



공개특허 10-2024-0127485

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)(11) 공개번호 10-2024-0127485
(43) 공개일자 2024년08월22일

(51) 국제특허분류(Int. Cl.) <i>HO4N 19/126</i> (2014.01) <i>HO4N 19/176</i> (2014.01) <i>HO4N 19/186</i> (2014.01) <i>HO4N 19/70</i> (2014.01)	(71) 출원인 톈센트 테크놀로지(센젠) 컴퍼니 리미티드 중국 518057 광동 센젠 난산 디스트릭트 미드웨스트 디스트릭트 오브 하이-테크 파크 커지중이 로드 텐센트 빌딩 35층
(52) CPC특허분류 <i>HO4N 19/126</i> (2015.01) <i>HO4N 19/176</i> (2015.01)	(72) 발명자 장 홍빈 중국 518057 광동 센젠 난산 디스트릭트 미드웨스트 디스트릭트 오브 하이-테크 파크 커지중이 로드 텐센트 빌딩 35층
(21) 출원번호 10-2024-7026780(분할)	리 상 중국 518057 광동 센젠 난산 디스트릭트 미드웨스트 디스트릭트 오브 하이-테크 파크 커지중이 로드 텐센트 빌딩 35층 (뒷면에 계속)
(22) 출원일자(국제) 2020년12월08일 심사청구일자 2024년08월08일	(74) 대리인 유미특허법인
(62) 원출원 특허 10-2022-7008146 원출원일자(국제) 2020년12월08일 심사청구일자 2022년03월11일	
(85) 번역문제출일자 2024년08월08일	
(86) 국제출원번호 PCT/CN2020/134581	
(87) 국제공개번호 WO 2021/121080 국제공개일자 2021년06월24일	
(30) 우선권주장 201911309768.6 2019년12월18일 중국(CN)	

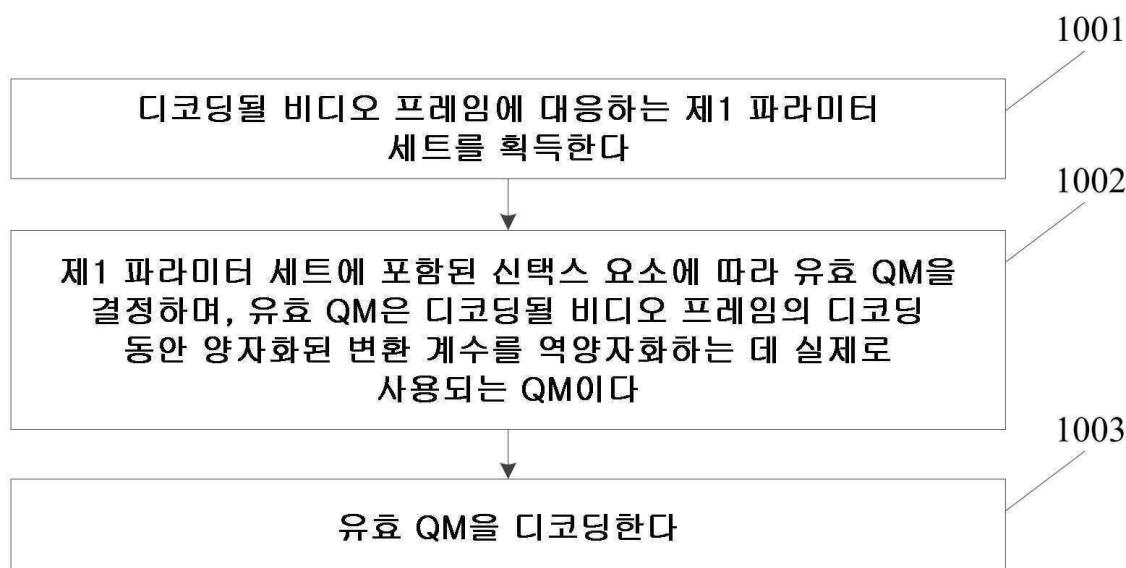
전체 청구항 수 : 총 18 항

(54) 발명의 명칭 비디오 디코딩 방법, 비디오 코딩 방법, 디바이스 및 장치, 저장 매체

(57) 요 약

본 출원은 비디오 디코딩 방법, 비디오 코딩 방법, 장치 및 장치, 및 저장 매체를 제공한다. 본 출원은 비디오 코딩 및 디코딩 처리의 기술 분야에 관한 것이다. 디코딩 방법은 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트를 획득하는 단계; 제1 파라미터 세트에 포함된 신택스 요소에 따라, 유효 QM을 결정하는 단계 - 여기서 유효 QM은 비디오 프레임의 디코딩 프로세스에서 양자화된 변환 계수에 대해 역양자화가 수행될 때 실제로 사용되는 QM임 -; 및 유효 QM을 디코딩하는 단계를 포함한다. 본 출원의 기술 솔루션을 사용함으로써, 디코더 측이 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소된다.

대 표 도 - 도10



(52) CPC특허분류

H04N 19/186 (2015.01)

H04N 19/70 (2015.01)

(72) 발명자

리 구이춘

중국 518057 광동 션젠 난산 디스트릭트 미드웨스
트 디스트릭트 오브 하이-테크 파크 커지중이 로드
텐센트 빌딩 35층

류 산

중국 518057 광동 션젠 난산 디스트릭트 미드웨스
트 디스트릭트 오브 하이-테크 파크 커지중이 로드
텐센트 빌딩 35층

명세서

청구범위

청구항 1

비디오 디코딩 방법으로서,

디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트를 획득하는 단계 - 상기 제1 파라미터 세트는 양자화 행렬(quantization matrix, QM)을 정의하는 데 사용되는 관련 신택스 요소를 포함함 - ;

상기 제1 파라미터 세트에 포함된 제1 플래그 신택스 요소에 따라 유효 QM을 결정하는 단계 - 상기 제1 플래그 신택스 요소는 크로마 QM이 디코딩될 필요가 있는지 여부를 나타내도록 구성되고, 상기 유효 QM은 상기 디코딩될 비디오 프레임의 디코딩 동안 양자화된 변환 계수를 역양자화하기 위해 실제로 사용되는 QM이며, 공통 예측 모드 및 공통 크기를 갖는 제1 크로마 QM 및 제2 크로마 QM은 상기 제1 크로마 QM 및 상기 제2 크로마 QM이 디코딩될 필요가 있는지 여부를 나타내는 상기 제1 플래그 신택스 요소를 공유함 - ; 및

상기 유효 QM을 디코딩하는 단계

를 포함하는 비디오 디코딩 방법.

청구항 2

제1항에 있어서,

상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 QM의 유효 크기 범위를 결정하는 단계; 및

상기 유효 크기 범위 내의 QM을 상기 유효 QM으로 결정하는 단계

를 더 포함하는, 비디오 디코딩 방법.

청구항 3

제2항에 있어서,

상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 QM의 유효 크기 범위를 결정하는 단계는:

상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 최소 루마 코딩 블록 크기, 루마 코딩 트리의 블록 크기, 및 최대 루마 변환 블록(TB) 크기를 결정하는 단계;

상기 최소 루마 코딩 블록 크기, 상기 루마 코딩 트리의 블록 크기 및 상기 최대 루마 TB 크기에 따라 루마 QM의 유효 크기 범위를 결정하는 단계 - 상기 루마 QM의 유효 크기 범위는 상기 루마 QM의 최소 크기 및 최대 크기를 포함함 - ; 및

상기 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하는 단계 - 상기 크로마 QM의 유효 크기 범위는 상기 크로마 QM의 최소 크기 및 최대 크기를 포함함 -

를 포함하는, 비디오 디코딩 방법.

청구항 4

제3항에 있어서,

상기 최소 루마 코딩 블록 크기, 상기 루마 코딩 트리의 블록 크기 및 상기 최대 루마 TB 크기에 따라 루마 QM의 유효 크기 범위를 결정하는 단계는:

상기 최소 루마 코딩 블록 크기에 따라 상기 루마 QM의 최소 크기를 결정하는 단계; 및

상기 루마 코딩 트리의 블록 크기 및 상기 최대 루마 TB 크기 중 더 큰 값을 상기 루마 QM의 최대 크기로 결정하는 단계

를 포함하는, 비디오 디코딩 방법.

청구항 5

제2항에 있어서,

상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 QM의 유효 크기 범위를 결정하는 단계는:

상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 루마 QM의 유효 크기 범위를 결정하는 단계 - 상기 루마 QM의 유효 크기 범위는 루마 QM의 최소 크기 및 최대 크기를 포함함 - ; 및

상기 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하는 단계 - 상기 크로마 QM의 유효 크기 범위는 크로마 QM의 최소 크기 및 최대 크기를 포함함 -

를 포함하는, 비디오 디코딩 방법.

청구항 6

제3항에 있어서,

상기 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하는 단계는:

상기 루마 QM의 최소 크기 및 상기 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 상기 크로마 QM의 최소 크기를 계산하는 단계; 및

상기 루마 QM의 최대 크기 및 상기 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 상기 크로마 QM의 최대 크기를 계산하는 단계

를 포함하는, 비디오 디코딩 방법.

청구항 7

제5항에서,

상기 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하는 단계는:

상기 루마 QM의 최소 크기 및 상기 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 상기 크로마 QM의 최소 크기를 계산하는 단계; 및

상기 루마 QM의 최대 크기 및 상기 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 상기 크로마 QM의 최대 크기를 계산하는 단계

를 포함하는, 비디오 디코딩 방법.

청구항 8

제3항에 있어서,

제1 조건 및 제2 조건 중 하나를 충족하는 제1 QM에 기초하여 상기 제1 QM을 상기 유효 QM으로 결정하는 단계를 더 포함하며,

상기 제1 조건은 `cIdx==0 && (matrixQMSize >= minQMSizeY && matrixQMSize <= maxQMSizeY)`이고, 상기 제1 QM이 루마 성분이고 루마 TB의 양자화 프로세스에 사용됨을 나타내며, 상기 제1 QM은 상기 루마 QM의 유효 크기 범위 [MinQMSizeY, MaxQMSizeY] 내에 있으며, MinQMSizeY는 상기 루마 QM의 최소 크기를 나타내고, MaxQMSizeY는 상기 루마 QM의 최대 크기를 나타내며; 그리고

상기 제2 조건은 `cIdx!=0 && (matrixQMSize >= minQMSizeUV && matrixQMSize <= maxQMSizeUV)`이고, 상기 제1 QM이 크로마 성분이고 크로마 TB의 양자화 프로세스에 사용됨을 나타내며, 상기 제1 QM은 상기 크로마 QM의 유효 크기 범위 [MinQMSizeUV, MaxQMSizeUV] 내에 있으며, MinQMSizeUV는 상기 크로마 QM의 최소 크기를 나타내

고, MaxQMSizeUV는 상기 크로마 QM의 최대 크기를 나타내는, 비디오 디코딩 방법.

청구항 9

제5항에 있어서,

제1 조건 및 제2 조건 중 하나를 충족하는 제1 QM에 기초하여 상기 제1 QM을 상기 유효 QM으로 결정하는 단계를 더 포함하며,

상기 제1 조건은 cIdx==0 && (matrixQMSize >= minQMSizeY && matrixQMSize <= maxQMSizeY)이고, 상기 제1 QM이 루마 성분이고 루마 TB의 양자화 프로세스에 사용됨을 나타내며, 상기 제1 QM은 상기 루마 QM의 유효 크기 범위 [MinQMSizeY, MaxQMSizeY] 내에 있으며, MinQMSizeY는 상기 루마 QM의 최소 크기를 나타내고, MaxQMSizeY는 상기 루마 QM의 최대 크기를 나타내며; 그리고

상기 제2 조건은 cIdx!=0 && (matrixQMSize >= minQMSizeUV && matrixQMSize <= maxQMSizeUV)이고, 상기 제1 QM이 크로마 성분이고 크로마 TB의 양자화 프로세스에 사용됨을 나타내며, 상기 제1 QM은 상기 크로마 QM의 유효 크기 범위 [MinQMSizeUV, MaxQMSizeUV] 내에 있으며, MinQMSizeUV는 상기 크로마 QM의 최소 크기를 나타내고, MaxQMSizeUV는 상기 크로마 QM의 최대 크기를 나타내는, 비디오 디코딩 방법.

청구항 10

제1항에 있어서,

상기 제1 파라미터 세트로부터 제1 QM에 대응하는 플래그 선택스 요소의 값을 판독하는 단계;

상기 제1 QM에 대응하는 플래그 선택스 요소의 값이 제1 값인 것에 기초하여, 상기 제1 QM이 상기 유효 QM인 것으로 결정하는 단계; 및

상기 제1 QM에 대응하는 플래그 선택스 요소의 값이 제2 값인 것에 기초하여, 상기 제1 QM이 상기 유효 QM이 아닌 것으로 결정하는 단계

를 더 포함하는, 비디오 디코딩 방법.

청구항 11

제1항에 있어서,

상기 제1 플래그 선택스 요소는 scaling_matrix_present_flag인, 비디오 디코딩 방법.

청구항 12

제1항에 있어서,

상기 제1 파라미터 세트는 적응 파라미터 세트(adaptive parameter set, APS)인, 비디오 디코딩 방법.

청구항 13

제1항에 있어서,

상기 유효 QM이 아닌 다른 QM의 경우, QM의 모든 요소는 디폴트 값으로 사전 정의되는, 비디오 디코딩 방법.

청구항 14

제13항에 있어서,

상기 디폴트 값은 16인, 비디오 디코딩 방법.

청구항 15

비디오 인코딩 방법으로서,

인코딩될 비디오 프레임에 대응하는 유효 양자화 매트릭스(quantization matrix, QM)를 결정하는 단계 - 상기 유효 QM은 상기 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM임 - ;

및

상기 유효 QM 및 상기 유효 QM을 결정하는 데 사용되는 선택스 요소를 인코딩하여, 제1 파라미터 세트에 대응하는 비트스트림을 생성하는 단계 - 상기 제1 파라미터 세트는 QM을 정의하는 데 사용되는 관련 선택스 요소를 포함하고, 상기 제1 파라미터 세트는 크로마 QM이 디코딩될 필요가 있는지 여부를 나타내도록 구성된 제1 플래그 선택스 요소를 포함하며, 공통 예측 모드 및 공통 크기를 갖는 제1 크로마 QM 및 제2 크로마 QM은 상기 제1 크로마 QM 및 상기 제2 크로마 QM이 디코딩될 필요가 있는지 여부를 나타내는 상기 제1 플래그 선택스 요소를 공유함 -

를 포함하는 비디오 인코딩 방법.

청구항 16

컴퓨터 디바이스로서,

프로세서 및 메모리를 포함하고, 상기 메모리는 적어도 하나의 명령, 적어도 하나의 프로그램, 코드 세트, 또는 명령 세트를 저장하며, 상기 적어도 하나의 명령, 상기 적어도 하나의 프로그램, 상기 코드 세트, 또는 상기 명령 세트는 제1항 내지 제15항 중 어느 한 항에 따른 방법을 구현하기 위해 프로세서에 의해 로드되고 실행되는, 컴퓨터 디바이스

청구항 17

컴퓨터 판독 가능형 저장 매체로서,

적어도 하나의 명령, 적어도 하나의 프로그램, 코드 세트, 또는 명령 세트를 저장하며, 상기 적어도 하나의 명령, 상기 적어도 하나의 프로그램, 상기 코드 세트, 또는 상기 명령 세트는 제1항 내지 제15항 중 어느 한 항에 따른 방법을 구현하기 위해 프로세서에 의해 로드되고 실행되는, 컴퓨터 판독 가능형 저장 매체.

청구항 18

컴퓨터 판독 가능형 저장 매체로서,

제1항 내지 제14항 중 어느 한 항에 따른 디코딩 방법에 의해 디코딩되는 비트스트림 또는 제15항에 따른 방법에 따라 생성되는 비트스트림을 저장하는, 컴퓨터 판독 가능형 저장 매체.

발명의 설명

기술 분야

[0001] 본 출원은 2019년 12월 18일에 출원되고 발명의 명칭이 "비디오 디코딩 방법 및 디바이스, 장치 및 저장 매체"인 중국 특허 출원 번호 201911309768.6에 대한 우선권을 주장하며, 이는 그 전문이 참조로 여기에 통합된다.

[0002] 본 출원의 실시예는 비디오 인코딩 및 디코딩 기술, 특히 비디오 디코딩 방법 및 장치, 비디오 인코딩 방법 및 장치, 디바이스, 및 저장 매체에 관한 것이다.

배경 기술

[0003] H.266은 H.265/HEVC(High Efficient Video Coding)를 기반으로 개선된 차세대 비디오 인코딩 기술이며 공식적으로 VVC(Versatile Video Coding)로 명명되었다. 지속적인 업데이트와 개선을 위해 JVET(Joint Video Experts Team)에서 조직하고 안내한다.

