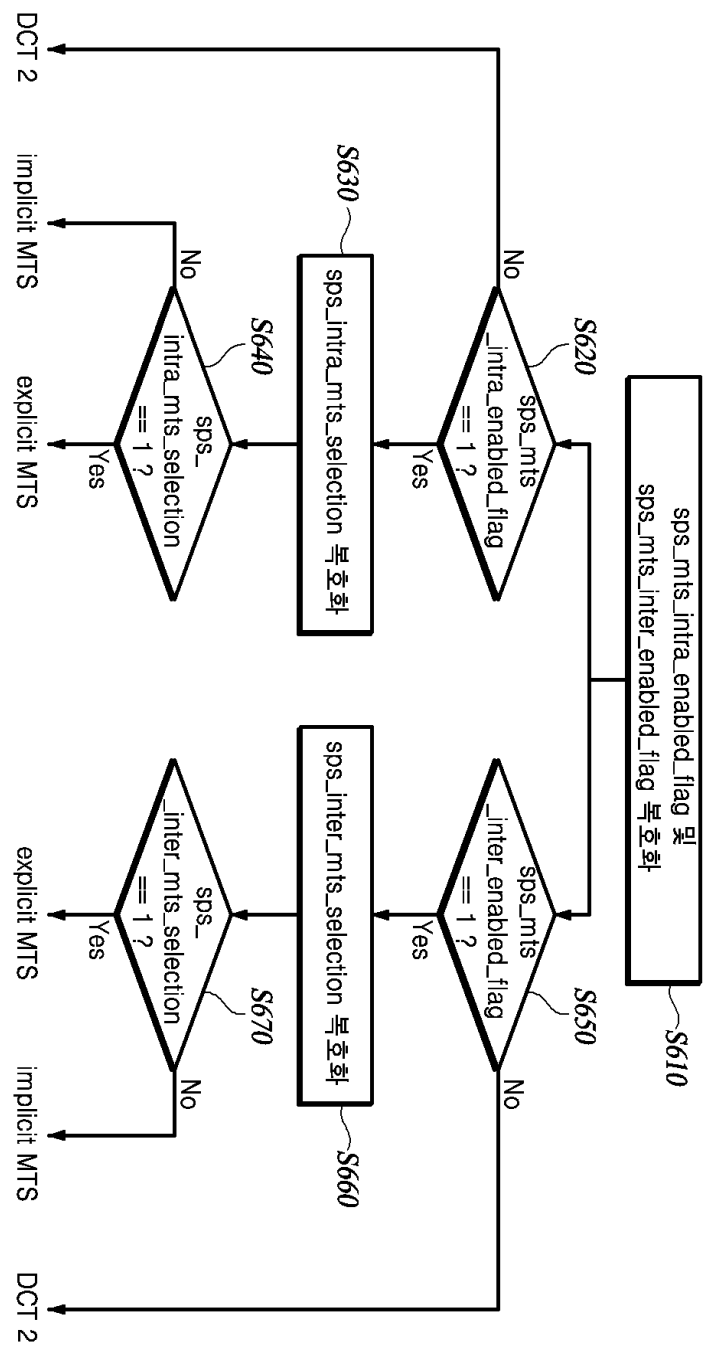
도면4


410
415
420
430
450
엔트로피 복호화부
재정렬부
역양자화부
역변환부
비트스트림
복원영상
440
444
인터 예측부
인트라 예측부
442
470
460
464
462
메모리
SAO
디블록킹 필터

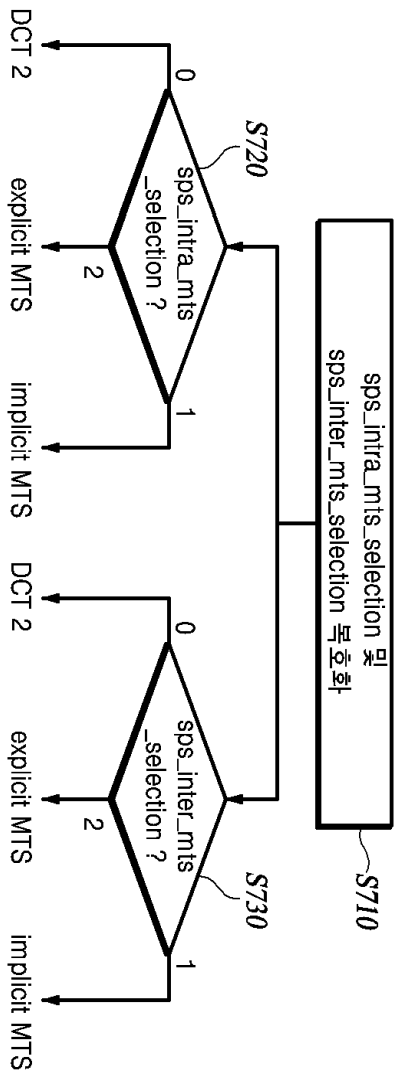
도면5


인트라 MTS 신택스 요소들 및 인터 MTS 신택스 요소들을 비트스트림의 SPS 레벨로부터 복호화
S510
현재블록의 변환계수들을 비트스트림으로부터 복호화
S520
현재블록의 예측 모드, 인트라 MTS 신택스 요소들 및 인터 MTS 신택스 요소들에 기반하여, 변환계수들의 역변환에 이용될 변환 커널들 결정
S530
결정된 변환 커널들 이용하여 변환계수들 역변환
S540