[0004] 14차 JVET 회의에서 주파수-종속 스케일링(frequency-dependent scaling)을 지원하기 위해 다음 두 가지 형태의 양자화 행렬(QM)이 VVC에서 사용될 수 있다고 결정되었다: 기본 QM 및 사용자 정의 QM. QM이 인에이블된 경우, 변환 블록(transform block, TB)의 변환 계수는 QM에 포함된 양자화 계수(즉, 정수 가중치)에 따라 개별적으로 양자화될 수 있다.

[0005] VVC에서 사용되는 양자화 행렬의 관련 디코딩 방법은 디코더 측의 계산 복잡도가 상대적으로 높다.

발명의 내용

[0006] 하나 이상의 실시예는 디코더 측의 계산 복잡도를 감소시킬 수 있는 비디오 디코딩 방법 및 장치, 비디오 인코

딩 방법 및 장치, 디바이스, 및 저장 매체를 제공한다. 기술 솔루션은 다음과 같다:

[0007] 실시예의 관점에 따라, 비디오 디코딩 방법은:

디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트를 획득하는 단계 - 상기 제1 파라미터 세트는 양자화 행렬(quantization matrix, QM)을 정의하는 데 사용되는 관련 신택스 요소를 포함함 - ;

[0009] 상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 유효 QM을 결정하는 단계 - 상기 유효 QM은 상기 디코딩될 비디오 프레임의 디코딩 동안 양자화된 변환 계수를 역양자화하기 위해 실제로 사용되는 QM임 - ; 및

[0010] 상기 유효 QM을 디코딩하는 단계

[0011] 를 포함한다.

[0012] 실시예의 관점에 따라, 비디오 인코딩 방법은:

[0013] 인코딩될 비디오 프레임에 대응하는 유효 양자화 매트릭스(effective quantization matrix, QM)를 결정하는 단계 - 상기 유효 QM은 상기 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM임 - ; 및

[0014] 상기 유효 QM 및 상기 유효 QM을 결정하는 데 사용되는 신택스 요소를 인코딩하여, QM을 정의하는 데 사용되는 관련 신택스 요소를 포함하는 제1 파라미터 세트에 대응하는 비트스트림을 생성하는 단계

[0015] 를 포함한다.

[0016] 실시예의 관점에 따라, 비디오 디코딩 장치는:

[0017] 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트를 획득하도록 구성되어 있는 파라미터 획득 모듈 - 상기 제1 파라미터 세트는 양자화 행렬(quantization matrix, QM)을 정의하는 데 사용되는 관련 신택스 요소를 포함함 - ;

[0018] 상기 제1 파라미터 세트에 포함된 신택스 요소에 따라 유효 QM을 결정하도록 구성되어 있는 QM 결정 모듈 - 상기 유효 QM은 상기 디코딩될 비디오 프레임의 디코딩 동안 양자화된 변환 계수를 역양자화하기 위해 실제로 사용되는 QM임 - ; 및

[0019] 상기 유효 QM을 디코딩하도록 구성되어 있는 QM 디코딩 모듈

[0020] 을 포함한다.

[0021] 실시예의 다른 관점에 따라, 비디오 인코딩 장치는:

[0022] 인코딩될 비디오 프레임에 대응하는 유효 QM을 결정하도록 구성되어 있는 QM 결정 모듈 - 상기 유효 QM은 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하는데 실제로 사용되는 QM임 - ; 및

[0023] 유효 QM 및 유효 QM을 결정하는 데 사용되는 신택스 요소를 인코딩하여, QM을 정의하는 데 사용되는 관련 신택스 요소를 포함하는 제1 파라미터 세트에 대응하는 비트스트림을 생성하도록 구성되어 있는 QM 인코딩 모듈

[0024] 을 포함한다.

[0025] 실시예의 관점에 따라, 컴퓨터 디바이스는 프로세서 및 메모리를 포함하고, 상기 메모리는 적어도 하나의 명령, 적어도 하나의 프로그램, 코드 세트, 또는 명령 세트를 저장하며, 상기 적어도 하나의 명령, 상기 적어도 하나의 프로그램, 상기 코드 세트, 또는 상기 명령 세트는 전술한 비디오 인코딩 방법을 구현하기 위해, 또는 전술한 비디오 디코딩 방법을 구현하기 위해 프로세서에 의해 로드되고 실행된다.

[0026] 실시예의 관점에 따라, 컴퓨터 관독 가능형 저장 매체는 적어도 하나의 명령, 적어도 하나의 프로그램, 코드 세트, 또는 명령 세트를 저장하며, 상기 적어도 하나의 명령, 상기 적어도 하나의 프로그램, 상기 코드 세트, 또는 상기 명령 세트는 전술한 비디오 인코딩 방법을 구현하기 위해, 또는 전술한 비디오 디코딩 방법을 구현하기 위해 프로세서에 의해 로드되고 실행된다.

[0027] 실시예의 관점에 따라, 컴퓨터 프로그램 제품이 제공되며, 컴퓨터 프로그램 제품은, 프로세서에 의해 실행될 때, 전술한 비디오 인코딩 방법을 구현하도록, 또는 전술한 비디오 디코딩 방법을 구현하도록 구성된다.

[0028] 본 출원의 실시예에서 제공되는 기술적 솔루션은 다음의 이로운 효과를 포함할 수 있다:

[0029] 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트가 획득되고; 유효 QM은 제1 파라미터 세트에 포함된 신택스 요소에 따라 결정되며, 유효 QM은 인코딩을 통해 디코딩될 비디오 프레임의 생성 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이고; 그런 다음 유효 QM이 디코딩된다. 이러한 방식으로 디코더 측에서는 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소한다.

도면의 간단한 설명

- [0030] 도 1은 일 실시예에 따른 비디오 인코딩의 개략도이다.
 도 2는 일 실시예에 따른 통신 시스템의 간략화된 블록도이다.
 도 3은 일 실시예에 따른 스트리밍 전송 환경에서 비디오 인코더 및 비디오 디코더의 배치 방식의 개략도이다.
 도 4는 일 실시예에 따른 인터-프레임 예측 모드에서의 인코딩의 개략도이다.
 도 5는 일 실시예에 따른 인트라-프레임 예측 모드에서의 인코딩의 개략도이다.
 도 6은 일 실시예에 따른 비디오 인코더의 기능적 모듈의 개략도이다.
 도 7은 일 실시예에 따른 비디오 디코더의 기능적 모듈들의 개략도이다.
 도 8은 일 실시예에 따른 다운샘플링 및 복사를 통해 QM을 생성하는 개략도이다.
 도 9는 일 실시예에 따른 대각 스캐닝 시퀀스의 개략도이다.
 도 10은 일 실시예에 따른 비디오 디코딩 방법의 흐름도이다.
 도 11은 일 실시예에 따른 비디오 인코딩 방법의 흐름도이다.
 도 12는 일 실시예에 따른 비디오 디코딩 장치의 블록도이다.
 도 13은 다른 실시예에 따른 비디오 디코딩 장치의 블록도이다.
 도 14는 일 실시예에 따른 비디오 인코딩 장치의 블록도이다.
 도 15는 일 실시예에 따른 컴퓨터 장치의 구조적 블록도이다.

발명을 실시하기 위한 구체적인 내용

- [0031] 본 출원의 목적, 기술적 솔루션 및 이점을 보다 명확하게 하기 위해, 다음은 첨부된 도면을 참조하여 본 출원의 구현을 상세히 설명한다.
- [0032] 도 1을 참조하면, 현재 블록(101)은 움직임 탐색 동안 인코더에 의해 발견된 샘플을 포함하고, 샘플은 동일한 크기를 갖고 공간 오프셋을 생성한 이전 블록에 따라 예측될 수 있다. 또한, 모션 벡터(motion vector, MV)는 직접 인코딩되거나 보다는 하나 이상의 참조 픽처와 연관된 메타데이터로부터 출력될 수 있다. 예를 들어, A0, A1, B0, B1 및 B2(각각 102 내지 106에 대응)의 5개의 주변 샘플 중 어느 하나와 연관된 MV는 (디코딩 순서에 따라) 가장 가까운 참조 픽처의 메타데이터에서 MV를 내보내는 데 사용된다.
- [0033] 도 2는 일 실시예에 따른 통신 시스템의 간략화된 블록도이다. 통신 시스템(200)은 복수의 장치를 포함한다. 디바이스들은, 예를 들어, 네트워크(250)를 이용하여 서로 통신할 수 있다. 예를 들어, 통신 시스템(200)은 네트워크(250)를 이용하여 서로 연결된 제1 디바이스(210) 및 제2 디바이스(220)를 포함한다. 도 2에 도시된 일 실시예에서, 제1 디바이스(210) 및 제2 디바이스(220)는 단방향 데이터 전송을 수행한다. 예를 들어, 제1 디바이스(210)는 제1 디바이스(210)에 의해 활성화된 비디오 픽처와 같은 비디오 데이터를 인코딩하여 네트워크(250)를 이용하여 비디오 데이터를 제2 디바이스(220)로 전송할 수 있다. 인코딩된 비디오 데이터는 하나 이상의 인코딩된 비디오 코드 스트림의 형태로 전송된다. 제2 디바이스(220)는 네트워크(250)로부터 인코딩된 비디오 데이터를 수신하고, 인코딩된 비디오 데이터를 디코딩하여 비디오 데이터를 재구성하고, 재구성된 비디오 데이터에 따라 비디오 픽처를 표시할 수 있다. 예를 들어, 단방향 데이터 전송은 미디어 서비스 애플리케이션에서 사용될 수 있다.
- [0034] 다른 일 실시예에서, 통신 시스템(200)은 인코딩된 비디오 데이터의 양방향 전송을 수행하는 제3 디바이스(230) 및 제4 디바이스(240)를 포함한다. 양방향 전송은 예를 들어 화상 회의 중에 발생할 수 있다. 양방향 데이터 전송의 경우, 제3 디바이스(230) 및 제4 디바이스(240) 각각은 비디오 데이터(예를 들어, 디바이스가 캡처한 비디오

픽처 스트림)를 인코딩하여 네트워크(250)를 이용함으로써 제3 디바이스(230) 및 제4 디바이스(240) 중 다른 디바이스로 전송할 수 있다. 제3 디바이스(230) 및 제4 디바이스(240) 각각은 제3 디바이스(230) 및 제4 디바이스(240) 중 다른 하나가 전송한 인코딩된 비디오 데이터를 더 수신할 수 있고, 인코딩된 비디오 데이터를 디코딩 할 수 있다. 비디오 데이터를 재구성하고, 재구성된 비디오 데이터에 따라 액세스 가능한 디스플레이 장치에 비디오 픽처를 표시할 수 있다.

[0035] 도 2에 도시된 바와 같이, 제1 디바이스(210), 제2 디바이스(220), 제3 디바이스(230), 제4 디바이스(240)는 서버, PC, 스마트폰 등의 컴퓨터 기기일 수 있다. 그렇지만, 본 출원에 개시된 원리는 여기에 제한되지 않는다. 실시예는 PC, 모바일 폰, 태블릿 컴퓨터, 미디어 플레이어 및/또는 전용 화상 회의 장치에 적용 가능하다. 네트워크(250)는 제1 디바이스(210), 제2 디바이스(220), 제3 디바이스(230) 및 제4 디바이스(240) 사이에서 인코딩된 비디오 데이터를 전송하는 임의의 수량의 네트워크를 나타내며, 예를 들어, 유선 및/또는 무선 통신 네트워크를 포함한다. 통신 네트워크(250)는 회선 교환 및/또는 패킷 교환 채널에서 데이터를 교환할 수 있다. 네트워크는 통신 네트워크, 근거리 통신망, 광역 통신망, 및/또는 인터넷을 포함할 수 있다. 본 출원의 목적을 위해, 아래에 설명되지 않는 한, 네트워크(250)의 아키텍처 및 토폴로지는 본 출원에 개시된 동작에 중요하지 않을 수 있다.

[0036] 일 실시예에서, 도 3은 스트리밍 전송 환경에서 비디오 인코더 및 비디오 디코더의 배치 방식을 나타낸다. 실시예는 이에 제한되지 않으며, 비디오 회의 애플리케이션 및 디지털 TV(텔레비전)와 같은 다른 비디오 지원 애플리케이션에 동일하게 적용되어 비디오 등을 컴팩트 디스크(CD), 디지털 다목적 디스크(DVD), 메모리 스틱 등의 디지털 미디어에 저장 및 압축할 수 있다.

[0037] 스트리밍 전송 시스템은 획득 서브시스템(313)을 포함할 수 있다. 획득 서브시스템은 디지털 카메라와 같은 비디오 소스(301)를 포함할 수 있다. 비디오 소스는 비압축 비디오 픽처 스트림(302)을 생성한다. 일 실시예에서, 비디오 픽처 스트림(302)은 디지털 카메라에 의해 캡처된 샘플을 포함한다. 인코딩된 비디오 데이터(304)(또는 인코딩된 비디오 비트스트림)와 비교하여, 비디오 픽처 스트림(302)은 큰 데이터 볼륨을 갖는 비디오 픽처 스트림을 강조하기 위해 굵은 선으로 기술된다. 비디오 픽처 스트림(302)은 전자 디바이스(320)에 의해 처리될 수 있다. 전자 디바이스(320)는 비디오 소스(301)에 연결된 비디오 인코더(303)를 포함한다. 비디오 인코더(303)는 하드웨어, 소프트웨어, 또는 구현하기 위한 하드웨어와 소프트웨어의 조합을 포함할 수 있다. 아래에 더 자세히 설명된 대로 공개된 주제의 측면을 수행한다. 비디오 픽처 스트림(302)과 비교하여, 인코딩된 비디오 데이터(304)(또는 인코딩된 비디오 비트스트림(304))는 비교적 작은 데이터 볼륨으로 인코딩된 비디오 데이터(304)(또는 인코딩된 비디오 비트스트림(304))를 강조하기 위해 가는 선으로 기술되며, 이는 향후 사용을 위해 스트리밍 전송 서버(305)에 저장된다. 도 3의 클라이언트 서브시스템(306) 및 클라이언트 서브시스템(308)과 같은 하나 이상의 스트리밍 전송 클라이언트 서브시스템은 스트리밍 전송 서버(305)에 액세스하여 인코딩된 비디오 데이터(304)의 사본(307) 및 사본(309)을 검색할 수 있다. 클라이언트 서브시스템(306)은 예를 들어, 전자 디바이스(330)의 비디오 디코더(310)를 포함할 수 있다. 인코딩된 비디오 데이터의 인코딩 사본(307)을 생성하고 디스플레이(312)(예를 들어, 디스플레이 스크린) 또는 다른 프레젠테이션 장치(도시되지 않음) 상에 제시될 수 있는 출력 비디오 픽처 스트림(311)을 생성한다. 일부 스트리밍 전송 시스템에서, 인코딩된 비디오 데이터(304), 비디오 데이터(307), 및 비디오 데이터(309)(예를 들어, 비디오 비트스트림)는 일부 비디오 인코딩/압축 표준에 따라 인코딩될 수 있다.

[0038] 전자 디바이스(320) 및 전자 디바이스(330)는 다른 구성요소(도시되지 않음)를 포함할 수 있다. 예를 들어, 전자 디바이스(320)는 비디오 디코더(도시되지 않음)를 포함할 수 있고, 전자 디바이스(330)는 비디오 인코더(도시되지 않음)를 더 포함할 수 있다. 비디오 디코더는 수신된 인코딩된 비디오 데이터를 디코딩하도록 구성되며; 비디오 인코더는 비디오 데이터를 인코딩하도록 구성된다.

[0039] 비디오 프레임의 이미지 블록이 인코딩될 때, 하나 이상의 인코딩된 참조 블록에 기초하여 예측 블록을 생성하기 위해 인터-프레임 예측 모드 또는 인트라-프레임 예측 모드가 사용될 수 있다. 예측 블록은 원래 블록의 추정된 베전일 수 있다. 예측 블록에서 원본 블록을 빼서 잔여 블록을 생성할 수 있으며, 그 반대의 경우도 마찬 가지이다. 잔여 블록은 예측된 잔여(또는 예측된 오차로 지칭됨)를 나타내기 위해 사용될 수 있다. 예측된 잔여를 나타내는 데 필요한 데이터 볼륨은 일반적으로 원래 블록을 나타내는 데 필요한 데이터 볼륨보다 작을 수 있으므로, 잔여 블록은 비교적 높은 압축 비율을 달성하기 위해 인코딩될 수 있다. 예를 들어, 도 4에 도시된 바와 같이, 인터-프레임 예측 모드의 경우, 인코딩된 참조 블록(41) 및 인코딩된 블록(42)은 2개의 상이한 비디오 프레임에 위치된다. 도 5에 도시된 바와 같이, 인트라-프레임 예측 모드의 경우, 인코딩 참조 블록(51)과 인코

당 대상 블록(52)은 동일한 비디오 프레임에 위치한다.

[0040] 이어서, 공간 영역에서 잔여 블록의 잔여 값은 주파수 도메인에서 변환 계수로 변환될 수 있다. 변환은 이산 코사인 변환(Discrete Cosine Transform, DCT)과 같은 2차원 변환을 통해 구현될 수 있다. 변환 행렬에서 낮은 인덱스 변환 계수(예를 들어, 왼쪽 상단 영역에 위치)는 큰 공간적 특징에 대응할 수 있으며 상대적으로 큰 크기를 가질 수 있다. 하이-인덱스 변환 계수(예를 들어, 우측 하단 영역에 위치)는 작은 공간적 특징에 대응할 수 있고 상대적으로 작은 크기를 가질 수 있다. 또한, 변환 행렬에 양자화 계수를 포함하는 QM을 적용하여 모든 변환 계수를 양자화된 변환 계수로 양자화할 수 있다. 양자화 결과는 변환 계수의 스케일 또는 크기가 감소될 수 있다는 것이다. 일부 하이-인덱스 변환 계수는 0으로 감소될 수 있으며, 그런 다음 후속 스캐닝 및 인코딩 작업에서 건너뛸 수 있다.

[0041] 도 6은 변환 모듈(62), 양자화 모듈(64) 및 엔트로피 인코딩 모듈(66)을 포함하는 비디오 인코더(60)의 일부를 도시한다. 도 6에 도시되지는 않았지만, 비디오 인코더(60)는 대안적으로 예측 모듈, 역양자화 모듈, 및 재구성 모듈과 같은 다른 모듈들을 포함할 수 있다는 것을 이해해야 한다. 동작 동안, 비디오 인코더(60)는 비디오 프레임을 획득할 수 있다. 비디오 프레임은 복수의 이미지 블록을 포함할 수 있다. 여기서는 편의상 하나의 이미지 블록을 인코딩하는 것을 예로 들어 설명할 수 있다. 이미지 블록을 인코딩하기 위해서는 먼저 이미지 블록의 추정으로서 예측 블록이 생성될 수 있다. 상기를 참조하면, 예측 블록은 인터-프레임 예측 모드 또는 인트라-프레임 예측 모드에서 예측 모듈에 의해 생성될 수 있다. 이후, 이미지 블록과 예측 블록의 차이를 계산하여 잔여 블록을 생성할 수 있다. 잔여 블록은 변환 모듈(62)에 의해 변환 계수로 변환될 수 있다. 변환 프로세스에서 공간 영역의 잔여 값은 큰 특징과 작은 특징을 포함하고, 주파수 도메인에서 변환 계수로 변환된다. 주파수 도메인은 고주파수 대역과 저주파 대역을 포함한다. 후속적으로 양자화 모듈(64)은 QM을 사용하여 변환 계수를 양자화함으로써 양자화된 변환 계수를 생성할 수 있다. 또한, 양자화된 변환 계수는 엔트로피 인코딩 모듈(66)에 의해 인코딩될 수 있고, 최종적으로 비디오 인코더(60)로부터 비트스트림의 일부로서 전송된다.

[0042] 도 7은 엔트로피 디코딩 모듈(72), 역양자화 모듈(74) 및 역변환 모듈(76)을 포함하는 비디오 디코더(70)의 일부를 나타낸다. 도 7에 도시되지는 않았지만, 비디오 디코더(70)는 대안적으로 예측 모듈, 변환 모듈, 및 양자화 모듈과 같은 다른 모듈들을 포함할 수 있다는 것을 이해해야 한다. 동작 동안, 비디오 디코더(70)는 비디오 인코더(60)로부터 출력된 비트스트림을 수신하고, 인터-프레임 예측 모드 또는 인트라-프레임 예측 모드에서 비트스트림을 디코딩하고, 재구성된 비디오 프레임을 출력할 수 있다. 엔트로피 디코더(72)는 입력된 비트스트림에 대해 엔트로피 디코딩을 수행하여 양자화된 변환 계수를 생성할 수 있다. 역양자화 모듈(74)은 QM에 기초하여 양자화된 변환 계수를 역양자화함으로써 역양자화된 변환 계수를 획득할 수 있다. 역변환 모듈(76)은 역양자화된 변환 계수를 역변환하여 재구성된 잔여 블록을 생성한다. 다음으로, 재구성된 잔여 블록과 예측 블록에 따라 재구성 이미지 블록이 생성된다.

[0043] 이상에서, QM은 비디오 인코딩 및 디코딩 프로세스에서 필수적인 부분임을 알 수 있다. 변환 계수의 정보가 얼마나 유지되거나 필터링되는지는 QM의 구성에 따라 달라질 수 있으며 QM은 인코딩 성능 및 인코딩 품질에 영향을 미칠 수 있다. 실제로 QM은 인코더와 디코더 모두에 필요하다. 또한, 이미지를 정확하게 디코딩하기 위해서는 QM에서 양자화 계수에 대한 정보를 인코더에서 인코딩해야 하며, 그 정보는 인코더에서 디코더로 전달된다. 비디오 인코딩 및 디코딩 기술 및 표준에서 QM은 때때로 스케일링 매트릭스 또는 가중치 매트릭스로 지칭될 수 있다. 따라서, 본 명세서에서 사용되는 "QM"이라는 용어는 양자화 행렬, 스케일링 행렬, 가중치 행렬 및 기타 등가적인 용어를 포괄하는 총칭일 수 있다.

[0044] 다음은 본 개시의 실시예와 관련된 몇몇 개념을 설명한다.

1. 양자화 행렬

[0046] 최신 버전의 VVC 테스트 모델(VTM)(즉, VTM7)에서는 정사각형 및 비정사각 TB가 허용되므로 상대적으로 많은 QM이 존재한다. QM 시그널링의 비트 수와 메모리 요구 사항을 줄이기 위해 VVC의 비정방형 TB 및 대형 정방형 TB에 대해 업샘플링 및 복사 설계가 채택되었다.

[0047] 비정방형 TB는 VVC 비트스트림에 존재하지 않으며, 디코더 측에서 대응하는 정방형 QM을 복사함으로써 획득된다. 보다 구체적으로, 32×32 QM의 0번째 행, 8번째 행, 16번째 행, 24번째 행을 복사하여 32×4 QM을 얻는다. 도 8에 도시된 바와 같이, 32×32 QM은 32×4 QM을 얻기 위해 다운샘플링되며, 여기서 사선으로 채워진 0번째 행, 8번째 행, 16번째 행 및 24번째 행은 32×32 QM에서 32×4 QM으로 복사된다.

[0048] 정사각형 TB의 크기가 8×8 보다 큰 경우, VTM7에서 대응하는 QM 크기는 8×8 로 제한된다. 업샘플링 방법은 8×8

QM에 사용되어 16×16 QM, 32×32 QM 및 64×64 QM을 생성한다. 보다 구체적으로, 16×16 크기의 QM을 생성하기 위해 16×16 크기의 QM에 대응하는 8×8 크기의 QM의 각 요소가 업샘플링되어 2×2 영역에 복사되고; 32×32 크기의 QM을 생성하기 위해 32×32 크기의 QM에 대응하는 8×8 크기의 QM의 각 요소가 업샘플링되어 4×4 영역에 복사된다.

- [0049] 최대 28개의 QM이 VTM7에서 인코딩될 필요가 있다. 표 1에서 QM의 식별자(id) 변수는 각각 표 2 및 표 3에 명시된 변수 sizeId 및 변수 matrixId에 따라 결정된다. sizeId는 QM의 크기를 나타내고, matrixId는 예측 모드(predMode) 및 색상 성분(cIdx)에 기반한 QM 유형 id이다.

표 1

id	0	1	2	3	4	5	6	7	8	9
sizeId	1	1	2	2	2	2	2	2	3	3
matrixId	4	5	0	1	2	3	4	5	0	1
id	10	11	12	13	14	15	16	17	18	19
sizeId	3	3	3	3	4	4	4	4	4	4
matrixId	2	3	4	5	0	1	2	3	4	5
id	20	21	22	23	24	25	26	27		
sizeId	5	5	5	5	5	5	6	6		
matrixId	0	1	2	3	4	5	0	3		

- [0051] 표 1--sizeId 및 matrixId에 따라 QM의 id를 결정한다.

표 2

Size of QM	sizeId	Decoded QM Parameter()	
2x2	1	2x2	--
4x4	2	4x4	--
8x8	3	8x8	--
16x16	4	8x8	1 DC
32x32	5	8x8	1 DC
64x64	6	8x8	1 DC

- [0053] 표 2--sizeId의 정의 표준

표 3

sizeId	predMode	cIdx	matrixId
2, 3, 4, 5, 6	MODE_INTRA	0 (Y)	0
2, 3, 4, 5, 6	MODE_INTRA	1 (Cb)	1
2, 3, 4, 5, 6	MODE_INTRA	2 (Cr)	2
2, 3, 4, 5, 6	MODE_INTER, MODE_IBC	0 (Y)	3
1, 2, 3, 4, 5, 6	MODE_INTER, MODE_IBC	1 (Cb)	4
1, 2, 3, 4, 5, 6	MODE_INTER, MODE_IBC	2 (Cr)	5

- [0055] 표 3--matrixId의 정의 표준

- [0056] 표 2에서 sizeId가 3보다 큰 경우 직류(DC) 계수가 있고, DC 계수는 QM에서 (0, 0) 위치의 원소 값이다. VVC에서 DC 값이 0이면 기본 QM이 QM으로 사용되지만 QM은 계속 전송된다. 주된 이유는 인코딩되지 않은 QM의 경우 QM을 참조할 수 있기 때문이다. DC 값이 0이 아닌 경우, 사용자 정의 QM을 QM으로 사용하고, QM을 아래에서 설명하는 인코딩 방식으로 인코딩하여 전송한다.

- [0057] 표 3에서 MODE_INTRA는 인트라-프레임 예측 모드를 나타내고, MODE_INTER는 인터-프레임 예측 모드를 나타내고,

MODE_IBC는 인트라 블록 사본(intra block copy, IBC) 예측 모드를 나타내고, Y는 루마를 나타내고, Cb 및 Cr은 크로마를 나타낸다.

[0058] 2. QM 인코딩 방식

비트 오버헤드를 줄이기 위해 VTM7에서 인트라-프레임 예측 인코딩 및 인터-프레임 예측 인코딩을 통해 28개의 QM을 인코딩한다.

[0060] 인트라-프레임 예측 모드에서는 대각 주사 시퀀스의 QM에 차동 펄스 코드 변조(Differential Pulse Code Modulation, DPCM) 인코딩이 적용된다. DPCM 인트라-프레임 잔여도 비트스트림으로 전송되어야 한다. 예를 들어, 도 9에 도시된 바와 같이, 일 실시예에 따른 대각 스캐닝 시퀀스의 개략도에서, 4×4 크기의 QM이 예로 사용된다. 대각선 스캐닝 시퀀스는 $(0, 0), (1, 0), (0, 1), (2, 0), (1, 1), \dots, (2, 3), (3, 3)$ 이다.

[0061] 인터-프레임 예측 모드에는 복사 모드와 예측 모드의 두 가지가 있다. 복사 모드에서 현재 인코딩될 QM은 디코딩이 가능한 QM과 완전히 동일하며 참조 QM이라고 한다. 이것은 또한 복사 모드가 인터-프레임 잔여를 0으로 하고 잔여 알림을 위한 신호를 전송할 필요가 없다는 것을 의미한다. 인코더는 현재 QM과 참조 QM 사이의 증분 ID를 전송하여 디코더가 참조 QM을 직접 복사하여 현재 QM을 재구성할 수 있다. 예측 모드는 복사 모드와 유사하지만 추가 인터-프레임 잔여가 있다. DPCM 인코딩은 대각 스캐닝 시퀀스에서 인터-프레임 잔여에 적용되며, 인코더는 DPCM 인터-프레임 잔여를 비트스트림으로 전송해야 한다.

[0062] 전술한 바와 같이, QM의 sizeId가 3보다 큰 경우, QM의 각 요소를 큰 정사각형 영역에 복사하기 위해 업샘플링 알고리즘이 사용된다. 위치 $(0, 0)$ 의 DC 계수는 비디오 재구성에 가장 중요하므로 VTM7에서는 DC 계수가 다른 QM의 해당 요소에서 복사되지 않고 직접 인코딩된다. 각 QM에 대해 QM의 세 가지 계산 모드(즉, 인터-프레임 예측 모드의 복사 모드, 인터-프레임 예측 모드의 예측 모드 및 인트라-프레임 예측 모드)의 비트 비용은 모드 결정을 통해 계산되고, 가장 낮은 비트 비용을 갖는 하나의 모드가 최종 최적 모드로 선택된다. 이어서, 최적 모드를 사용하여 QM을 인코딩한다.

[0063] 3. QM 시그널링

[0064] VVC는 QM을 사용하여 TB의 주파수 의존 양자화를 지원한다. QM이 W라고 가정하고, $W[x][y]$ 는 TB의 (x, y) 위치에서 변환 계수의 QM 가중치를 나타낸다. 변환 계수 $\text{coeff}[x][y]$ 에 대해 양자화된 변환 계수 레벨 $[x][y]$ 은 다음 수식 1을 사용하여 계산된다.

$$\text{level}[x][y] = (\text{coeff}[x][y] \times \frac{16}{W[x][y] \times QP} + offset) \quad \text{수식 1}$$

[0065] 여기서 QP는 양자화 파라미터(또는 양자화 단계로 지칭될 수 있음)이고 오프셋 값이다. $W[x][y]=16$ 은 변환 계수가 (x, y) 위치에서 가중되지 않음을 나타낸다. 또한, QM에 있는 모든 요소의 값이 16인 경우 효과는 QM을 사용하지 않는 효과와 동일하다.

[0066] 시퀀스 파라미터 세트(SPS)의 선택스 요소 `sps_scaling_list_enable_flag`는 SPS가 픽처 헤더(PH)에서 참조된 픽처에 대해 QM이 인에이블되는지 여부를 나타내기 위해 사용된다. 플래그가 인에이블되면, 즉 `sps_scaling_list_enable_flag`가 인에이블되면 PH의 추가 플래그를 사용하여 모든 요소가 16인 기본 QM을 사용할지 또는 사용자 정의 QM을 사용할지를 제어한다. VTM7에서 사용자 정의 QM은 적응 파라미터 세트(Adaptive Parameter Set, APS)로 통지된다. 사용자 정의 QM이 SPS 및 PH에서 인에이블된 경우, PH가 참조된 이미지의 QM 세트를 지정하기 위해 하나의 APS 인덱스가 PH에서 전송될 수 있다.

[0067] APS에서는 QM 인코딩 모드의 최대 28개 그룹, Δid (증분 id), AC 계수 및 DC 계수를 알려야 한다. 각 APS에서 28개의 QM 그룹이 id의 오름차순으로 인코딩 및 디코딩된다.

[0068] VVC 드래프트 7에서 QM 코딩 모드, Δid (증분 id), AC 계수 및 DC 계수의 선택스 및 시맨틱은 아래 표 4에 정의되어 있다.

표 4

<code>scaling_list_data() {</code>	디스크립터
<code>scaling_matrix_for_lfnst_disabled_flag</code>	<code>u(1)</code>
<code>for(id = 0; id < 28; id ++)</code>	

matrixSize = (id < 2) ? 2 : ((id < 8) ? 4 : 8)	
scaling_list_copy_mode_flag[id]	u(1)
if(!scaling_list_copy_mode_flag[id])	
scaling_list_pred_mode_flag[id]	u(1)
if(scaling_list_copy_mode_flag[id] scaling_list_pred_mode_flag[id]) &&	
id != 0 && id != 2 && id != 8)	
scaling_list_pred_id_delta[id]	ue(v)
if(!scaling_list_copy_mode_flag[id]) {	
nextCoef = 0	
if(id > 13) {	
scaling_list_dc_coef[id - 14]	se(v)
nextCoef += scaling_list_dc_coef[id - 14]	
}	
for(i = 0; i < matrixSize * matrixSize; i++) {	
x = DiagScanOrder[3][3][i][0]	
y = DiagScanOrder[3][3][i][1]	
if(!(id > 25 && x >= 4 && y >= 4)) {	
scaling_list_delta_coef[id][i]	se(v)
nextCoef += scaling_list_delta_coef[id][i]	
}	
ScalingList[id][i] = nextCoef	
}	
}	
}	

- [0071] scaling_list_copy_mode_flag[id]가 1이라는 것은 현재 QM의 요소 값이 현재 QM의 기준 QM의 요소 값과 동일함을 나타낸다. 참조 QM은 scaling_list_pred_id_delta[id]로 표시된다. scaling_list_copy_mode_flag[id]가 0인 것은 scaling_list_pred_mode_flag가 존재함을 나타낸다.
- [0072] scaling_list_pred_mode_flag[id]가 1인 것은 현재 QM이 기준 QM으로부터 예측될 수 있음을 나타낸다. 참조 QM은 scaling_list_pred_id_delta[id]로 표시된다. scaling_list_pred_mode_flag[id]가 0과 같으면 현재 QM의 요소 값이 명시적으로 신호를 전송하여 통지됨을 나타낸다. scaling_list_pred_mode_flag[id]가 존재하지 않는 경우, 그 값은 0으로 유추된다.
- [0073] scaling_list_pred_id_delta[id]는 예측 QM, 즉 ScalingMatrixPred[id]를 추론하는 데 사용되는 참조 QM을 나타낸다. scaling_list_pred_id_delta[id]가 존재하지 않는 경우, 그 값은 0으로 추론된다. scaling_list_pred_id_delta[id]의 값은 0 내지 maxIdDelta의 범위 내에 있어야 한다. maxIdDelta는 id에 따라 추론되며, 아래 수식 2와 같다.
- [0074] maxIdDelta=id<2?id((id<8)?(id-2):(id-8)) 수식 2
- [0075] 즉, id<2이면 maxIdDelta=id; id≥2이고 id<8이면 maxIdDelta=id-2이다. id≥8이면 maxIdDelta=id-8이다.
- [0076] 변수 refId 및 matrixSize는 다음 수식을 사용하여 계산된다:
- [0077] refId=id- scaling_list_pred_id_delta[id] 수식 3
- [0078] matrixSize=(id<2) ?:((id<8)?4:8) 수식 4
- [0079] 즉, id<2, matrixSize=2이다; id≥2 및 id<8이면, matrixSize=4이다; 그리고 id≥8이면, matrixSize=8이다.
- [0080] matrixSize×matrixSize의 QM 예측 행렬은 ScalingMatrixPred[x][y]로 표현되며, 여기서 x∈[0, matrixSize-1], y∈[0, matrixSize-1]이고; 변수 ScalingMatrixDCPred는 DC의 예측 값으로 표시되며 다음과 같이 계산된다.
- [0081] scaling_list_copy_mode_flag[id] 및 scaling_list_pred_mode_flag[id]가 모두 0인 경우 ScalingMatrixPred의 모든 요소는 8로 설정되고 ScalingMatrixDCPred의 값은 8로 설정된다.
- [0082] 그렇지 않고, scaling_list_pred_id_delta[id]가 0과 같을 때, ScalingMatrixPred의 모든 요소는 16과 같도

록 설정되고, ScalingMatrixDCPred의 값은 16과 같도록 설정된다.