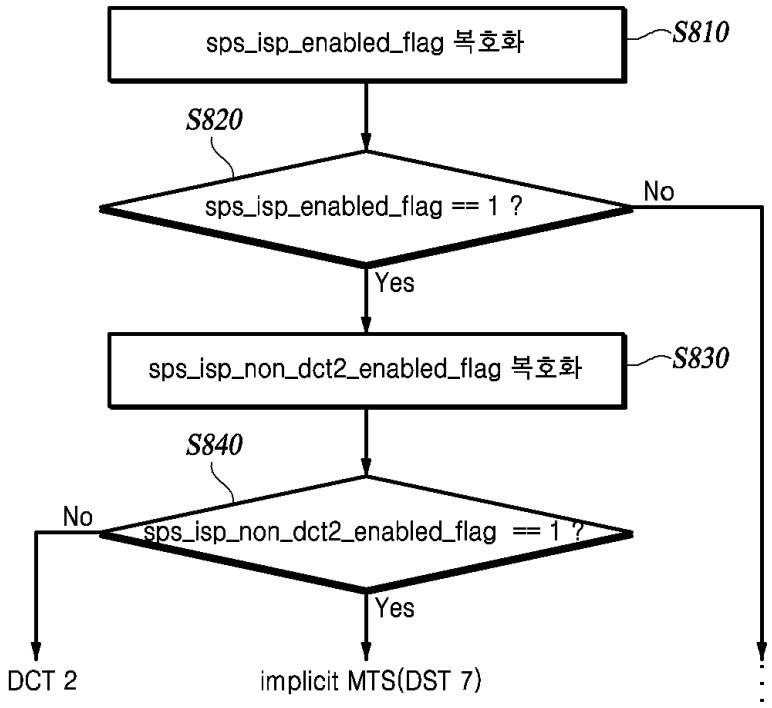
도면6


sps_mts_intra_enabled_flag 및
sps_mts_inter_enabled_flag 복호화
S610
sps_mts_intra_enabled_flag == 1 ?
S620
No
Yes
sps_intra_mts_selection 복호화
S630
sps_intra_mts_selection == 1 ?
S640
No
Yes
DCT 2
implicit MTS
explicit MTS
sps_mts_inter_enabled_flag == 1 ?
S650
No
Yes
sps_inter_mts_selection 복호화
S660
sps_inter_mts_selection == 1 ?
S670
No
Yes
DCT 2
implicit MTS
explicit MTS

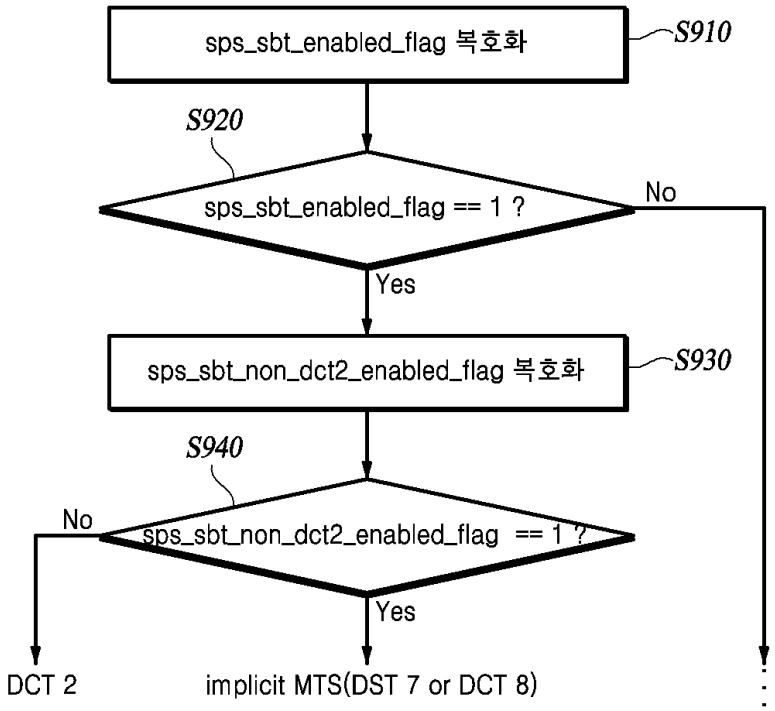
도면7


sps_intra_mts_selection 및
sps_inter_mts_selection 복호화
S710
S720
sps_intra_mts
_selection ?
0
1
2
DCT 2
explicit MTS
implicit MTS
S730
sps_inter_mts
_selection ?
0
1
2
DCT 2
explicit MTS
implicit MTS