[0083] 그렇지 않고, scaling_list_copy_mode_flag[id] 또는 scaling_list_pred_mode_flag[id]가 1이고 scaling_list_pred_id_delta[id]가 0보다 큰 경우, ScalingMatrixPred는 ScalingMatrixPred[refId]와 같도록 설정되고 ScalingMatrix의 값은 다음과 같이 계산된다: refId가 13보다 크면 ScalingMatrixDCPred의 값이 ScalingMatrixDCRec[refId - 14]와 같도록 설정되고; 그렇지 않으면(즉, refId가 13보다 작거나 같으면) ScalingMatrixDCPred의 값은 ScalingMatrixPred[0][0]과 같도록 설정된다.

[0084] scaling_list_dc_coef[id - 14]는 id가 13보다 클 때 변수 ScalingMatrixDC[id - 14]의 값을 계산하는 데 사용되며, 이는 아래의 수식 5와 같다:

[0085]
$$\text{ScalingMatrixDCRec}[\text{id} - 14] = (\text{ScalingMatrixDCPred} + \text{scaling_list_dc_coef}[\text{id} - 14] + 256) \% 256$$

수식 5

[0086] 여기서 %는 나머지를 구하는 것을 나타낸다.

[0087] scaling_list_dc_coef[id - 14]가 존재하지 않는 경우, 그 값은 0으로 추론된다. scaling_list_dc_coef[id - 14]의 값은 -128 내지 127(-128 및 127 포함)의 범위 내에 있어야 한다. ScalingMatrixDCRec[id - 14]의 값은 0보다 커야 한다.

[0088] scaling_list_delta_coef[id][i]는 scaling_list_copy_mode_flag[id]가 0일 때 ScalingList[id][i]와 이전 행렬 계수 ScalingList[id][i-1]의 차이를 나타낸다. scaling_list_delta_coef[id][i]의 값은 -128 내지 127(-128 및 127 포함) 범위 내에 있어야 한다. scaling_list_copy_mode_flag[id]가 1일 때 ScalingList[id]의 모든 요소는 0으로 설정된다.

[0089] matrixSize×matrixSize의 QM의 ScalingMatrixRec[id]는 하기 수식 6을 이용하여 계산될 수 있다:

[0090]
$$\text{ScalingMatrixRec}[\text{id}][\text{x}][\text{y}] = (\text{ScalingMatrixPred}[\text{x}][\text{y}] + \text{ScalingList}[\text{id}][\text{k}] + 256) \% 256$$

수식 6

[0091] 여기서 %는 나머지 $k \in [0, (\text{matrixSize} \times \text{matrixSize}-1)]$ 의 획득을 나타낸다.

[0092] $x = \text{DiagScanOrder}[\text{Log}_2(\text{matrixSize})][\text{Log}_2(\text{matrixSize})][k][0]$ 이고, 그리고

[0093] $y = \text{DiagScanOrder}[\text{Log}_2(\text{matrixSize})][\text{Log}_2(\text{matrixSize})][k][1]$ 이다.

[0094] ScalingMatrixRec[id][x][y]의 값은 0보다 커야 한다.

[0095] 디코딩 프로세스, 즉 QM의 ScalingMatrixRec[id][x][y] 및 ScalingMatrixDCRec을 얻기 위해 전술한 신택스 요소에 따라 디코딩을 수행하는 프로세스를 고려한다.

[0096] 4. SPS를 이용한 TB의 크기 제한

[0097] VVC 드래프트 7에서, TB 크기 제약과 관련된 SPS 신택스 및 의미론적 정의는 아래의 표 5에 나타나 있다:

표 5

[0098]	seq_parameter_set_rbsp() {	디스크립터
	...	
	chroma_format_idc	u(2)
	if(chroma_format_idc == 3)	
	separate_colour_plane_flag	u(1)
	...	
	sps_log2_ctu_size_minus5	u(2)
	...	
	sps_max_luma_transform_size_64_flag	u(1)
	...	
	log2_min_luma_coding_block_size_minus2	ue(v)
	...	
	if(separate_colour_plane_flag == 1)	
	colour_plane_id	u(2)

...	
}	

[0099] `sps_max_luma_transform_size_64_flag`가 1과 같다는 것은 루마 샘플링에서 최대 TB의 크기가 64와 같다는 것을 나타내고, `sps_max_luma_transform_size_64_flag`가 0과 같다는 것은 루마 샘플링에서 최대 TB의 크기가 32와 같다는 것을 나타낸다.

[0100] `chroma_format_idc`는 루마 샘플링에 대응하는 크로마 샘플링을 나타내며, 이는 하기 표 6과 같다:

표 6

<code>chroma_format_idc</code>	<code>separate_colour_plane_flag</code>	크로마 포맷	<code>SubWidthC</code>	<code>SubHeightC</code>
0	0	모노크롬	1	1
1	0	4:2:0	2	2
2	0	4:2:2	2	1
3	0	4:4:4	1	1
3	1	4:4:4	1	1

[0102] 표 6에서 `SubWidthC` 및 `SubHeightC`는 각각 크로마 성분에 대응하는 코딩 트리 단위(Coding Tree Unit, CTU)의 너비와 높이를 나타내고, 모노크롬(Monochrome)은 크로마 성분이 없음을 나타낸다.

[0103] 1과 동일한 `separate_colour_plane_flag`는 4:4:4 크로마 포맷의 3가지 색상 성분이 개별적으로 인코딩됨을 나타낸다. 0과 동일한 `separate_colour_plane_flag`는 색상 성분이 별도로 인코딩되지 않음을 나타낸다. `Separate_colour_plane_flag`가 존재하지 않는 경우, 그 값은 0으로 추론된다.

[0104] `Separate_colour_plane_flag`가 1일 때, 인코딩된 이미지는 3개의 개별 성분에 의해 형성되고, 각 성분은 컬러 평면(Y, Cb 또는 Cr)의 인코딩된 샘플에 의해 형성되며, 모노크롬 인코딩 신택스가 사용된다. 이 경우 각 색상 평면은 지정된 `color_plane_id` 값과 연결된다.

[0105] `color_plane_id`는 PH와 관련된 슬라이스와 관련된 컬러 평면을 지정한다. `split_colour_plane_flag`가 1과 같을 때 `color_plane_id`의 값은 0에서 2(0과 2 포함)의 범위 내에 있어야 한다. `color_plane_id`의 값 0, 1 및 2는 각각 평면 Y, Cb 및 Cr에 대응한다. `color_plane_id` 값이 다른 이미지의 디코딩 프로세스 간에는 종속성이 없다.

[0106] `sps_log2_ctu_size_minus5 plus 5`는 각 CTU의 루마 코딩 트리 블록 크기를 나타낸다. 2보다 작거나 같은 `sps_log2_ctu_size_minus5` 값은 비트스트림 일관성을 위한 요구 사항이다.

[0107] 최대 루마 코딩 블록 크기는 `sps_log2_ctu_size_minus5`에 기초하여 계산될 수 있다:

$$\text{CtbLog2SizeY} = \text{sps_log2_ctu_size_minus5} + 5$$

$$\text{CtbSizeY} = 1 \ll \text{CtbLog2SizeY}$$

[0110] 여기서 `CtbSizeY`는 최대 루마 코딩 블록 크기를 나타내고, `CtbLog2SizeY`는 2를 기준으로 하는 `CtbSizeY`의 로그를 나타내며, `<<`는 왼쪽 시프트 연산자이다.

[0111] `log2_min_luma_coding_block_size_minus2 plus 2`는 최소 루마 코딩 블록 크기를 나타낸다. `log2_min_luma_coding_block_size_minus2`의 값 범위는 0 및 `sps_log2_ctu_size_minus5+3`을 포함하여 0에서 `sps_log2_ctu_size_minus5+3`이어야 한다.

[0112] `MinCbLog2SizeY`, `MinCbSizeY`, `VSize`의 계산 프로세스는 다음과 같다:

$$\text{MinCbLog2SizeY} = \text{log2_min_luma_coding_block_size_minus2} + 2 \quad \text{수식 7}$$

$$\text{MinCbSizeY} = 1 \ll \text{MinCbLog2SizeY} \quad \text{수식 8}$$

$$\text{VSize} = \text{최소}(64, \text{CtbSizeY}) \quad \text{수식 9}$$

[0116] 여기서 `MinCbSizeY`는 최소 루마 코딩 블록 크기를 나타내고, `MinCbLog2SizeY`는 2를 기준으로 `MinCbSizeY`의 로그를 나타내며, `VSize`는 최대 루마 코딩 블록 크기를 나타내며, `<<`는 왼쪽 시프트 연산자이다. `MinCbSizeY`의 값

은 VSize보다 작거나 같아야 한다.

[0117] 각 크로마 코딩 트리 블록(coding tree block, CTB)의 너비와 높이, 즉 변수 CtbWidthC와 CtbHeightC는 다음과 같이 결정된다:

[0118] chroma_format_idc가 0(모노컬러)과 동일하거나 Separate_color_Plane_flag가 1과 동일하다면, CtbWidthC 및 CtbHeightC는 모두 0과 동일하다.

[0119] 그렇지 않으면, CtbWidthC 및 CtbHeightC는 아래 수식을 사용하여 계산된다:

$$\text{CtbWidthC} = \text{CtbSizeY} / \text{SubWidthC} \quad \text{수식 10}$$

$$\text{CtbHeightC} = \text{CtbSizeY} / \text{SubHeightC} \quad \text{수식 11}$$

[0122] 여기서 CtbSizeY는 루마 CTB의 크기를 나타낸다.

[0123] 현재 VVC에서 사용되는 QM을 인코딩하는 방법은 28개의 QM이 모두 APS에서 인코딩되어 전송되기 때문에 QM 시그널링이 상대적으로 많은 양의 코드워드를 점유해야 하고, 비트 오버헤드가 상대적으로 크며, 연산량이 많다. 디코더 측의 복잡성이 증가한다. 본 출원의 실시예에서 제공된 기술 솔루션에서, 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트가 획득되고; 유효 QM은 제1 파라미터 세트에 포함된 신택스 요소에 따라 결정되며, 유효 QM은 인코딩을 통해 디코딩될 비디오 프레임의 생성 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이고; 그런 다음 유효 QM이 디코더에 의해 디코딩된다. 이러한 방식으로 인코더 측에서는 유효 QM만 인코딩하여 전송하므로 QM 시그널링이 점유해야 하는 코드워드를 줄이고 비트 오버헤드를 줄이는 데 도움이 된다. 디코더 측에서는 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소한다.

[0124] 실시예에서 제공하는 기술 솔루션은 H.266/VVC 표준, 차세대 비디오 인코딩 및 디코딩 표준 또는 기타 인코딩 및 디코딩 표준에 적용될 수 있다.

[0125] 실시예에서 제공되는 비디오 디코딩 방법에서, 동작의 실행 엔티티는 디코더 측 디바이스이고, 실시예에서 제공되는 비디오 인코딩 방법에서, 동작의 실행 엔티티는 인코더 측 디바이스이다. 디코더 측 장치와 인코더 측 장치는 모두 컴퓨터 장치일 수 있다. 컴퓨터 장치는 PC, 휴대폰, 태블릿 컴퓨터, 미디어 플레이어, 전용 화상 회의 장치 또는 서버와 같이 데이터 계산, 처리 및 저장 기능을 갖춘 전자 디바이스를 말한다.

[0126] 또한, 여기에 제공된 방법은 단독으로 사용하거나 임의의 순서로 다른 방법과 조합하여 사용할 수 있다. 본 명세서에 제공된 방법에 기초한 인코더 및 디코더는 하나 이상의 프로세서 또는 하나 이상의 집적 회로에 의해 구현될 수 있다. 기술 솔루션은 여러 실시예를 사용하여 아래에 설명된다.

[0127] 도 10은 일 실시예에 따른 비디오 디코딩 방법의 흐름도이다. 본 실시예에서, 주로 위에서 소개한 디코더 측 장치에 방법을 적용한 예를 사용하여 설명된다. 방법은 다음 동작(1001 내지 1003)을 포함할 수 있다:

[0128] 동작 1001: 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트를 획득한다.

[0129] 디코딩될 비디오 프레임은 디코딩될 비디오에서 임의의 디코딩될 비디오 프레임(또는 이미지 프레임으로 지칭됨)일 수 있다. 제1 파라미터 세트는 QM을 정의하는 데 사용되는 관련 신택스 요소를 포함하는 파라미터 세트이다. 예를 들어, 디코더 측 장치는 제1 파라미터 세트에 포함된 신택스 요소에 따른 디코딩을 통해 QM을 획득할 수 있다.

[0130] 선택적으로, 제1 파라미터 세트는 APS이다. 다른 실시예에서, 제1 파라미터 세트는 대안적으로 APS가 아닐 수 있고 대안적으로 SPS와 같은 다른 파라미터일 수 있다.

[0131] 동작 1002: 제1 파라미터 세트에 포함된 신택스 요소에 따라 유효 QM을 결정하며, 유효 QM은 디코딩될 비디오 프레임의 디코딩 동안 양자화된 변환 계수를 역양자화하는 데 실제로 사용되는 QM이다.

[0132] 양자화된 변환 계수를 역양자화하는 데 사용될 수 있는 QM의 양이 n이라고 가정하고, 유효 QM의 양은 n보다 작거나 같을 수 있고, n은 양의 정수일 수 있다. 예를 들어, n개의 QM 모두가 양자화된 변환 계수를 역양자화하기 위해 실제로 사용되는 경우, 유효 QM의 양은 n이고; 그리고 모든 n개의 QM 중 일부 QM(예를 들어, m개의 QM, m은 n보다 작은 양의 정수)이 양자화된 변환 계수를 역양자화하기 위해 실제로 사용되는 경우, 유효 QM의 양은 m이다.

[0133] 제1 파라미터 세트에서 유효 QM을 결정하는 데 사용되는 신택스 요소를 정의함으로써, 디코더 측 장치는 신택스 요소를 판독함으로써 어떤 QM이 유효 QM이고 어떤 QM이 유효 QM이 아닌지 결정할 수 있다. 유효 QM이 아닌 QM

(무효 QM이라 할 수 있음), 즉 인코딩을 통해 디코딩될 비디오 프레임 생성 시 변환 계수를 양자화하는데 실제로 사용되지 않는 QM의 경우, 디코더 측 장치가 QM을 디코딩하는 것이 불필요할 수 있다.

[0134] 선택적으로, 유효 QM이 아닌 다른 QM의 경우, QM의 모든 요소는 디폴트 값에 의해 미리 정의된다. 예를 들어, 디폴트 값은 16일 수 있다. 수식 1을 참조하면, 이 경우 TB의 모든 변환 계수의 스케일링 양자화 계수는 1이므로 QM을 사용하지 않는 것과 같은 효과가 있다.

[0135] 동작 1003: 유효 QM을 디코딩한다.

[0136] 유효 QM이 결정된 후, 하나 이상의 유효 QM이 있을 수 있기 때문에 디코더 측 장치는 각각의 유효 QM을 개별적으로 디코딩할 필요가 있다. 유효 QM이 예로 사용된다. 유효 QM을 디코딩할 때, 유효 QM에 대응하는 인코딩 모드를 결정하고, 그 인코딩 모드에 따라 유효 QM을 디코딩할 수 있다.

[0137] 예를 들어, 표 1을 참조하면, 변환 계수를 양자화하는 데 사용될 수 있는 QM의 양은 28일 수 있다. 그 안의 12개의 QM이 유효 QM이라고 결정된다고 가정하면, 디코더 측 장치는 12개의 유효 QM을 디코딩하기만 하면 되고 나머지 16개의 비효율 QM을 디코딩할 필요가 없다.

[0138] 상기에 기초하여, 본 출원의 이 실시예에서 제공되는 기술 솔루션에서, 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트가 획득되고; 유효 QM은 제1 파라미터 세트에 포함된 신택스 요소에 따라 결정되며, 유효 QM은 인코딩을 통해 디코딩될 비디오 프레임의 생성 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이고; 그런 다음 유효 QM이 디코딩된다. 이러한 방식으로 디코더 측에서는 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소한다.

[0139] 일 실시예에서, 제1 파라미터 세트에 포함된 신택스 요소에 따라 유효 QM을 결정하는 것은 전술한 하위 동작을 포함한다:

[0140] 1: 제1 파라미터 세트에 포함된 신택스 요소에 따라 QM의 유효 크기 범위를 결정한다.

[0141] QM의 유효 크기 범위는 디코딩 시 양자화된 변환 계수를 역양자화하기 위해 실제로 사용되는 QM의 최소 크기 및 최대 크기를 정의한다. 또한, QM 크기의 값은 2, 4, 8, 16, 32, 64와 같이 2의 지수 거듭제곱이다.

[0142] 2: 유효 크기 범위에 속하는 QM을 유효 QM으로 결정한다.

[0143] 예를 들어, QM의 유효 크기 범위가 [4, 32]인 경우, 유효 QM은 4×4 크기의 QM, 8×8 크기의 QM, 16×16 크기의 QM, 및 32×32 크기의 QM을 포함한다. 다른 예로, QM의 유효 크기 범위가 [8, 16]인 경우, 유효 QM은 8×8 크기의 QM과 16×16 크기의 QM을 포함한다.

[0144] 예를 들어, 결정된 QM의 유효 크기 범위가 [8, 16]이라고 가정하고, 표 1 및 표 2를 참조하면, 크기가 8×8 인 QM에 대응하는 sizeId는 3이고, 크기가 16×16 인 QM에 대응하는 sizeId는 4이다. 디코더 측 장치는 id가 8 내지 19인 총 12개의 QM이 유효 QM이고 id가 0 내지 7 및 20인 나머지 16개의 QM을 결정한다. 27개는 무효 QM이다.

[0145] 일 예에서, QM의 유효 크기 범위는 다음과 같은 방식으로 제1 파라미터 세트에 포함된 신택스 요소에 따라 결정된다:

[0146] 1.1: 제1 파라미터 세트에 포함된 신택스 요소에 따라 최소 루마 코딩 블록 크기, 루마 코딩 트리의 블록 크기, 및 최대 루마 TB 크기를 결정한다.

[0147] 선택적으로, 제1 신택스 요소는 제1 파라미터 세트에 정의되고, 제1 신택스 요소는 최소 루마 코딩 블록 크기를 나타내기 위해 사용되며; 제2 신택스 요소는 제1 파라미터 세트에 정의되고, 제2 신택스 요소는 루마 코딩 트리의 블록 크기를 나타내는 데 사용되며; 제3 신택스 요소는 제1 파라미터 세트에서 정의되고, 제3 신택스 요소는 최대 루마 TB 크기를 표시하는 데 사용된다. 디코더 측 장치는 제1 파라미터 세트에서 제1 신택스 요소, 제2 신택스 요소 및 제3 신택스 요소를 읽고 최소 루마 코딩 블록 크기, 루마 코딩 트리의 블록 크기 및 최대 루마 TB 크기를 결정한다.

[0148] 1.2: 최소 루마 코딩 블록 크기, 루마 코딩 트리의 블록 크기, 및 최대 루마 TB 크기에 따라 루마 QM의 유효 크기 범위를 결정하고, 루마 QM의 유효 크기 범위는 루마 QM의 최소 크기 및 최대 크기를 포함한다.

[0149] 선택적으로, 디코더 측 장치는 최소 루마 코딩 블록 크기에 따라 루마 QM의 최소 크기를 결정한다. 예를 들어, 최소 루마 코딩 블록 크기는 루마 QM의 최소 크기로 결정된다. 디코더 측 장치는 루마 코딩 트리의 블록 크기 및 최대 루마 TB 크기 중 더 큰 값을 루마 QM의 최대 크기로 결정한다. 예를 들어, 루마 코딩 트리의 블록 크기

가 최대 루마 TB 크기보다 큰 경우, 루마 코딩 트리의 블록 크기는 루마 QM의 최대 크기로 결정되고; 루마 코딩 트리의 블록 크기가 최대 루마 TB 크기보다 작은 경우, 최대 루마 TB 크기는 루마 QM의 최대 크기로 결정되고; 그리고 루마 코딩 트리의 블록 크기가 최대 루마 TB 크기와 같을 때, 둘은 동일하기 때문에 루마 코딩 트리의 블록 크기는 루마 QM의 최대 크기로 결정되거나, 또는 최대 루마 TB 크기는 루마 QM의 최대 크기로 결정되며, 이는 동일한 결과를 갖는다.

[0150] 1.3: 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하며, 크로마 QM의 유효 크기 범위는 크로마 QM의 최소 크기 및 최대 크기를 포함한다.

[0151] 선택적으로, 제4 신택스 요소는 제1 파라미터 세트에 정의되고, 제4 신택스 요소는 루마 성분에 대한 크로마 성분의 샘플링 레이트를 나타내기 위해 사용된다.

[0152] 선택적으로, 디코더 측 장치는 루마 QM의 최소 크기 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 최소 크기를 계산하고; 루마 QM의 최대 크기 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 최대 크기를 계산한다.

[0153] 일 실시예에서, 제1 파라미터 세트가 APS인 예가 사용된다. APS에 포함된 신택스 요소와 신택스 구조 테이블은 아래의 표 7과 같다:

표 7

[0154]	scaling_list_data() { scaling_matrix_for_lfnst_disabled_flag aps_qm_size_info_present_flag if (aps_qm_size_info_present_flag) { aps_log2_ctu_size_minus5 aps_log2_min_luma_coding_block_size_minus2 aps_max_luma_transform_size_64_flag aps_chroma_format_idc } for(id = 0; id < 28; id ++) { matrixSize = (id < 2) ? 2 : ((id < 8) ? 4 : 8) if((cIdx==0 && (matrixQMSize >= minQMSizeY && matrixQMSize <= maxQMSizeY) . (cIdx!=0 && (matrixQMSize >= minQMSizeUV && matrixQMSize <= maxQMSizeUV)) { scaling_list_copy_mode_flag[id] if(!scaling_list_copy_mode_flag[id]) scaling_list_pred_mode_flag[id] if((scaling_list_copy_mode_flag[id] scaling_list_pred_mode_flag [id]) && id != 0&& id != 2&& id != 8) scaling_list_pred_id_delta[id] if(!scaling_list_copy_mode_flag[id]) { nextCoef = 0 if(id > 13) { scaling_list_dc_coef[id - 14] nextCoef += scaling_list_dc_coef[id - 14] } for(i = 0; i < matrixSize * matrixSize; i++) { x = DiagScanOrder[3][3][i][0] y = DiagScanOrder[3][3][i][1] if(!(id > 25 && x >= 4 && y >= 4)) { scaling_list_delta_coef[id][i] nextCoef += scaling_list_delta_coef[id][i] } ScalingList[id][i] = nextCoef } } } }	디스크립터 u(1) u(1) u(2) ue(v) u(1) u(2) u(1) ue(v) u(1) ue(v) se(v) se(v) se(v)
--------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------

}	
---	--

- [0155] `aps_qm_size_info_present_flag`는 QM 크기와 관련된 선택스 요소가 비트스트림에 존재하는지 여부를 나타낸다. `aps_qm_size_info_present_flag`의 값이 1이면 비트스트림에 QM 크기와 관련된 선택스 요소가 존재함을 나타내며, 이를 기반으로 QM의 유효 크기 범위를 결정하여 어떤 크기의 QM을 디코딩해야 하는지 결정할 수 있다. `aps_qm_size_info_present_flag`의 값이 0이면 비트스트림에 QM 크기와 관련된 선택스 요소가 존재하지 않으며 모든 크기의 QM이 디코딩되어야 함을 나타낸다. `aps_log2_ctu_size_minus5`의 값은 루마 코딩 트리의 블록 크기를 나타낸다. 그 값은 선택스 요소 `sps_log2_ctu_size_minus5`의 값과 동일하도록 지정된다.
- [0156] `aps_log2_min_luma_coding_block_size_minus2` plus 2의 값은 최소 루마 코딩 블록 크기를 나타낸다. 그 값은 선택스 요소 `sps_log2_min_luma_coding_block_size_minus2`의 값과 동일하도록 지정된다.
- [0157] `aps_max_luma_transform_size_64_flag`의 값이 1인 것은 최대 루마 TB 크기가 64임을 나타내고; 값이 0인 것은 최대 루마 TB 크기가 32임을 나타낸다. 그 값은 선택스 요소 `sps_max_luma_transform_size_64_flag`의 값과 동일하도록 지정된다.
- [0158] `aps_chroma_format_idc`는 루마 성분에 대한 크로마 성분의 샘플링 레이트를 나타내며, 이는 구체적으로 표 6에 나타나 있다. 그 값은 선택스 요소 `chroma_format_idc`의 값과 동일하도록 지정된다.
- [0159] 전술한 선택스 요소에 기초하여 변수 `minQMSizeY`(루마 QM의 최소 크기를 나타냄) 및 `maxQMSizeY`(루마 QM의 최대 크기를 나타냄)의 추론 프로세스는 다음과 같다:
- [0160] 선택스 요소 `aps_qm_size_info_present_flag`의 값이 1일 때,
- ```
minQMSizeY = 1 << (aps_log2_min_luma_coding_block_size_minus2+2) 수식 12
```
- [0161] `maxQMSizeY = max( 1<< (aps_log2_ctu_size_minus5+5), aps_max_luma_transform_size_64_flag ? 64: 32)`  
수식 13
- [0163] 여기서 <<는 왼쪽 시프트 연산자이고 그리고 ?:는 삼항 조건 연산자(ternary condition operator)이다.
- [0164] 선택스 요소 `aps_qm_size_info_present_flag`의 값이 0일 때,
- [0165] `minQMSizeY = 4`이고;
- [0166] `maxQMSizeY = 64`이다.
- [0167] 변수 `minQMSizeUV`(크로마 QM의 최소 크기) 및 `maxQMSizeUV`(크로마 QM의 최대 크기)의 추론 프로세스는 다음과 같다:
- [0168] 선택스 요소 `aps_qm_size_info_present_flag`의 값이 1일 때,
- ```
minQMSizeUV=( !aps_chroma_format_idc)?0:minQMSizeY/SubWidthC 수식 14
```
- [0170] `maxQMSizeUV=(!aps_chroma_format_idc)?0:maxQMSizeY/SubHeightC` 수식 15
- [0171] where!는 논리적 not 연산을 나타내고 ?:는 삼항 조건 연산자이다.
- [0172] 수식 14 및 수식 15의 정의는 다음과 같다:
- [0173] `aps_chroma_format_idc`가 존재하지 않으면 `minQMSizeUV=0`이고; 그렇지 않으면 `minQMSizeUV = minQMSizeY / SubWidthC`; 그리고
- [0174] `aps_chroma_format_idc`가 존재하지 않으면 `maxQMSizeUV=0`이고; 그렇지 않으면 `maxQMSizeUV = maxQMSizeY / SubHeightC`이다.
- [0175] 선택스 요소 `aps_qm_size_info_present_flag`의 값이 0일 때,
- [0176] `minQMSizeUV = 2`이고;
- [0177] `maxQMSizeUV = 32`이다.
- [0178] 표 7의 선택스 구조 테이블에서 변수 `cIdx`는 현재 QM에 대응하는 색상 성분을 나타낸다. 루마 성분 Y의 경우, 그 값은 0이고; 크로마 Cb의 경우, 그 값은 1이고; 크로마 Cr의 경우, 그 값은 2이다. 변수 `matrixSize`는 현재

QM의 실제 인코딩 크기를 나타내며, 이는 표 2의 세 번째 열로 표시된다. 변수 matrixQMSize는 현재 QM에 대응하는 TB 크기를 나타내며, 이는 표 1 및 표 2에 나와 있다.

[0179] 표 7의 선택스 구조 테이블에서 디코더 측 장치는 먼저 본 출원에서 제안하는 두 가지 조건을 결정한 후 현재 QM을 디코딩할지 여부를 결정한다. 제1 QM이 유효 QM인지 판단하는 예(제1 QM은 사용 가능한 모든 QM, 즉, 전술한 28개의 QM 중 어느 하나일 수 있음)가 사용된다. 제1 QM이 제1 조건 및 제2 조건 중 하나를 충족하는 경우 제1 QM이 유효 QM으로 결정된다.

[0180] 제1 조건은 `cIdx==0 &&(matrixQMSize >= minQMSizeY && matrixQMSize <= maxQMSizeY)`이며, 제1 QM이 루마 성분이고 루마 TB의 양자화 프로세스에 사용됨을 나타내고, 제1 QM은 루마 QM의 유효 크기 범위 [MinQMSizeY, MaxQMSizeY] 내에 있고, MinQMSizeY는 루마 QM의 최소 크기를 나타내고, MaxQMSizeY는 루마 QM의 최대 크기를 나타낸다. 제2 조건은 `cIdx!=0 &&(matrixQMSize >= minQMSizeUV && matrixQMSize <= maxQMSizeUV)`이며, 제1 QM이 크로마 성분이고 크로마 TB의 양자화 프로세스에 사용됨을 나타내고, 제1 QM은 크로마 QM의 유효 크기 범위 [MinQMSizeUV, MaxQMSizeUV] 내에 있고, MinQMSizeUV는 크로마 QM의 최소 크기를 나타내고, MaxQMSizeUV는 크로마 QM의 최대 크기를 나타낸다.

[0181] 전술한 예에서, 디코더 측 장치는 제1 파라미터 세트에 포함된 선택스 요소에 기초하여 QM의 유효 크기 범위를 계산하고, 그런 다음 유효 크기 범위에 따라 유효 QM을 결정할 필요가 있다. 후술되는 예에서, 루마 QM의 유효 크기 범위의 선택스 요소는 대안적으로 제1 파라미터 세트에서 직접 정의될 수 있다. 선택스 요소를 읽은 후, 디코더 측 장치는 루마 QM의 유효 크기 범위를 직접 획득하고, 그런 다음 크로마 포맷을 참조하여 루마 QM의 유효 크기 범위에 따라 크로마 QM의 유효 크기 범위를 결정할 수 있다. 세부 사항은 다음과 같다:

[0182] 다른 예에서, QM의 유효 크기 범위는 다음 방식으로 제1 파라미터 세트에 포함된 선택스 요소에 따라 결정된다:

[0183] 1.1: 루마 QM의 최소 크기 및 최대 크기를 포함하는 루마 QM의 유효 크기 범위인 제1 파라미터 세트에 포함된 선택스 요소에 따라 루마 QM의 유효 크기 범위를 결정한다.

[0184] 선택적으로, 제5 선택스 요소는 제1 파라미터 세트에 정의되고, 제5 선택스 요소는 루마 QM의 최소 크기를 나타내는 데 사용되며; 제6 선택스 요소는 APS에서 정의되고, 제6 선택스 요소는 루마 QM의 최대 크기를 표시하는 데 사용된다. 디코더 측 장치는 루마 QM의 최소 크기 및 최대 크기를 결정하기 위해 제1 파라미터 세트로부터 제5 선택스 요소 및 제6 선택스 요소를 판독한다.

[0185] 1.2: 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하며, 크로마 QM의 유효 크기 범위는 크로마 QM의 최소 크기 및 최대 크기를 포함한다.

[0186] 선택적으로, 제4 선택스 요소는 제1 파라미터 세트에 정의되고, 제4 선택스 요소는 루마 성분에 대한 크로마 성분의 샘플링 레이트를 표시하는 데 사용된다.

[0187] 선택적으로, 디코더 측 장치는 루마 QM의 최소 크기 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 최소 크기를 계산하고; 루마 QM의 최대 크기 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 최대 크기를 계산한다.

[0188] 일 실시예에서, 제1 파라미터 세트가 APS인 예가 사용된다. APS에 포함된 선택스 요소와 구문 구조 테이블은 아래 표 8과 같다.