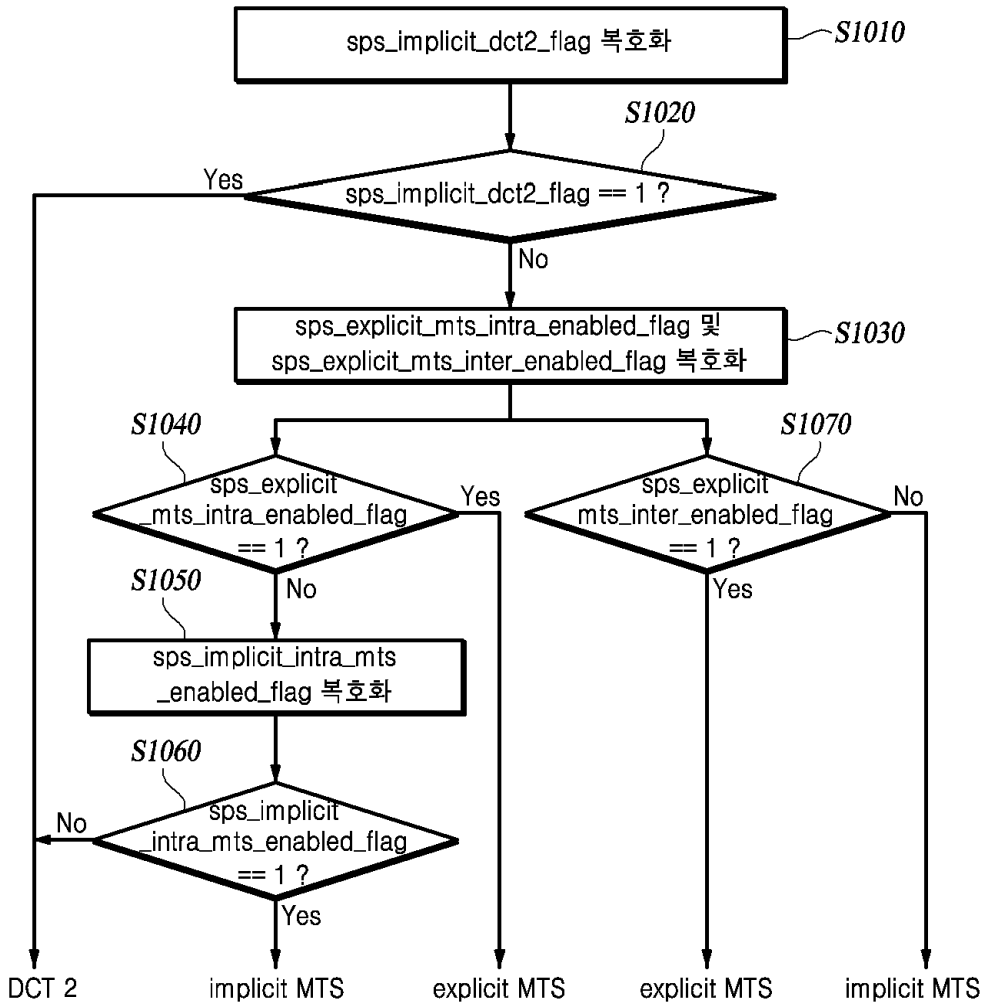
도면8


sps_isp_enabled_flag 복호화
S810
S820
sps_isp_enabled_flag == 1 ?
No
Yes
sps_isp_non_dct2_enabled_flag 복호화
S830
S840
No
sps_isp_non_dct2_enabled_flag == 1 ?
Yes
DCT 2
implicit MTS(DST 7)

도면9


sps_sbt_enabled_flag 복호화
S910
S920
sps_sbt_enabled_flag == 1 ?
No
Yes
sps_sbt_non_dct2_enabled_flag 복호화
S930
S940
No
sps_sbt_non_dct2_enabled_flag == 1 ?
Yes
DCT 2
implicit MTS(DST 7 or DCT 8)

도면10


sps_implicit_dct2_flag 복호화
S1010
S1020
sps_implicit_dct2_flag == 1 ?
Yes
No
sps_explicit_mts_intra_enabled_flag 및
sps_explicit_mts_inter_enabled_flag 복호화
S1030
S1040
sps_explicit
_mts_intra_enabled_flag
== 1 ?
Yes
No
S1050
sps_implicit_intra_mts
_enabled_flag 복호화
S1060
sps_implicit
_intra_mts_enabled_flag
== 1 ?
No
Yes
S1070
sps_explicit
mts_inter_enabled_flag
== 1 ?
No
Yes
DCT 2
implicit MTS
explicit MTS
explicit MTS
implicit MTS