표 8

	디스크립터
<code>scaling_list_data() {</code>	
<code>scaling_matrix_for_lfnst_disabled_flag</code>	u(1)
<code>aps_qm_size_info_present_flag</code>	u(1)
<code>if (aps_qm_size_info_present_flag) {</code>	
<code>aps_log2_min_luma_qm_size_minus2</code>	ue(v)
<code>aps_log2_max_luma_qm_size_minus5</code>	ue(v)
<code>aps_chroma_format_idc</code>	ue(v)
<code>}</code>	
<code>for(id = 0; id < 28; id ++) {</code>	
<code>matrixSize = (id < 2) ? 2 : ((id < 8) ? 4 : 8)</code>	

if((cIdx==0 && (matrixQMSize >= minQMSizeY && matrixQMSize <= maxQMSizeY). (cIdx!=0 && (matrixQMSize >= minQMSizeUV && matrixQMSize <= maxQMSizeUV)) {	
scaling_list_copy_mode_flag[id]	u(1)
if(!scaling_list_copy_mode_flag[id])	
scaling_list_pred_mode_flag[id]	u(1)
if((scaling_list_copy_mode_flag[id] scaling_list_pred_mode_flag [id]) && id != 0&& id != 2&& id != 8)	
scaling_list_pred_id_delta[id]	ue(v)
if(!scaling_list_copy_mode_flag[id]) {	
nextCoef = 0	
if(id > 13) {	
scaling_list_dc_coef[id - 14]	se(v)
nextCoef += scaling_list_dc_coef[id - 14]	
}	
for(i = 0; i < matrixSize * matrixSize; i++) {	
x = DiagScanOrder[3][3][i][0]	
y = DiagScanOrder[3][3][i][1]	
if(!(id > 25 && x >= 4 && y >= 4)) {	
scaling_list_delta_coef[id][i]	se(v)
nextCoef += scaling_list_delta_coef[id][i]	
}	
ScalingList[id][i] = nextCoef	
}	
}	
}	
}	
}	

[0190] aps_qm_size_info_present_flag는 QM 크기와 관련된 선택스 요소가 비트스트림에 존재하는지 여부를 나타낸다. aps_qm_size_info_present_flag의 값이 1이면 QM 크기와 관련된 선택스 요소가 비트스트림에 나타나며, 이를 기반으로 QM의 유효 크기 범위가 결정되어 어떤 크기의 QM이 디코딩되어야 하는지 결정할 수 있다. aps_qm_size_info_present_flag의 값이 0이면 비트스트림에 QM 크기와 관련된 선택스 요소가 존재하지 않으며 모든 크기의 QM이 디코딩되어야 함을 나타낸다.

[0191] aps_log2_min_luma_qm_size_minus2 plus 2의 값은 루마 QM의 최소 크기를 나타낸다.

[0192] aps_log2_max_luma_qm_size_minus5 plus 5의 값은 루마 QM의 최대 크기를 나타낸다.

[0193] 위의 선택스 요소에 기초하여 변수 minQMSizeY(루마 QM의 최소 크기를 나타냄) 및 maxQMSizeY(루마 QM의 최대 크기를 나타냄)의 추론 프로세스는 다음과 같다:

[0194] 선택스 요소 aps_qm_size_info_present_flag의 값이 1일 때,

[0195] minQMSizeY = 1 << (aps_log2_min_luma_qm_size_minus2 + 2) 수식 16

[0196] maxQMSizeY = 1 << (aps_log2_max_luma_qm_size_minus5 + 5) 수식 17

[0197] 여기서 <<는 왼쪽 시프트 연산자이다.

[0198] aps_qm_size_info_present_flag의 값이 1일 때, minQMSizeY maxQMSizeY는 SPS 선택스 요소 및 VSize 값에 기초하여 계산된 TB 크기 변수 MinCbSizeY와 [0175] 각각 동일하도록 지정된다.

[0199] 선택스 요소 aps_qm_size_info_present_flag의 값이 0일 때,

[0200] minQMSizeY = 4이고;

[0201] maxQMSizeY = 64이다.

[0202] aps_chroma_format_idc는 루마 성분에 대한 크로마 성분의 샘플링 레이트를 나타내며, 이는 구체적으로 표 6에 나타나 있다. 그 값은 선택스 요소 chroma_format_idc의 값과 동일하도록 지정된다.

- [0203] 변수 minQMSizeUV(크로마 QM의 최소 크기) 및 maxQMSizeUV(크로마 QM의 최대 크기)의 추론 프로세스는 다음과 같다:
- [0204] 선택스 요소 aps_qm_size_info_present_flag의 값이 1일 때,
- [0205] $\text{minQMSizeUV} = (\text{!aps_chroma_format_idc}) ? 0 : \text{minQMSizeY} / \text{SubWidthC}$ 수식 18
- [0206] $\text{maxQMSizeUV} = (\text{!aps_chroma_format_idc}) ? 0 : \text{maxQMSizeY} / \text{SubHeightC}$ 수식 19
- [0207] where !는 논리적 not 연산을 나타내고 ?는: 삼항 조건 연산자이다.
- [0208] 선택스 요소 aps_qm_size_info_present_flag의 값이 0일 때,
- [0209] $\text{minQMSizeUV} = 2$ 이고;
- [0210] $\text{maxQMSizeUV} = 32$ 이다.
- [0211] 일부 다른 예들에서, 디코더 측 디바이스는 대안적으로 SPS에 포함된 선택스 요소들에 따라 유효 QM을 결정할 수도 있다. 구체적으로, 디코더 측 장치는 SPS에 포함된 선택스 요소에 따라 루마 QM의 유효 크기 범위 [MinQMSizeY, MaxQMSizeY] 및 크로마 QM의 유효 크기 범위 [MinQMSizeUV, MaxQMSizeUV]를 계산할 수 있다. MinQMSizeY 변수는 루마 QM의 최소 크기를 나타내고, MaxQMSizeY 변수는 루마 QM의 최대 크기를 나타내고, MinQMSizeUV 변수는 크로마 QM의 최소 크기를 나타내고, MaxQMSizeUV 변수는 크로마 QM의 최대 크기를 나타낸다.
- [0212] 표 5에 나타낸 SPS 선택스 구조 테이블을 참조하면, 전술한 변수는 다음 수식을 사용하여 계산될 수 있다:
- [0213] $\text{MinQMSizeY} = 1 << (\text{log2_min_luma_coding_block_size_minus2} + 2)$ 수식 20
- [0214] $\text{MaxQMSizeY} = \max(1 << (\text{sps_log2_ctu_size_minus5} + 5), \text{sps_max_luma_transform_size_64_flag} ? 64 : 32)$ 수식 21
- [0215] $\text{MinQMSizeUV} = (\text{!chroma_format_idc}) ? 0 : \text{MinQMSizeY} / \text{SubWidthC}$ 수식 22
- [0216] $\text{MaxQMSizeUV} = (\text{!chroma_format_idc}) ? 0 : \text{MaxQMSizeY} / \text{SubHeightC}$ 수식 23
- [0217] 여기서 <<는 왼쪽 시프트 연산자, ! 논리적 not 연산을 나타내고 ?는: 삼항 조건 연산자이다.
- [0218] SPS에 포함된 선택스 요소에 따라 유효 QM을 결정하는 것과 비교하여, APS에서 관련 선택스 요소를 정의하고, APS에 정의된 관련 선택스 요소에 따라 유효 QM을 결정함으로써, APS와 APS 간의 종속성을 파싱한다. APS의 디코딩이 SPS의 선택스 요소에 의존할 필요가 없도록 SPS 비트스트림이 제거될 수 있다.
- [0219] 일 실시예에서, 제1 파라미터 세트에 포함된 선택스 요소에 따라 유효 QM을 결정하는 것은 전술한 하위 단계를 포함한다:
- [0220] 1) 제1 파라미터 세트로부터 제1 QM에 대응하는 플래그 선택스 요소의 값을 판독하는 단계;
 - [0221] 2) 제1 QM에 대응하는 플래그 선택스 요소의 값이 제1 값인 경우, 제1 QM이 유효 QM이라고 결정하는 단계; 및
 - [0222] 3) 제1 QM에 대응하는 플래그 선택스 요소의 값이 제2 값인 경우, 제1 QM이 유효 QM이 아니라고 결정하는 단계.
- [0223] 이 실시예에서, 플래그 선택스 요소는 APS에 정의되고, QM이 유효 QM인지 여부는 플래그 선택스 요소에 의해 지시된다. 플래그 선택스 요소의 디스크립터는 1비트의 부호 없는 정수를 나타내는 u(1)일 수 있다. 예를 들어, 플래그 선택스 요소의 값이 1이면 QM이 유효 QM이고 디코딩되어야 함을 나타낸다. 플래그 선택스 요소의 값이 0인 것은 QM이 유효 QM이 아니며 디코딩될 필요가 없음을 표시한다. 디코딩되지 않은 QM의 경우 QM의 모든 요소가 디폴트 값으로 사전 정의된다. 선택적으로 디폴트 값은 16이다. 수식 1을 참조하면 이 경우 TB의 모든 변환 계수의 스케일링 양자화 계수는 1이므로 QM을 사용하지 않는 것과 같은 효과가 있다.
- [0224] 또한, 제1 QM은 임의의 사용 가능한 QM, 즉 전술한 28개의 QM 중 어느 하나일 수 있다.
- [0225] 선택적으로, 제1 파라미터 세트는 APS이다. 확실히, 일부 다른 실시예에서, 제1 파라미터 세트는 대안적으로 APS가 아닐 수 있다. 제1 파라미터는 이에 한정되지 않고 실시예에 따라 달라질 수 있다.
- [0226] 일 실시예에서, 제1 파라미터 세트가 APS인 예가 사용된다. APS에 포함된 선택스 요소와 선택스 구조 테이블은 아래 표 9와 같다.

표 9

[0227]

scaling_list_data() {	디스크립터
scaling_matrix_for_lfnst_disabled_flag	u(1)
for(id = 0; id < 28; id ++) {	
matrixSize = (id < 2) ? 2 : ((id < 8) ? 4 : 8)	
scaling_matrix_present_flag[id]	u(1)
if(scaling_matrix_present_flag[id]) {	
scaling_list_copy_mode_flag[id]	u(1)
if(!scaling_list_copy_mode_flag[id])	
scaling_list_pred_mode_flag[id]	u(1)
if((scaling_list_copy_mode_flag[id] scaling_list_pred_mode_flag[id])	
&& id != 0 && id != 2 && id != 8)	
scaling_list_pred_id_delta[id]	ue(v)
if(!scaling_list_copy_mode_flag[id]) {	
nextCoef = 0	
if(id > 13) {	
scaling_list_dc_coef[id - 14]	se(v)
nextCoef += scaling_list_dc_coef[id - 14]	
}	
for(i = 0; i < matrixSize * matrixSize; i++) {	
x = DiagScanOrder[3][3][i][0]	
y = DiagScanOrder[3][3][i][1]	
if(!(id > 25 && x >= 4 && y >= 4)) {	
scaling_list_delta_coef[id][i]	se(v)
nextCoef += scaling_list_delta_coef[id][i]	
}	
ScalingList[id][i] = nextCoef	
}	
}	
}	
}	
}	

[0228]

선택적으로, 플래그 선택스 요소는 scaling_matrix_present_flag이다. scaling_matrix_present_flag[id]의 값이 1이면 현재 QM이 디코딩되어야 함을 나타낸다. scaling_matrix_present_flag[id]의 값이 0이면 현재 QM이 디코딩될 필요가 없음을 나타내고 디코더 측 장치는 QM의 모든 요소가 16이라고 추론할 수 있다.

[0229]

선택적으로, 루마 QM은 루마 QM이 디코딩될 필요가 있는지 여부를 나타내는 하나의 플래그 선택스 요소에 대응한다. 동일한 예측 모드 및 동일한 크기를 갖는 제1 크로마 QM(Cb에 대응하는 QM) 및 제2 크로마 QM(Cr에 대응하는 QM)은 동일한 플래그 선택스 요소를 공유하며, 이는 제1 크로마 QM 및 제2 크로마가 QM이 디코딩되어야 하는지를 한다. 즉, 제1 크로마 QM 및 제2 크로마 QM에 대해 플래그 선택스 요소를 별도로 사용할 필요가 없어 QM 인코딩 시그널링의 비트 오버헤드를 더 줄일 수 있다.

[0230]

일 실시예에서, 제1 파라미터 세트가 APS인 예가 사용된다. APS에 포함된 선택스 요소와 선택스 구조 테이블은 아래 표 10과 같다:

표 10

[0231]

scaling_list_data() {	디스크립터
scaling_matrix_for_lfnst_disabled_flag	u(1)
for(id = 0; id < 28; id ++) {	
matrixSize = (id < 2) ? 2 : ((id < 8) ? 4 : 8)	
if(cIdx <= 1)	
scaling_matrix_present_flag[predMode != MODE_INTRA][cIdx != 0][sizeId]	u(1)
if(scaling_matrix_present_flag[predMode != MODE_INTRA][cIdx != 0][sizeId]) {	

scaling_list_copy_mode_flag[id]	u(1)
if(!scaling_list_copy_mode_flag[id])	
scaling_list_pred_mode_flag[id]	u(1)
if((scaling_list_copy_mode_flag[id] scaling_list_pred_mode_flag[id]) && id != 0&& id != 2&& id != 8)	
scaling_list_pred_id_delta[id]	ue(v)
if(!scaling_list_copy_mode_flag[id]) {	
nextCoef = 0	
if(id > 13) {	
scaling_list_dc_coef[id - 14]	se(v)
nextCoef += scaling_list_dc_coef[id - 14]	
}	
for(i = 0; i < matrixSize * matrixSize; i++) {	
x = DiagScanOrder[3][3][i][0]	
y = DiagScanOrder[3][3][i][1]	
if(!(id > 25 && x >= 4 && y >= 4)) {	
scaling_list_delta_coef[id][i]	se(v)
nextCoef += scaling_list_delta_coef[id][i]	
}	
ScalingList[id][i] = nextCoef	
}	
}	
}	
}	
}	

[0232] scaling_matrix_present_flag[predMode != MODE_INTRA][cIdx != 0][sizeId]의 값이 1일 때, 루마 QM으로 디코딩될 때 루마 QM이 APS로 인코딩됨을 나타내고; 크로마 QM으로 디코딩할 때, APS에서 predMode의 인코딩 예측 모드를 가지며 동일한 크기를 갖는 크로마 Cb 및 크로마 Cr에 대응하는 QM을 나타낸다. 신팩스 요소의 값이 0인 경우, 루마 QM 또는 2개의 크로마 QM이 디코딩될 필요가 없음을 나타내며, 디코더 측 장치는 그 요소가 모두 16인 것으로 추론할 수 있다.

[0233] 인코더 측 장치는 결정된 QM에 대응하는 플래그 신팩스 요소의 값을 결정할 수 있으며, 즉 QM의 크기에 따라, 또는 QM에 대응하는 인코딩 예측 모드에 따라, 또는 QM에 대응하는 YUV 색상 성분에 따라, 어느 QM이 인코딩되어야 하고 어느 QM이 인코딩되지 않아야 하는지를 결정하거나, 또는 QM의 크기, 인코딩 예측 모드 및 YUV 색상 성분 중 복수의 요소를 참조하여 포괄적인 고려가 수행될 수 있다. 실시예는 이에 제한되지 않는다.

[0234] 본 실시예에서, 플래그 신팩스 요소는 제1 파라미터 세트에 정의되고, QM이 유효 QM인지 여부는 플래그 신팩스 요소에 의해 표시되어, QM이 디코딩될 필요가 있는지 여부가 보다 유연하게 표시될 수 있다.

[0235] 도 11은 일 실시예에 따른 비디오 인코딩 방법의 흐름도이다. 본 실시예에서는, 주로 위에서 소개한 인코더측에 방법을 적용한 예를 사용하여 설명한다. 방법은 다음 동작(1101 내지 1102)을 포함할 수 있다:

[0236] 동작 1101: 인코딩될 비디오 프레임에 대응하는 유효 QM을 결정하고, 유효 QM은 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이다.

[0237] 인코딩될 비디오 프레임은 인코딩될 비디오에서 인코딩될 임의의 비디오 프레임(또는 이미지 프레임으로 지칭됨)일 수 있다.

[0238] 변환 계수를 양자화하는 데 사용될 수 있는 QM의 양이 n이고, 유효 QM의 양이 n보다 작거나 n과 같을 수 있다고 가정하고, n은 양의 정수이다. 예를 들어, 모든 n개의 QM이 실제로 변환 계수를 양자화하는 데 사용되는 경우 유효 QM의 양은 n이다. 변환 계수를 양자화하는 데 실제로 사용되는 모든 n개의 QM에서 일부 QM(예를 들어, m개의 QM, m은 n보다 작은 양의 정수)인 경우 유효 QM의 양은 m이다.

[0239] 선택적으로, 유효 QM이 아닌 임의의 다른 QM에 대해 QM의 모든 요소는 디폴트 값에 의해 사전 정의된다. 예를 들어, 디폴트 값은 16일 수 있다. 수식 1을 참조하면, 이 경우 TB의 모든 변환 계수의 스케일링 양자화 계수는 1이므로 QM을 사용하지 않는 것과 같은 효과가 있다.

- [0240] 동작 1102: 유효 QM 및 유효 QM을 결정하는 데 사용되는 선택스 요소를 인코딩하여 제1 파라미터 세트에 대응하는 비트스트림을 생성하고, 제1 파라미터 세트는 QM을 정의하는 데 사용되는 관련 선택스 요소를 포함하는 파라미터 세트이다.
- [0241] 유효 QM이 결정된 후, 하나 이상의 유효 QM이 있을 수 있기 때문에 인코더 측에서는 각각의 유효 QM을 개별적으로 인코딩할 필요가 있다. 유효 QM이 예로 사용된다. 유효 QM을 인코딩할 때, 유효 QM에 대응하는 최적의 모드를 결정한 후, 최적의 모드에 따라 유효 QM을 인코딩한다. 최적 모드는 전술한 3가지 후보 모드: 인터-프레임 예측 모드의 사본 모드, 인터-프레임 예측 모드의 예측 모드 및 인트라-프레임 예측 모드 중에서 선택된 가장 낮은 비트 코스트를 갖는 모드일 수 있다.
- [0242] 예를 들어, 표 1을 참조하면, 변환 계수를 양자화하는 데 사용될 수 있는 QM의 양은 28일 수 있다. 12개의 QM이 유효 QM으로 결정된다고 가정하고, 인코더 측 장치는 다음을 인코딩하기만 하면 된다. 12개의 유효 QM이 있고 다른 16개의 무효 QM을 인코딩할 필요가 없다.
- [0243] 이 외에, 유효 QM을 인코딩하는 것 외에도 인코더 측 장치는 유효 QM을 결정하는 데 사용되는 선택스 요소도 인코딩하여 디코더 측 장치가 선택스 요소에 따라 유효 QM을 결정하도록 해야 한다. 인코더 측 장치는 유효 QM 및 유효 QM을 결정하는 데 사용되는 선택스 요소를 인코딩하여 제1 파라미터 세트에 대응하는 비트스트림을 생성한다. 제1 파라미터 세트는 APS일 수 있거나, QM을 정의하는 데 사용되는 관련 선택스 요소의 다른 파라미터 세트일 수 있다. 이는 일 예로서 제공되는 것으로, 실시예가 이에 한정되는 것은 아니다.
- [0244] 상기에 기초하여, 본 실시예에서 제공되는 기술 솔루션에서, 인코딩될 비디오 프레임에 대응하는 유효 QM이 결정되고, 유효 QM은 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이고; 그런 다음 유효 QM 및 유효 QM을 결정하는 데 사용되는 선택스 요소가 인코딩되어 제1 파라미터 세트에 대응하는 비트스트림을 생성한다. 이러한 방식으로 인코더 측에서는 유효 QM만 인코딩하여 전송하므로 QM 시그널링이 점유해야 하는 코드워드를 줄이고 비트 오버헤드를 줄이는 데 도움이 된다. 디코더 측에서는 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소한다.
- [0245] 또한, 인코더 측 장치의 인코딩 프로세스는 디코더 측 장치의 디코딩 프로세스에 대응한다. 상세하게 설명되지 않은 인코딩 프로세스의 세부사항에 대해서는 디코딩 프로세스 실시예와 관련된 전술한 설명을 참조할 수 있으며, 세부사항은 여기서 다시 설명하지 않는다.
- [0246] 방법 실시예를 수행하는 데 사용될 수 있는 장치 실시예가 아래에 설명된다. 장치 실시예에서 개시되지 않은 세부사항에 대해서는 방법 실시예를 참조할 수 있다.
- [0247] 도 12는 일 실시예에 따른 비디오 디코딩 장치의 블록도이다. 상기 장치는 전술한 비디오 디코딩 방법의 예를 구현하는 기능을 가지며, 그 기능은 하드웨어 또는 해당 소프트웨어를 실행하는 하드웨어에 의해 구현될 수 있다. 장치는 위에서 설명된 디코더 측 장치일 수 있거나 디코더 측 장치에 배치될 수 있다. 장치(1200)는 파라미터 획득 모듈(1210), QM 결정 모듈(1220), 및 QM 디코딩 모듈(1230)을 포함할 수 있다.
- [0248] 파라미터 획득 모듈(1210)은 디코딩될 비디오 프레임에 대응하는 제1 파라미터 세트를 획득하도록 구성되며, 제1 파라미터 세트는 QM을 정의하는 데 사용되는 관련 선택스 요소의 파라미터 세트이다.
- [0249] QM 결정 모듈(1220)은 제1 파라미터 세트에 포함된 선택스 요소에 따라 유효 QM을 결정하도록 구성되며, 유효 QM은 디코딩될 비디오 프레임의 디코딩 동안 양자화된 변환 계수를 역양자화하는 데 실제로 사용되는 QM이다.
- [0250] QM 디코딩 모듈(1230)은 유효 QM을 디코딩하도록 구성된다.
- [0251] 일 실시예에서, 도 13에 도시된 바와 같이, QM 결정 모듈(1220)은 범위 결정 유닛(1221) 및 QM 결정 유닛(1222)을 포함한다.
- [0252] 범위 결정 유닛(1221)은 제1 파라미터 세트에 포함된 선택스 요소에 따라 QM의 유효 크기 범위를 결정하도록 구성된다.
- [0253] QM 결정 유닛(1222)은 유효 크기 범위에 속하는 QM을 유효 QM으로 결정하도록 구성된다.
- [0254] 일 실시예에서, 범위 결정 유닛(1221)은:
- [0255] 제1 파라미터 세트에 포함된 선택스 요소에 따라 최소 루마 코딩 블록 크기, 루마 코딩 트리의 블록 크기, 및 최대 루마 TB 크기를 결정하고;

- [0256] 최소 루마 코딩 블록 크기, 루마 코딩 트리의 블록 크기, 및 최대 루마 TB 크기에 따라 루마 QM의 유효 크기 범위를 결정하고, 루마 QM의 유효 크기 범위는 루마 QM의 최소 크기 및 최대 크기를 포함하며; 그리고
- [0257] 루마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하도록 구성되고, 크로마 QM의 유효 크기 범위는 크로마 QM의 최소 크기 및 최대 크기를 포함한다.
- [0258] 일 실시예에서, 범위 결정 유닛(1221)은:
- [0259] 최소 루마 코딩 블록 크기에 따라 루마 QM의 최소 크기를 결정하고; 그리고
- [0260] 루마 코딩 트리의 블록 크기 및 최대 루마 TB 크기 중 더 큰 값을 루마 QM의 최대 크기로 결정하도록 구성된다.
- [0261] 일 실시예에서, 범위 결정 유닛(1221)은:
- [0262] 루마 QM의 최소 크기 및 최대 크기를 포함하는 루마 QM의 유효 크기 범위인 제1 파라미터 세트에 포함된 선택스 요소에 따라 루마 QM의 유효 크기 범위를 결정하고; 그리고
- [0263] 크로마 QM의 유효 크기 범위 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 유효 크기 범위를 결정하도록 구성되고, 크로마 QM의 유효 크기 범위는 크로마 QM의 최소 크기 및 최대 크기를 포함한다.
- [0264] 일 실시예에서, 범위 결정 유닛(1221)은:
- [0265] 루마 QM의 최소 크기 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 최소 크기를 계산하고; 그리고
- [0266] 루마 QM의 최대 크기 및 루마 성분에 대한 크로마 성분의 샘플링 레이트에 따라 크로마 QM의 최대 크기를 계산하도록 구성된다.
- [0267] 일 실시예에서, QM 결정 유닛(1222)은:
- [0268] 제1 QM이 제1 조건 및 제2 조건 중 하나를 충족하는 경우 제1 QM을 유효 QM으로 결정하고,
- [0269] 제1 조건은 `cIdx==0 &&(matrixQMSize >= minQMSizeY && matrixQMSize <= maxQMSizeY)`이고, 제1 QM이 루마 성분이고 루마 TB의 양자화 프로세스에 사용됨을 나타내며, 제1 QM은 루마 QM의 유효 크기 범위 [MinQMSizeY, MaxQMSizeY] 내에 있고, MinQMSizeY는 루마 QM의 최소 크기를 나타내고, MaxQMSizeY는 루마 QM의 최대 크기를 나타내며; 그리고
- [0270] 제2 조건은 `cIdx!=0 &&(matrixQMSize >= minQMSizeUV && matrixQMSize <= maxQMSizeUV)`이고, 제1 QM이 크로마 성분이고 크로마 TB의 양자화 프로세스에 사용됨을 나타내며, 제1 QM은 크로마 QM의 유효 크기 범위 [MinQMSizeUV, MaxQMSizeUV] 내에 있고, MinQMSizeUV는 크로마 QM의 최소 크기를 나타내고, MaxQMSizeUV는 크로마 QM의 최대 크기를 나타낸다.
- [0271] 일 실시예에서, 도 13에 도시된 바와 같이, QM 결정 모듈(1220)은 요소 판독 유닛(1223) 및 QM 결정 유닛(1224)을 포함한다.
- [0272] 요소 판독 유닛(1223)은 제1 파라미터 세트로부터 제1 QM에 대응하는 플래그 선택스 요소의 값을 판독하도록 구성된다.
- [0273] QM 결정 유닛(1224)은: 제1 QM에 대응하는 플래그 선택스 요소의 값이 제1 값인 경우, 제1 QM이 유효 QM이라고 결정하고; 제1 QM에 대응하는 플래그 선택스 요소의 값이 제2 값인 경우, 제1 QM이 유효 QM이 아니라고 결정한다.
- [0274] 일 실시예에서, 동일한 예측 모드 및 동일한 크기를 갖는 제1 크로마 QM 및 제2 크로마 QM은 동일한 플래그 선택스 요소를 공유한다.
- [0275] 일 실시예에서, 플래그 선택스 요소는 `scaling_matrix_present_flag`이다.
- [0276] 일 실시예에서, 제1 파라미터 세트는 APS이다.
- [0277] 일 실시예에서, 유효 QM이 아닌 임의의 다른 QM에 대해, QM의 모든 요소는 디폴트 값에 의해 미리 정의된다.
- [0278] 일 실시예에서, 디폴트 값은 16이다.
- [0279] 상기에 기초하여, 이 실시예에서 제공되는 기술 솔루션에서, 디코딩될 비디오 프레임에 대응하는 제1 파라미터

세트가 획득되고; 유효 QM은 제1 파라미터 세트에 포함된 선택스 요소에 따라 결정되며, 유효 QM은 인코딩을 통해 디코딩될 비디오 프레임의 생성 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이고; 그런 다음 유효 QM이 디코딩된다. 이러한 방식으로 디코더 측에서는 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소한다.

[0280] 도 14는 일 실시예에 따른 비디오 인코딩 장치의 블록도이다. 장치는 전술한 비디오 인코딩 방법의 예를 구현하는 기능을 가지며, 그 기능은 하드웨어에 의해 또는 해당 소프트웨어를 실행하는 하드웨어에 의해 구현될 수 있다. 장치는 위에서 설명된 인코더 측 장치일 수 있거나 인코더 측 장치에 배치될 수 있다. 장치(1400)는 QM 결정 모듈(1410) 및 QM 인코딩 모듈(1420)을 포함할 수 있다.

[0281] QM 결정 모듈(1410)은 인코딩될 비디오 프레임에 대응하는 유효 QM을 결정하도록 구성되며, 유효 QM은 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이다.

[0282] QM 인코딩 모듈(1420)은 유효 QM 및 유효 QM을 결정하는 데 사용되는 선택스 요소를 인코딩하여 제1 파라미터 세트에 대응하는 비트스트림을 생성하도록 구성되며, 제1 파라미터 세트는 사용된 관련 선택스 요소의 파라미터 세트임 QM을 정의한다.

[0283] 상기에 기초하여, 인코딩될 비디오 프레임에 대응하는 유효 QM이 결정되고, 유효 QM은 인코딩될 비디오 프레임의 인코딩 동안 변환 계수를 양자화하기 위해 실제로 사용되는 QM이고; 유효 QM 및 유효 QM을 결정하는 데 사용되는 선택스 요소가 인코딩되어 제1 파라미터 세트에 대응하는 비트스트림을 생성한다. 이러한 방식으로 인코더 측에서는 유효 QM만 인코딩하여 전송하므로 QM 시그널링이 점유해야 하는 코드워드를 줄이고 비트 오버헤드를 줄이는 데 도움이 된다. 디코더 측에서는 유효 QM을 디코딩하기만 하면 되므로 디코더 측의 계산 복잡도가 감소한다.

[0284] 전술한 실시예에서 제공된 장치가 장치의 기능을 구현하는 경우, 전술한 기능 모듈의 분할은 설명을 위한 예시일 뿐이다. 실제 응용에서 요구 사항에 따라 기능이 다른 기능 모듈에 할당되고 완료될 수 있다. 즉, 장치의 내부 구조가 다른 기능 모듈로 분할되어 위에서 설명한 기능의 전부 또는 일부를 구현한다. 또한, 전술한 실시예에서 제공된 장치 및 방법 실시예는 동일한 개념에 속한다. 특정 구현 프로세스에 대해 방법 실시예를 참조할 수 있으며 세부 사항은 여기에서 다시 설명되지 않는다.

[0285] 도 15는 일 실시예에 따른 컴퓨터 장치의 개략적인 블록도이다. 컴퓨터 디바이스는 위에서 설명된 인코더 측 디바이스일 수 있거나, 또는 위에서 설명된 디코더 측 디바이스일 수 있다. 컴퓨터 장치(150)는 프로세서(151), 메모리(152), 통신 인터페이스(153), 인코더/디코더(154), 및 버스(155)를 포함할 수 있다.

[0286] 프로세서(151)는 하나 이상의 프로세싱 코어를 포함하고, 프로세서(151)는 소프트웨어 프로그램 및 모듈을 실행하여 다양한 기능적 애플리케이션 및 정보 처리를 수행한다.

[0287] 메모리(152)는 컴퓨터 프로그램을 저장하도록 구성될 수 있다. 프로세서(151)는 컴퓨터 프로그램을 실행하거나, 전술한 비디오 인코딩 방법을 구현하거나, 전술한 비디오 디코딩 방법을 구현하도록 구성된다.

[0288] 통신 인터페이스(153)는 예를 들어 오디오 및 비디오 데이터를 수신/전송하는 다른 장치와 통신하도록 구성될 수 있다.

[0289] 인코더/디코더(154)는 인코딩 및 디코딩 기능, 예를 들어 오디오 및 비디오 데이터의 인코딩 및 디코딩을 구현하도록 구성될 수 있다.

[0290] 메모리(152)는 버스(155)를 통해 프로세서(151)에 연결된다.

[0291] 또한, 메모리(152)는 임의의 유형의 휘발성 또는 비휘발성 저장 장치 또는 이들의 조합으로 구현될 수 있다. 휘발성 또는 비휘발성 저장 장치에는 자기 디스크, 광 디스크, 전기적으로 지울 수 있는 프로그래머블 읽기 전용 메모리(electrically-erasable programmable read-only memory, EEPROM), 지울 수 있는 프로그래머블 읽기 전용 메모리(erasable programmable read-only memory, EPROM), 정적 랜덤 액세스 메모리(static random access memory, SRAM), 읽기 전용 메모리(read-only memory, ROM), 자기 메모리, 플래시 메모리 또는 프로그래머블 읽기 전용 메모리(programmable read-only memory, PROM)를 포함하되 이에 제한되지 않는다.

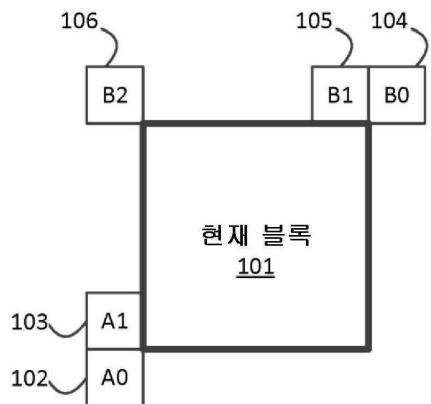
[0292] 당업자는 도 15에 도시된 구조는 컴퓨터 장치(150)에 대한 어떠한 제한도 구성하지 않으며, 컴퓨터 장치는 도면에 도시된 것보다 더 많은 구성요소 또는 더 적은 구성요소를 포함할 수 있거나, 일부 구성요소가 결합될 수 있거나, 다른 구성요소 배치가 사용될 수 있다는 것을 이해할 수 있을 것이다. 컴퓨터 판독 가능형 저장 매체는

비 일시적일 수 있다.

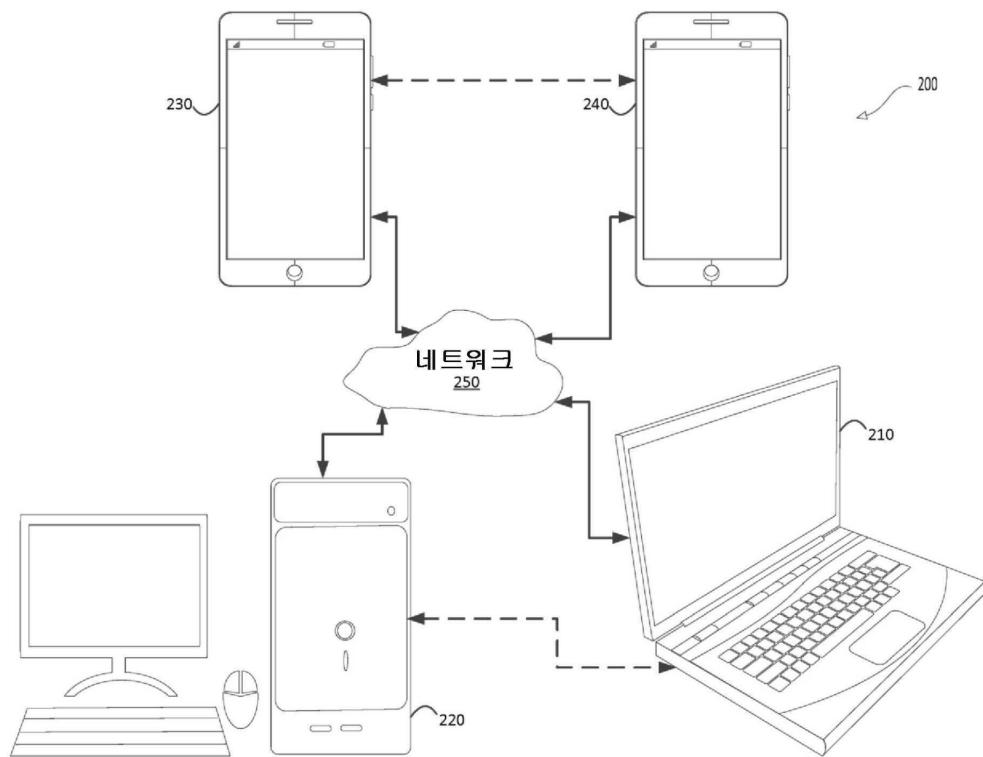
- [0293] 일 실시예에서, 컴퓨터 판독 가능형 저장 매체가 제공되며, 컴퓨터 판독 가능형 저장 매체는 적어도 하나의 명령, 적어도 하나의 프로그램, 코드 세트, 또는 명령 세트를 저장하며, 적어도 하나의 명령, 적어도 하나의 프로그램, 코드 세트 또는 명령 세트는 프로세서에 의해 실행될 때 전술한 비디오 디코딩 방법을 구현하거나 전술한 비디오 인코딩 방법을 구현한다.
- [0294] 실시예에서, 컴퓨터 프로그램 제품이 더 제공되며, 컴퓨터 프로그램 제품은 프로세서에 의해 실행될 때 전술한 비디오 디코딩 방법을 구현하거나 전술한 비디오 인코딩 방법을 구현하도록 구성된다.
- [0295] 명세서에서 언급된 복수는 둘 이상을 의미한다. "및/또는"은 연관된 개체에 대한 연관 관계를 설명하고 세 가지 관계가 존재할 수 있음을 나타낸다. 예를 들어, A 및/또는 B는 A만 존재하고, A와 B가 모두 존재하고, B만 존재하는 세 가지 경우를 나타낼 수 있다. 문자 "/"는 일반적으로 연결된 개체 간의 "또는" 관계를 나타낸다.
- [0296] 전술한 설명은 단지 본 출원의 실시예일 뿐, 본 출원을 제한하려는 의도는 아니다. 본 출원의 정신과 원칙 내에서 이루어진 모든 수정, 동등한 교체 또는 개선은 본 출원의 보호 범위에 속한다.

도면

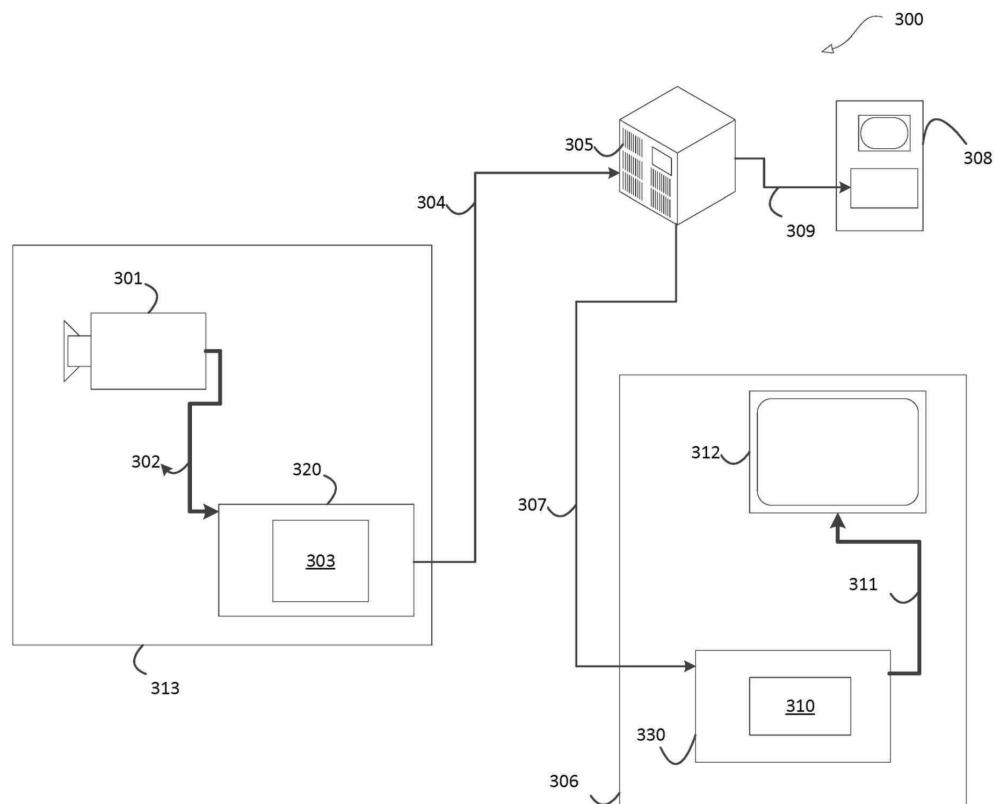
도면1

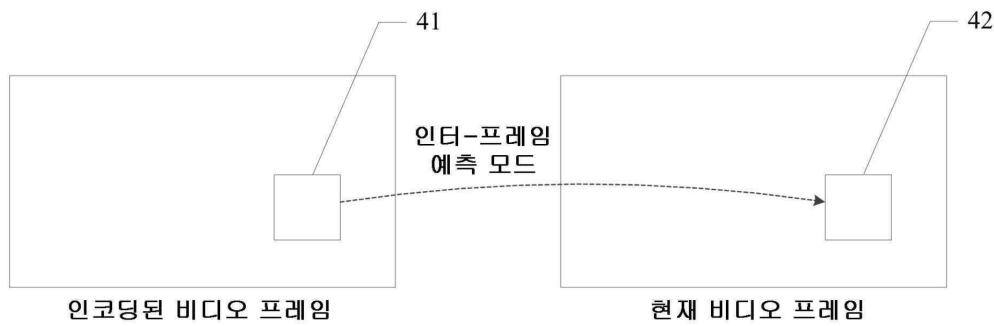
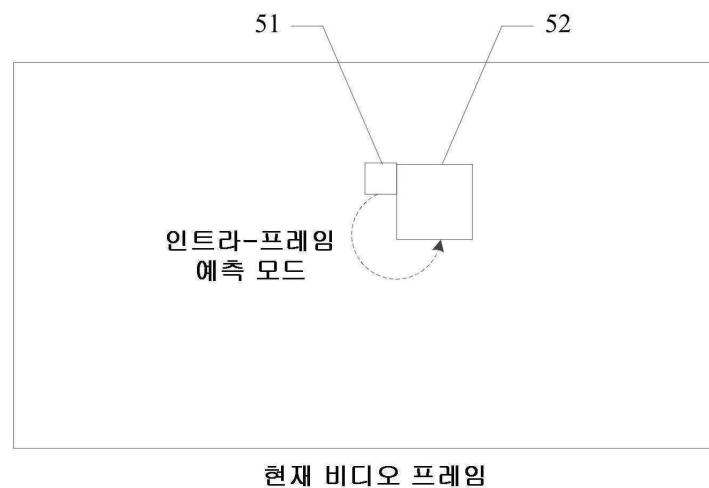
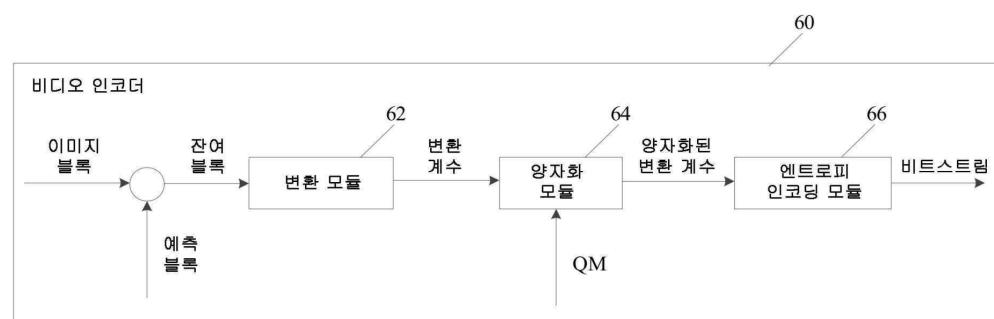


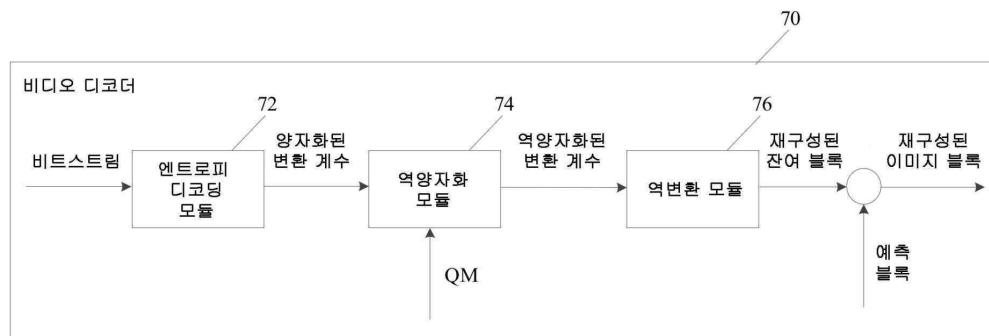
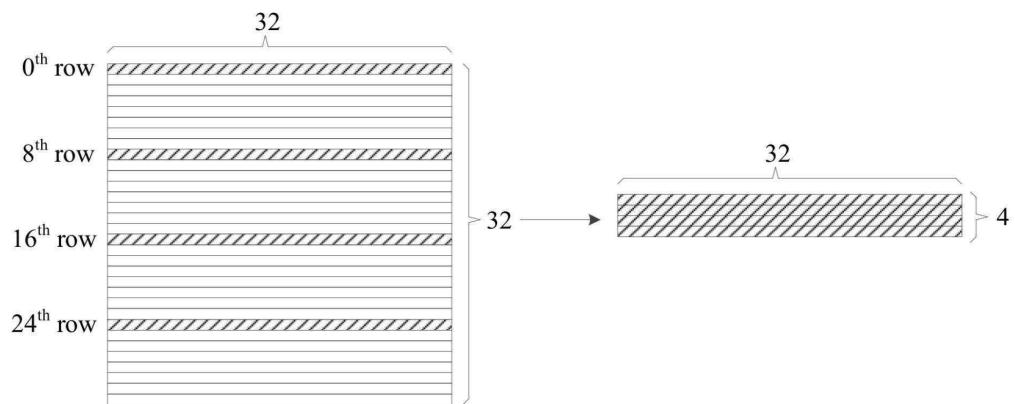
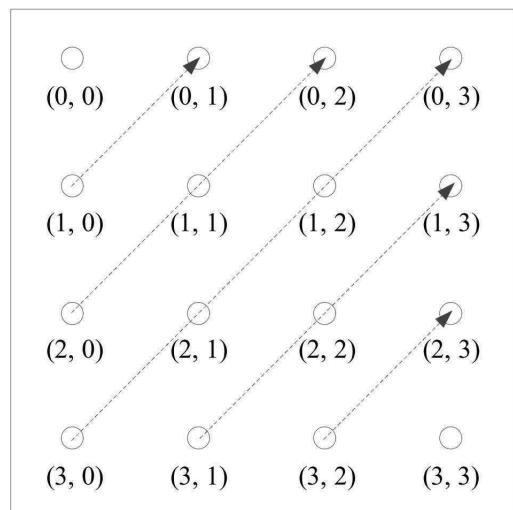
도면2

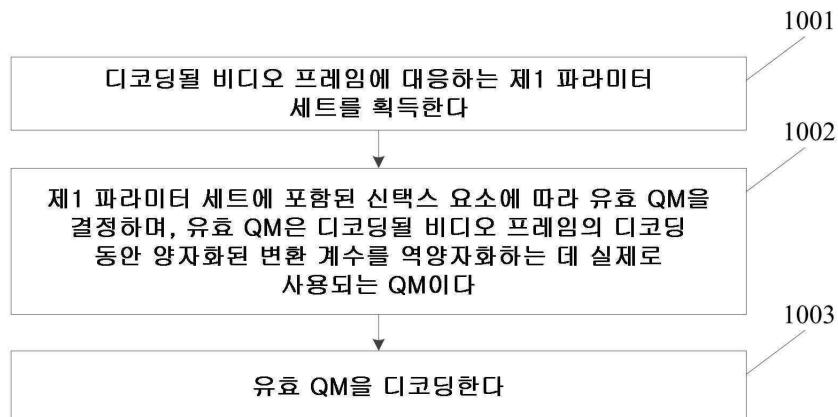
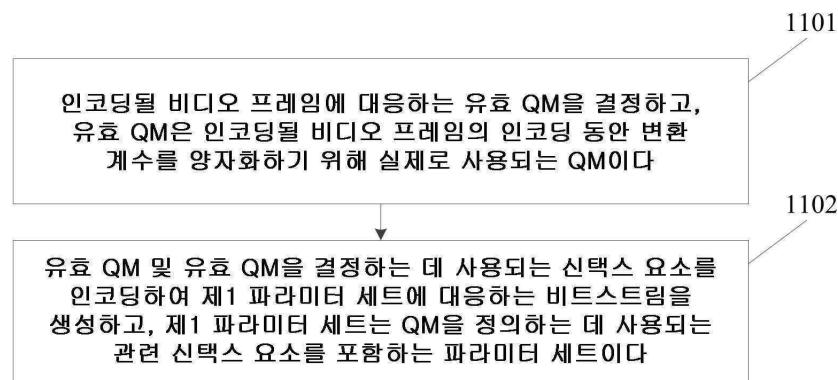


도면3

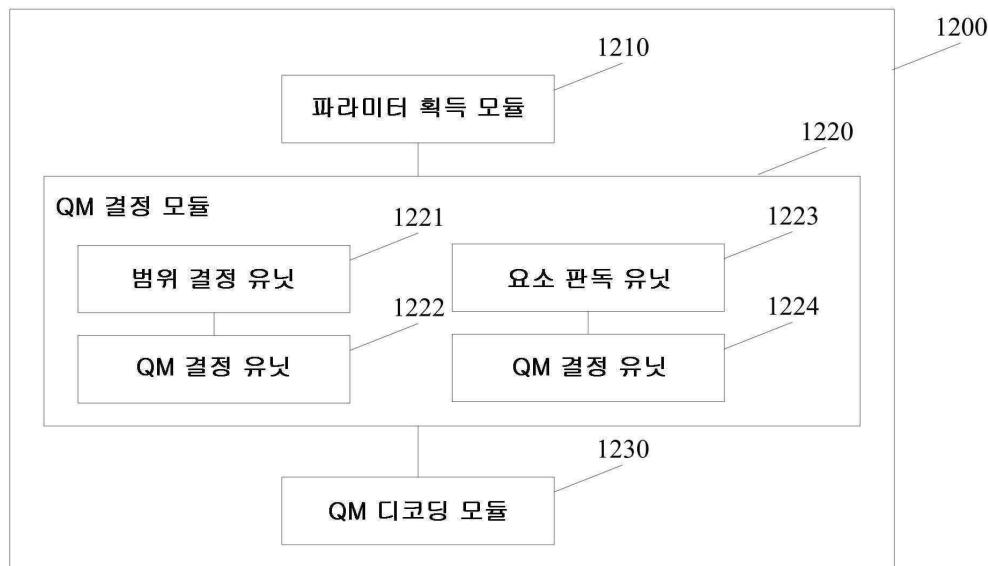


도면4**도면5****도면6**

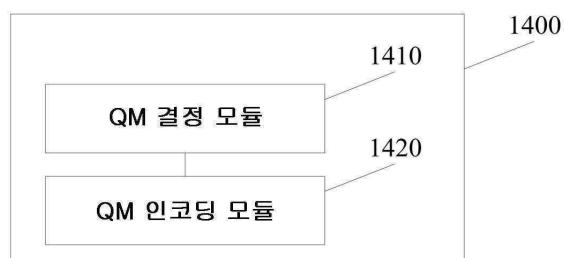
도면7**도면8****도면9**

도면10**도면11****도면12**

도면13



도면14



도면15

